Rissanen Data Analysis: Examining Dataset Characteristics via Description Length

Ethan Perez [1] **Douwe Kiela** [2] **Kyunghyun Cho** [1 3]

Abstract

We introduce a method to determine if a certain capability helps to achieve an accurate model of given data. We view labels as being generated from the inputs by a program composed of subroutines with different capabilities, and we posit that a subroutine is useful if and only if the minimal program that invokes it is shorter than the one that does not. Since minimum program length is uncomputable, we instead estimate the labels' minimum description length (MDL) as a proxy, giving us a theoretically-grounded method for analyzing dataset characteristics. We call the method Rissanen Data Analysis (RDA) after the father of MDL, and we showcase its applicability on a wide variety of settings in NLP, ranging from evaluating the utility of generating subquestions before answering a question, to analyzing the value of rationales and explanations, to investigating the importance of different parts of speech, and uncovering dataset gender bias.[1]

1. Introduction

In many practical learning scenarios, it is useful to know what capabilities would help to achieve a good model of the data. According to Occam's Razor, a good model is one that provides a simple explanation for the data (Blumer et al., 1987), which means that the capability to perform a task is helpful when it enables us to find simpler explanations of the data. Kolmogorov complexity (Kolmogorov, 1968) formalizes the notion of simplicity as the length of the shortest program required to generate the labels of the data given the inputs. In this work, we estimate the Kolmogorov complexity of the data by approximately computing the data's Minimum Description Length (MDL; Rissanen, 1978), and we examine how the data complexity changes as we add or remove different features from the input. We name our method Rissanen Data Analysis (RDA) after the father of the MDL principle, and we use it to examine several open questions about popular datasets, with a focus on NLP.

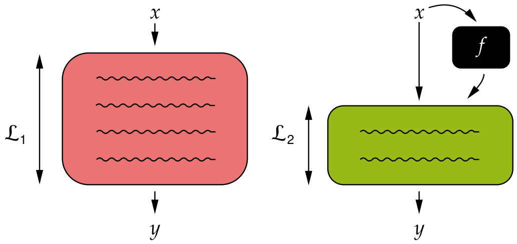

Figure 1. A capability f is useful if it shortens the minimum program needed to perform a task, as measured by Minimum Description Lengths $\mathcal{L}_1$ and $\mathcal{L}_2$. To give a concrete example, if x is a question and y is an answer, then f can be an oracle that answers relevant subquestions.

We view a capability as a function $f(x)$ that transforms x in some way (e.g., adding a feature), and we say that f is helpful if invoking it leads to a shorter minimum program for mapping x to the corresponding label in a dataset (see Fig. 1 for an illustration). Finding a short program is equivalent to finding a compressed version of the labels given the inputs, since the program can be run to generate the labels. Thus, we can measure the shortest program's length by estimating the labels' maximally compressed length, or Minimum Description Length (MDL; Rissanen, 1978; Grünwald, 2004). While prior work in machine learning uses MDL for model optimization (Hinton & van Camp, 1993), selection (Yogatama et al., 2019), and model probing (Voita & Titov, 2020; Lovering et al., 2021), we use MDL for a very different end: to understand the data itself ("dataset probing").

RDA addresses empirical and theoretical inadequacies of prior data analysis methods. For example, two common approaches are to evaluate the performance of a model when the inputs are modified or ablated (1) at training and test time or (2) at test time only. Training time input modification has been used to evaluate the usefulness of the

[1]New York University [2]Facebook AI Research [3]CIFAR Fellow in Learning in Machines & Brains. Correspondence to: Ethan Perez <perez@nyu.edu>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021. Copyright 2021 by the author(s).

[1] Code at https://github.com/ethanjperez/rda along with a script to conduct RDA on your own dataset.

capability to decompose a question into subquestions (Min et al., 2019b; Perez et al., 2020), to access the image for image-based question-answering (Antol et al., 2015; Zhang et al., 2016a), and to view the premise when detecting if it entails a hypothesis (Gururangan et al., 2018; Poliak et al., 2018; Tsuchiya, 2018). However, these works evaluate performance only on held-out dev examples, a fraction of the total examples in the dataset, which also are often drawn from a different distribution (e.g., in terms of quality). To understand what datasets teach our models, we must examine the entire dataset, which also gives us more examples for evaluation. Furthermore, a capability's usefulness to a model in high-data regimes does not necessarily reflect its usefulness in low-data regimes, which have increasingly become of interest (Lake et al., 2017; Guzmán et al., 2019; Brown et al., 2020). Test time ablation has been used to evaluate the capability to view word order (Pham et al., 2020; Sinha et al., 2020; Gupta et al., 2021) or words of different types (Sugawara et al., 2020), or to perform multi-hop reasoning (Jiang & Bansal, 2019). However, it is hard to rule out factors that may explain poor performance (e.g., distribution shift) or good performance (e.g., other ways to solve a problem). In our work, we examine an intrinsic property of the dataset, MDL, and we provide a theoretical argument justifying why it is the correct measure to use.

We use RDA to provide insights on a variety of datasets. First, we verify that description length is reduced when we invoke a capability f that is known to be helpful on a carefully-controlled synthetic task. Next, we examine HOTPOTQA (Yang et al., 2018), a benchmark for answering questions, where prior work has both claimed that decomposing questions into subquestions is helpful (Min et al., 2019b; Perez et al., 2020) and called such claims into question (Min et al., 2019a; Jiang & Bansal, 2019; Chen & Durrett, 2019). RDA shows that subquestions are indeed helpful and exposes how evaluation procedures in prior work may have caused the value of question decomposition to be underestimated. We then evaluate if explanations are useful for recognizing textual entailment using the e-SNLI dataset (Camburu et al., 2018). Both written explanations and decision-relevant keyword markings ("rationales") are helpful, but rationales are more useful than explanations. Lastly, we examine a variety of popular NLP tasks, evaluating the extent to which they require relying on word order, different types of words, and gender bias. Overall, our results indicate that RDA can be used to answer a broad variety of questions about datasets.

2. Rissanen Data Analysis

How can we determine whether or not a certain capability $f(x)$ is helpful for building a good model of the data? To answer this question, we view a dataset with inputs $x_{1:N}$ and labels $y_{1:N}$ as generated by a program that maps $x_n \to y_n$. Let the length of the shortest such program P be $\mathcal{L}(y_{1:N}|x_{1:N})$, the data's Kolmogorov complexity. We view a capability as a function f that maps x_n to a possibly helpful output $f(x_n)$, with $\mathcal{L}(y_{1:N}|x_{1:N}, f)$ being the length of the shortest label-generating program when access to f is given. We say that f is helpful exactly when:

$$\mathcal{L}(y_{1:N}|x_{1:N}, f) < \mathcal{L}(y_{1:N}|x_{1:N}) \tag{1}$$

2.1. Minimum Description Length

To use Eq. 1 in practice, we need to find the shortest program P, which is uncomputable in general. However, because P is a program that generates $y_{1:N}$ given $x_{1:N}$, we can instead consider any compressed version of $y_{1:N}$, along with an accompanying decompression algorithm that produces $y_{1:N}$ given $x_{1:N}$ and the compressed $y_{1:N}$. To find $\mathcal{L}$, then, we find the length of the maximally compressed $y_{1:N}$, or Minimum Description Length (MDL; Rissanen, 1978). While MDL is not computable, just like Kolmogorov complexity, many methods have been proposed to estimate MDL by restricting the set of allowed compression algorithms (see Grünwald, 2004, for an overview). These methods are all compatible with RDA, and here, we use online (or prequential) coding (Rissanen, 1984; Dawid, 1984), an effective method for estimating MDL when used with deep learning (Blier & Ollivier, 2018).

2.2. Online Coding

To examine how much $y_{1:N}$ can be compressed, we look at the minimum number of bits (minimal codelength) needed by a sender Alice to transmit $y_{1:N}$ to a receiver Bob, when both share $x_{1:N}$. Without loss of generality, we assume y_n is an element from a finite set. In online coding, Alice first sends Bob the learning algorithm $\mathcal{A}$, including the model architecture, trainable parameters θ, optimization procedure, hyperparameter selection method, initialization scheme, random seed, and pseudo-random number generator. Alice and Bob each initialize a model p_{θ_1} using the random seed and pseudo-random number generator, such that both models are identical.

Next, Alice sends each label y_n one by one. Shannon (1948) showed that there exists a minimum code to send y_n with $-\log_2 p_{\theta_n}(y_n|x_n)$ bits when Alice and Bob share p_{θ_n} and x_n. After Alice sends y_n, Alice and Bob use $\mathcal{A}$ to train a better model $p_{\theta_{n+1}}(y|x)$ on $(x_{1:n}, y_{1:n})$ to get shorter codes for future labels. The codelength for $y_{1:N}$ is then:

$$\mathcal{L}_p(y_{1:N}|x_{1:N}) = \sum_{n=1}^{N} -\log_2 p_{\theta_n}(y_n|x_n). \tag{2}$$

Intuitively, $\mathcal{L}_p(y_{1:N}|x_{1:N})$ is the area under the "online" learning curve that shows how the cross-entropy loss goes down as the training set size increases.

Overall, Alice's message consists of $\mathcal{A}$ plus the label encoding ($\mathcal{L}_p(y_{1:N}|x_{1:N})$ bits). When Alice and Bob share f, Alice's message consists of $\mathcal{A}$ plus $\mathcal{L}_p(y_{1:N}|x_{1:N}, f)$ bits to encode the labels with a model $p_\theta(y|x, f)$. f is helpful when the message is shorter with f than without, i.e., when:

$$\mathcal{L}_p(y_{1:N}|x_{1:N}, f) < \mathcal{L}_p(y_{1:N}|x_{1:N})$$

2.3. Practical Implementation with Block-wise Coding

The online code in Eq. 2 is expensive to compute. It has a computational complexity that is quadratic in N (assuming linear time learning), which is prohibitive for large N and compute-intensive $\mathcal{A}$. Following Blier & Ollivier (2018), we upper bound online codelength by having Alice and Bob only train the model upon having sent $0 = t_0 < t_1 < \cdots < t_S = N$ labels. Alice thus sends all labels in a "block" $y_{t_s+1:t_{s+1}}$ at once using $p_{\theta_{t_s}}$, giving codelength:

$$\bar{\mathcal{L}}_p(y_{1:N}|x_{1:N}) = \sum_{s=0}^{S-1} \sum_{n=t_s+1}^{t_{s+1}} -\log_2 p_{\theta_{t_s}}(y_n|x_n)$$

Since θ_{t_0} has no training data, Alice sends Bob the first block using a uniform prior.

Alleviating the sensitivity to learning algorithm To limit the effect of the choice of learning algorithm $\mathcal{A}$, we may ensemble many model classes. To do so, we have Alice train M models of different classes and send the next block's labels using the model that gives the shortest codelength. To tell Bob which model to use to decompress a block's labels, Alice also sends $\log_2 M$ bits per block $s = 1, \ldots, S-1$, adding $(S-1)\ \log_2 M$ to MDL. In this way, MDL relies less on the behavior of a single model class.

2.4. Experimental Setup

To evaluate MDL, we first randomly sort examples in the dataset. We use $S = 9$ blocks where $t_0 = 0$ and $t_1 = 64 < \cdots < t_S = N$ such that $\frac{t_{s+1}}{t_s}$ is constant (log-uniform spacing). To train a model on the first s blocks, we split the available examples into train (90%) and dev (10%) sets, choosing hyperparameters and early stopping epoch using dev loss (codelength). We otherwise follow each model's training strategy and hyperparameter ranges as suggested by its original paper. We then evaluate the codelength of the $(s+1)$-th block. As a baseline, we show $\mathcal{H}(y)$, the codelength with the label prior $p(y)$ as p_θ.

Various random factors impact MDL, such as the order of examples, model initialization, and randomness during training. Thus, we report the mean and std. error of MDL over 5 random seeds. For computational efficiency, we only sweep over hyperparameters for the first random seed and reuse the best hyperparameters for the remaining seeds. For all experiments, our code and reported codelengths are available at https://github.com/ethanjperez/rda, along with a script to conduct RDA with your own models and datasets.

3. Validating Rissanen Data Analysis

Having described our experimental setup, we now verify that $\bar{\mathcal{L}}_p(y_{1:N}|x_{1:N}, f) < \bar{\mathcal{L}}_p(y_{1:N}|x_{1:N})$ holds in practice when we use an f that we know is helpful. To this end, we use CLEVR (Johnson et al., 2017), an image-based question-answering (QA) dataset. CLEVR is a synthetic dataset where many questions are designed to benefit from answering subquestions. For example, to answer the CLEVR question "*Are there more cubes than spheres?*" it helps to know the answer to the subquestions "*How many cubes are there?*" and "*How many spheres are there?*" We hypothesize that MDL decreases as we give a model answers to subquestions.

We test our hypothesis on three types of CLEVR questions. "Integer Comparison" questions ask to compare the numbers of two kinds of objects and have two subquestions (example above). "Attribute Comparison" questions ask to compare the properties of two objects, i.e., "*Is the metal object the same color as the rubber thing?*", where there are two subquestions which each ask about the property of a single object, i.e., "*What color is the metal object?*" and "*What color is the rubber thing?*" "Same Property As" questions ask whether or not one object has the same property as another object, i.e., "*What material is the sphere with the same color as the rubber cylinder?*", where there is one subquestion that asks about a property of one object, i.e., "*What color is the rubber cylinder?*" Since CLEVR is synthetic, we obtain oracle answers to subquestions ("subanswers") programmatically, using the ground-truth question programs given by CLEVR. We append subanswers to the question (in order), and we evaluate the utility of providing 0-2 subanswers.

Model We use the FiLM model from Perez et al. (2018) which combines a convolutional network for the image with a GRU for the question (Cho et al., 2014). The model minimizes cross-entropy loss (27-way classification). We follow training strategy from Perez et al. (2018) using the public code, except we train for at most 20 epochs (not 80), since we only train on subsets of CLEVR.

Results Fig. 2 shows codelengths (left) and MDL (right). For all question types, $\bar{\mathcal{L}}_p(y_{1:N}|x_{1:N}, f) < \bar{\mathcal{L}}_p(y_{1:N}|x_{1:N})$ when all oracle subanswers are given, as expected. For "Integer Comparison" (top) and "Attribute Comparison"

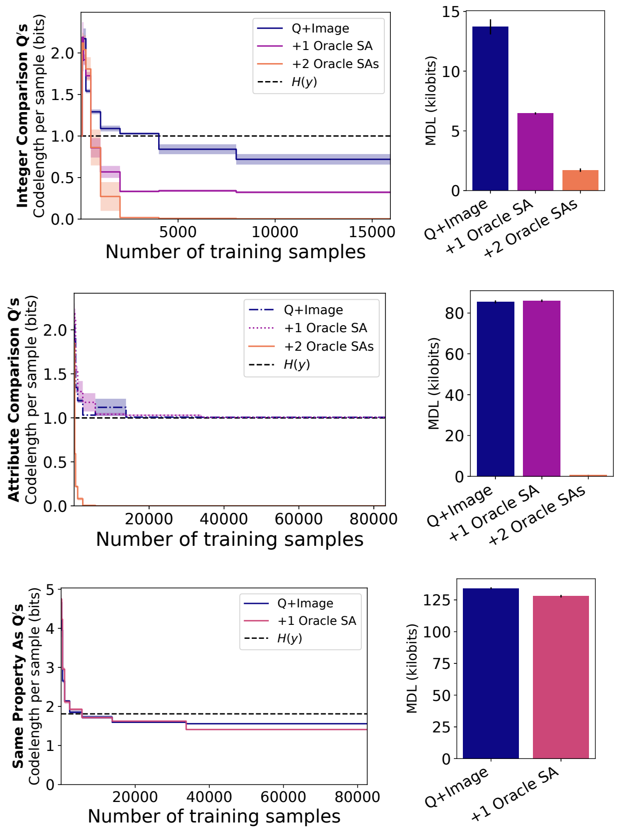

Figure 2. **Left**: Answer codelengths for different CLEVR question types with/without adding oracle answers to subquestions ("subanswers") to the input. **Right**: Subanswers reduce MDL.

(middle), the reduction in MDL is larger than for "Same Property As" questions (bottom). For comparison questions, the subanswers can be used without the image to determine the answer, explaining the larger decreases in MDL. Our results align with our expectations about when answers to subquestions are helpful, empirically validating RDA.

4. Examining Dataset Characteristics

We now use MDL to determine what capabilities are helpful on popular datasets with pertinent open questions.

4.1. Is it helpful to answer subquestions?

Yang et al. (2018) proposed HOTPOTQA as a dataset that benefits from decomposing questions into subquestions, but recent work has called the benefit into doubt (Min et al., 2019a; Jiang & Bansal, 2019; Chen & Durrett, 2019) while there is also evidence that decomposition helps (Min et al., 2019b; Perez et al., 2020). We use RDA to determine if subquestions and their answers are useful.

Dataset HOTPOTQA consists of crowdsourced questions (*"Are Coldplay and Pierre Bouvier from the same country?"*) whose answers are intended to rely on information from two Wikipedia paragraphs. The input consists of these two "supporting" paragraphs, 8 "distractor" paragraphs, and the question. Answers are either `yes`, `no`, or a text span in an input paragraph.

Model We use the LONGFORMER (Beltagy et al., 2020), a transformer (Vaswani et al., 2017) modified to handle long inputs as in HOTPOTQA. We evaluate MDL for two models, the official $\text{LONGFORMER}_{\text{BASE}}$ initialized with pretrained weights trained on language modeling and another model with the same architecture that we train from scratch, which we refer to as $\text{TRANSFORMER}_{\text{BASE}}$. We train the model to predict the span's start token and end token by minimizing the negative log-likelihood for each prediction. We treat yes/no questions as span prediction as well by prepending `yes` and `no` to the input, following Perez et al. (2020). We use the implementation from Wolf et al. (2020). See Appendix §B.2 for hyperparameters.

Providing subanswers We consider a subanswer to be a paragraph containing question-relevant information, because Perez et al. (2020) claimed that subquestions help by using a QA model to find question-relevant text. We indicate up to two subanswers to the model by prepending "$>$" to the first subanswer paragraph and "$\gg$" to the second.

Selecting subanswers We consider 5 methods for selecting subanswers. First, we use the two supporting paragraphs as oracle subanswers. Next, we consider the answers to subquestions generated by four different methods. Three are unsupervised methods from Perez et al. (2020): pseudo-decomposition (retrieval-based subquestions), seq2seq (subquestions from a sequence-to-sequence model), and ONUS (One-to-N Unsupervised Sequence transduction). Last, we test the ability of a more recent, large language model (GPT3; Brown et al., 2020) to generate subquestions using a few labeled question-decomposition examples. Since generating with GPT3 is expensive, we use its generated subquestions as training data for a smaller T5 model (Raffel et al., 2020), a "Distilled Language Model" (DLM, see Appendix §B.1 for details). To answer generated subquestions, we use the same QA model from Perez et al. (2020), an ensemble of two $\text{ROBERTA}_{\text{LARGE}}$ (Liu et al., 2019) models finetuned on SQuAD (Rajpurkar et al., 2016) to predict answer spans. We use the paragraphs containing predicted answer spans to subquestions as subanswers.

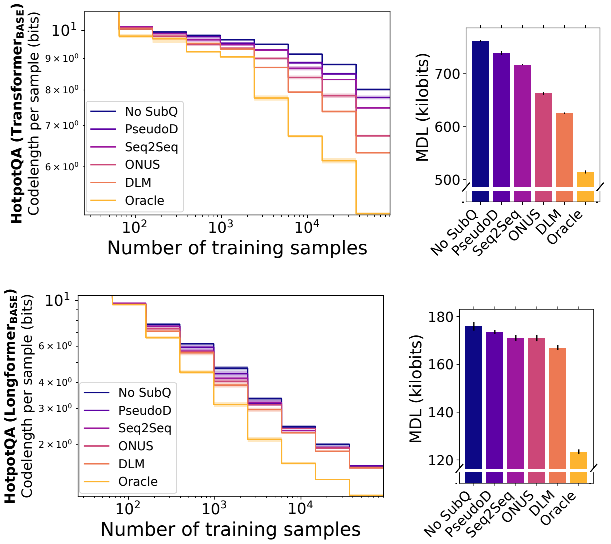

Figure 3. **Left**: Codelengths for HotpotQA when encoding labels with Longformer$_{\text{BASE}}$ trained from scratch (top) or pretrained weights (bottom), with the answers to subquestions (subanswers) from various decomposition methods. (Plots on log-log scale.) **Right**: MDL for decomposition methods when training from scratch (top) or pretrained weights (bottom). Subanswers help to compress the answers, especially when training from scratch, but with much room for improvement w.r.t. oracle subanswers.

4.1.1. Results

Fig. 3 shows codelengths (left) and MDL (right). For Transformer$_{\text{BASE}}$ (top), decompositions consistently and significantly reduces codelength and MDL. Decomposition methods vary in how much they reduce MDL, ranked from worst to best as: no decomposition, Pseudo-Decomposition, Seq2Seq, ONUS, DLM, and oracle. Overall, the capability to answer subquestions reduces program length, especially when subquestions and their answers are of high quality.

For Longformer$_{\text{BASE}}$ (Fig. 3 bottom), all decomposition methods also reduce codelength and MDL, though to a lesser extent. To examine why, we plot the codelength reduction from decomposition against the original codelength for Longformer$_{\text{BASE}}$ in Fig. 4 (left). As the original codelength decreases, the benefit from decomposition increases, until the no-decomposition baseline reaches a certain loss, at which point the benefit from decomposition decreases. We hypothesize that a certain, minimum amount of task understanding is necessary before decompositions are useful (see Appendix §B.3 for similar findings with Transformer$_{\text{BASE}}$). However, as loss decreases, the task-relevant capabilities can be learned from the data directly, without decomposition.

Our finding suggests that decompositions help disproportionately in the high/mid- loss regimes rather than the low-loss regime, where QA systems are usually evaluated (i.e., when training on all examples). The limited value in low-loss regimes occurs because models approach the same, minimum loss $H(y|x)$ in the limit of dataset size. Our observation partly explains why earlier work (Min et al., 2019a; Chen & Durrett, 2019), which only evaluated final performance, drew the conclusion that HotpotQA does not benefit much from multi-step reasoning or question decomposition. In contrast, MDL actually does capture differences in performance across data regimes, showing that RDA is the right approach going forward, especially given the growing interest in few-shot data regimes (Lake et al., 2017; Brown et al., 2020)

4.2. Are Explanations and Rationales Useful?

Recent work has proposed methods that give reasons for an answer before predicting the answer, to improve performance. Such reasons may come in the form of written explanations (Camburu et al., 2018; Rajani et al., 2019; Wiegreffe et al., 2020) or locating task-relevant input words (Zhang et al., 2016b; Perez et al., 2019; Pruthi et al., 2020). As a testbed, such work often uses natural language inference (NLI) – checking if a premise entails or contradicts (or neither) a hypothesis. To explore if this direction is promising, we use RDA to evaluate whether providing a reason is a useful capability, using NLI as a case study.

Dataset We use the e-SNLI (Camburu et al., 2018) dataset, which annotated each example in SNLI (Bowman et al., 2015) with two forms of reasons: an extractive rationale that marks entailment-relevant words and a written explanation of the right answer. We randomly sample 10k examples from e-SNLI to examine the usefulness of rationales and explanations. To illustrate, e-SNLI contains an example of contradiction where the premise is "*A man and a woman are dancing in the* ***crowd****.*" and the hypothesis is "*A man and woman dance* ***alone****.*" The rationale is bolded, and the explanation is "*Being in a crowd means not alone.*"

Adding explanations and rationales We view rationales/explanations as generated by a function f executed on the input. To test if f reduces MDL, we add the rationale by surrounding each entailment-relevant word with asterisks, and we add the explanation before the hypothesis, separated by a special token. For comparison, we also evaluate MDL when including only the explanation as input and only the rationale patterns as input. For the latter, we include the rationale without the actual premise/hypothesis words by replacing each rationale word with "*" and other words with "_".

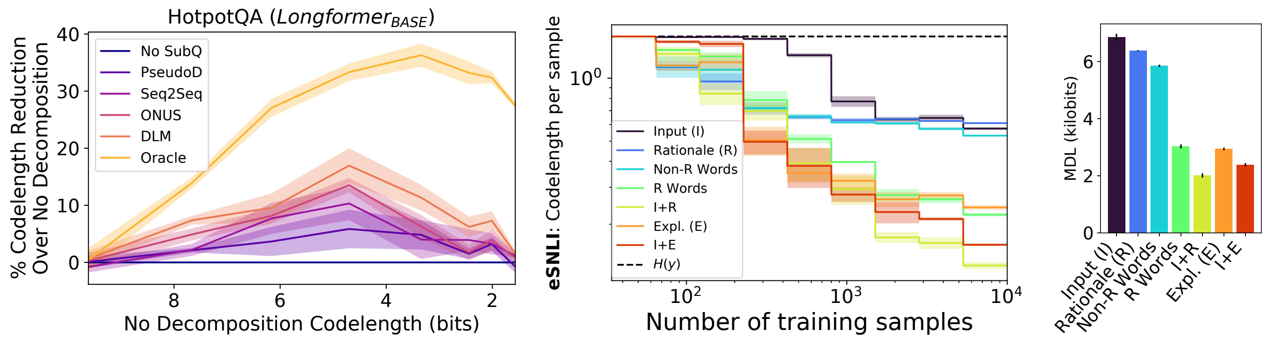

Figure 4. **Left**: On HOTPOTQA, the reduction in codelength over the no-decomposition baseline from using subanswers from various decomposition methods (mean and std. err. for $\text{LONGFORMER}_{\text{BASE}}$). **Middle**: Codelengths for e-SNLI with/without extractive rationales or written explanations. **Right**: On e-SNLI, MDL reduces significantly when rationales and explanations are given alongside the input.

Model We use an ensemble model composed of the following model classes: FastText Bag-of-Words (Joulin et al., 2017), transformers (Vaswani et al., 2017) trained from scratch (110M and 340M parameter versions), $\text{BART}_{\text{BASE}}$ (encoder-decoder; Lewis et al., 2020), $\text{ALBERT}_{\text{BASE}}$ (encoder-only; Lan et al., 2020), $\text{RoBERTa}_{\text{BASE}}$ and $\text{RoBERTa}_{\text{LARGE}}$ (encoder-only; Liu et al., 2019) and the distilled version DistilRoBERTa (Sanh et al., 2019), and GPT2 (decoder-only; Radford et al., 2019) and DistilGPT2 (Sanh et al., 2019). For each model, we minimize cross-entropy loss and tune softmax temperature[2] on dev to alleviate overconfidence on unseen examples (Guo et al., 2017; Desai & Durrett, 2020). We follow each models' official training strategy and hyperparameter sweeps (Appendix §B.4), using the FastText codebase[3] and HuggingFace Transformers (Wolf et al., 2020) with PyTorch Lightning (Falcon et al., 2019) for other models.

4.2.1. Results

Fig. 4 shows codelengths (middle) and MDL (right). Adding rationales to the input greatly reduces MDL compared to using the normal input ("Input (I)") or rationale markings without input words ("Rationale (R)"), suggesting that rationales complement the input. The reduction comes from focusing on rationale words specifically. We see almost as large MDL reductions when only including rationale-marked words and masking non-rationale words ("R Words" vs. "I+R"). In contrast, we see little improvement over rationale markings alone when using only non-rationale words with rationale words masked ("Rationale (R)" vs."Non-R Words"). Our results show that for NLI, it is useful to first determine task-relevant words, suggesting directions for future work along the lines of Zhang et al. (2016b); Perez et al. (2019).

Similarly, explanations greatly reduce MDL (Fig. 4 right, rightmost two bars), especially when the input is also provided. This finding shows that explanations, like rationales, are also complementary to the input. Interestingly, adding rationales to the input reduces MDL more than adding explanations, suggesting that while explanations are useful, they are harder to use for label compression than rationales.

4.3. Examining Text Datasets

So far, we used RDA to determine when adding input features helps reduce label description lengths. Similarly, we evaluate when removing certain features increases description length, to determine what features help achieve a small MDL. Here, we view the "original" input as having certain features missing, and we evaluate the utility of a capability f that recovers the missing features to return the normal task input. If f reduces the label-generating program length, then it is useful to have access to f (the ablated features). To illustrate, we evaluate the usefulness of different kinds of words and of word order on the General Language Understanding Evaluation benchmark (GLUE; Wang et al., 2019), a central evaluation suite in NLP, as well as SNLI and Adversarial NLI (ANLI; Nie et al., 2020).

Datasets GLUE consists of 9 tasks (8 classification, 1 regression).[4] CoLA and SST-2 are single-sentence classification tasks. MRPC, QQP, and STS-B involve determining if two sentences are similar or paraphrases of each other. QNLI, RTE, MNLI, and WNLI are NLI tasks (we omit WNLI due to its size, 634 training examples). ANLI consists of NLI data collected in three rounds, where annotators wrote hypotheses that fooled state-of-the-art

[2] Search over $[10^{-1}, 10^2]$, 1000 log-uniformly spaced samples.

[3] https://github.com/facebookresearch/fastText

[4] See Appendix §A.1 for details on GLUE and Appendix §B.3 for details on regression.

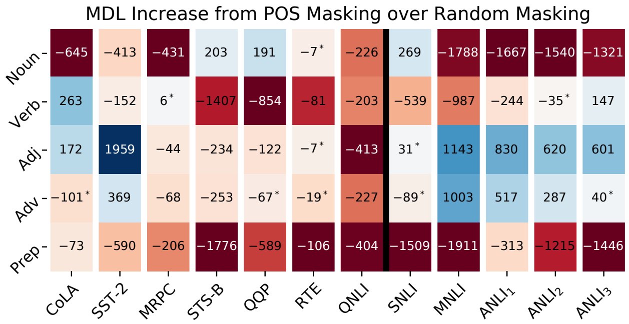

Figure 5. The importance of different POS words, given by $\text{MDL}_{-\text{POS}} - \text{MDL}_{-\text{Random}}$. 0 indicates that words of a given POS are as important as randomly-chosen words, while > 0 and < 0 indicate greater and lesser importance than randomly-chosen words, respectively. (*) indicates within std. error of 0. Color is normalized by column (dataset).

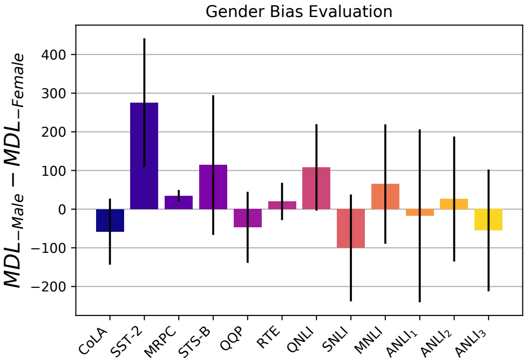

Figure 6. **Gender Bias**: MDL when masking masculine vs. feminine words (mean and std. err. over 5 random seeds). Values above zero (vs. below zero) indicate that male-gendered words (vs. female-gendered words) are more important for compressing labels. SST-2 shows the largest bias (male-favored).

NLI models trained on data from the previous round. We consider each round as a separate dataset, to examine how NLI datasets have evolved over time, from SNLI to MNLI to ANLI_1, ANLI_2, and ANLI_3.

Experimental setup We follow a similar setup as for e-SNLI (§4.2), using the 10-model ensemble and evaluating MDL on up to 10k examples per task.

4.3.1. The usefulness of part-of-speech words

We consider the original input to be the full input with words of a certain POS masked out ("_") and evaluate the utility of a capability f that fills in the masked words. To control for the number of words masked, we restrict f such that it returns a version of the input with the same proportion of words masked, chosen uniformly at random. If f is useful, then words of a given type are more useful for compression than randomly-chosen input words. In particular, we report the difference between MDL when (1) words of a given POS are masked and (2) the same fraction of words are masked uniformly at random: $\text{MDL}_{-\text{POS}} - \text{MDL}_{-\text{Random}}$. We evaluate nouns, verbs, adjectives, adverbs, and prepositions.[5]

We show results in Figure 5. Adjectives are much more useful than other POS for SST-2, a sentiment analysis task where relevant terms are evidently descriptive words (e.g., "the service was *terrible*"). For CoLA, verbs play an important role in determining if a sentence is linguistically acceptable, likely due to the many examples evaluating verb argument structure (e.g., "The toast burned." vs. "The toast buttered."). Other tasks (MRPC, RTE, and QNLI) do not rely significantly on any one POS, suggesting that they require reasoning over multiple POS in tandem. Nouns are consistently less useful on NLI tasks, suggesting that NLI datasets should be supplemented with knowledge-intensive tasks like open-domain QA that rely on names and entities, in order to holistically evaluate language understanding. Prepositions are not important for any GLUE task, suggesting where GLUE can be complemented with other tasks (e.g., from Kim et al., 2019) and illustrating how RDA can be used to help form comprehensive benchmarks in the future.

4.3.2. How useful are other word types?

Sugawara et al. (2020) hypothesized other word types that may be useful for NLP tasks. We use RDA to assess their usefulness as we did above (see Appendix §C for details). GLUE tasks vary in their reliance on "content" words. Logical words like *not* and *every* are particularly important for MNLI which involves detecting logical entailment. On the other hand, causal words (e.g., *because*, *since*, and *therefore*) are not particularly useful for GLUE.

4.3.3. Do Datasets Suffer from Gender Bias?

Gender bias in data is a prevalent issue in machine learning (Bolukbasi et al., 2016; Blodgett et al., 2020). For example, prior work found that machine learning systems are worse at classifying images of women (Phillips et al., 2000; Buolamwini & Gebru, 2018), at speech recognition for women and speakers from Scotland (Tatman, 2017), and at POS tagging for African American vernacular (Jørgensen et al., 2015). RDA can be used to diagnose such biases. Here, we do so by masking male-gendered words and

[5] We use POS tags from spaCy's large English model (Honnibal & Montani, 2017). For computational reasons, we omit other POS, as they occur less frequently and masking them did not greatly impact MDL in preliminary experiments.

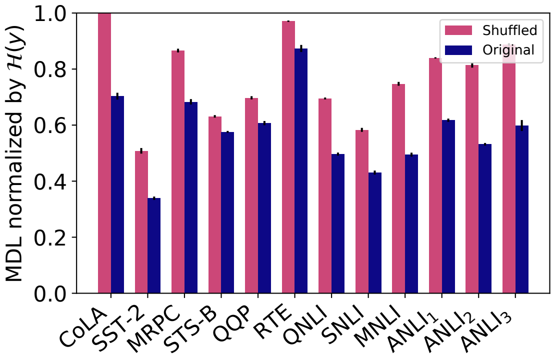

Figure 7. **MDL with/without Word Shuffling**, normalized by the MDL when encoding labels with $p(y)$ for reference. Word order reduces MDL on all tasks.

evaluating the utility of an oracle function f that reveals male-gendered words while masking female-gendered words. If f is useful and $\text{MDL}_{-\text{Male}} - \text{MDL}_{-\text{Female}} > 0$, then masculine words are more useful than feminine words for the dataset (gender bias). We use male and female word lists from Dinan et al. (2020a;b). The two lists are similar in size (∼530 words each) and POS distribution (52% nouns, 29% verbs, 18% adjectives), and the male- and female-gendered words occur with similar frequency. See Appendix §C for experiments controlling for word frequency.

Fig. 6 shows the results. Masculine words are more useful for SST-2 and MRPC while no GLUE datasets have feminine words as more useful. For SST-2, feminine words occur more frequently than masculine words (2.7% vs. 2.2%, evenly distributed across class labels), suggesting that RDA uncovers a gender bias that word counts do not. This result highlights the practical value of RDA in uncovering where evaluation benchmarks under-evaluate the performance of NLP systems on text related to different demographic groups.

4.3.4. How useful is word order?

Recent work claims that state-of-the-art NLP models do not use word order on GLUE (Pham et al., 2020; Sinha et al., 2020; Gupta et al., 2021). We use RDA to examine the utility of word order on GLUE by testing the value of unshuffling input words when they have been shuffled.

Fig. 7 shows MDL with and without shuffling, normalized by the MDL of the label-only prior $p(y)$ as a baseline. Word order helps to obtain smaller MDL on all tasks. For example, on MNLI, adding word order enables the labels to be compressed from $75\% \rightarrow 50\%$ of the baseline compression rate. For CoLA, the linguistic acceptability task, input word order is necessary to compress labels at all. Prior work may have come to different conclusions about the utility of word order because they evaluate the behavior of trained models on out-of-distribution (word-shuffled) text, while RDA estimates an intrinsic property of the dataset.

5. Related Work

In addition to prior work on data analysis (§1), there has been much related work on model analysis (e.g., Shi et al., 2016; Alain & Bengio, 2017; Conneau et al., 2018; Jia & Liang, 2017). This line of work sometimes uses similar techniques, such as input replacement (Perez et al., 2019; Jiang & Bansal, 2019; Pham et al., 2020; Sinha et al., 2020; Gupta et al., 2021) and estimating description length (Voita & Titov, 2020; Whitney et al., 2020; Lovering et al., 2021) or other information-theoretic measures (Pimentel et al., 2020), but for a very different end: to understand how models behave and what their representations encode. While model probing can uncover characteristics of the training data (e.g., race and gender bias; Caliskan et al., 2017), models also reflect other aspects of learning (Zhao et al., 2017), such as the optimization procedure, inductive bias of the model class and architecture, hyperparameters, and randomness during training. Instead of indirectly examining a dataset by probing models, we directly estimate a property intrinsic to the dataset. For further related work, see Appendix §D.

6. Conclusion

In this work, we proposed Rissanen Data Analysis (RDA), a method for examining the characteristics of a dataset. We began by viewing the labels of a dataset as being generated by a program over the inputs, then positing that a capability is helpful if it reduces the length of the shortest label-generating program. Instead of evaluating minimum program length directly, we use block-wise prequential coding to upper bound Minimum Description Length (MDL). While the choice of learning algorithm $\mathcal{A}$ influences absolute MDL values, we only interpret MDL *relative* to other MDL values estimated with the same $\mathcal{A}$. In particular, we conduct RDA by comparing MDL with or without access to a subroutine with a certain capability, and we say that a capability is useful when invoking the subroutine reduces MDL.

We then conducted an extensive empirical analyses of various datasets with RDA. First, we showed that RDA provides intuitive results on a carefully-controlled synthetic task. Next, we used RDA to evaluate the utility of generating and answering subquestions in answering a question, finding that subquestions are indeed useful. For NLI, we found it helpful to include rationales and explanations. Finally, we showcased the general nature of RDA by applying it on a variety of other NLP tasks, uncovering the value of word order across all tasks, as well as the most useful parts of speech for different tasks, among other things. While we experimented on NLP tasks, RDA can be used in other domains as well, e.g. to determine the value of color in image recognition, temporal resolution in speech recognition, or different modalities for multimodal problems. Our work opens up ample opportunity for future work: automatically uncovering dataset biases when writing data statements (Gebru et al., 2018; Bender & Friedman, 2018), selecting the datasets to include in future benchmarks, discovering which capabilities are helpful for different tasks, and also expanding on RDA itself, e.g., by investigating the underlying data distribution rather than a particular dataset (Whitney et al., 2020). Overall, RDA is a theoretically-justified tool that is empirically useful for examining the characteristics of a wide variety of datasets.

Acknowledgments

We thank Paul Christiano, Sam Bowman, Tim Dettmers, Alex Warstadt, William Huang, Richard Pang, Yian Zhang, Tiago Pimentel, Patrick Lewis, and our anonymous reviewers for helpful feedback on the paper. We are grateful to OpenAI for providing access to GPT-3 via their API Academic Access Program. We also thank Will Whitney and Peter Hase for helpful discussions, Adina Williams for the list of gendered words, Shenglong Wang for cluster support, and Amanda Ngo for help with figure design. HOTPOTQA is licensed under CC BY-SA 4.0. KC is partly supported by Samsung Advanced Institute of Technology (Next Generation Deep Learning: from pattern recognition to AI). KC also thanks Naver, eBay, NVIDIA, and NSF Award 1922658 for support. EP is supported by the NSF Graduate Research Fellowship and the Open Philanthropy AI Fellowship.

References

Alain, G. and Bengio, Y. Understanding intermediate layers using linear classifier probes. In *ICLR*, 04 2017. URL https://arxiv.org/abs/1610.01644.

Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Zitnick, C. L., and Parikh, D. VQA: Visual Question Answering. In *ICCV*, 2015. URL https://arxiv.org/abs/1505.00468.

Baker, F. B. and Kim, S.-H. *Item response theory: Parameter estimation techniques*. CRC Press, 2004. URL https://www.jstor.org/stable/1435270.

Beltagy, I., Peters, M. E., and Cohan, A. Longformer: The long-document transformer. *arXiv:2004.05150*, 2020. URL https://arxiv.org/abs/2004.05150.

Bender, E. M. and Friedman, B. Data statements for natural language processing: Toward mitigating system bias and enabling better science. *Transactions of the ACL*, 6:587–604, 2018. doi: 10.1162/tacl_a_00041. URL https://www.aclweb.org/anthology/Q18-1041.

Bentivogli, L., Dagan, I., Dang, H. T., Giampiccolo, D., and Magnini, B. The fifth pascal recognizing textual entailment challenge. In *In Proc Text Analysis Conference (TAC'09*, 2009. URL http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.232.1231.

Blier, L. and Ollivier, Y. The description length of deep learning models. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *NeurIPS*, volume 31, pp. 2216–2226. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper/2018/file/3b712de48137572f3849aabd5666a4e3-Paper.pdf.

Blodgett, S. L., Barocas, S., Daumé III, H., and Wallach, H. Language (technology) is power: A critical survey of "bias" in NLP. In *ACL*, pp. 5454–5476, Online, July 2020. ACL. doi: 10.18653/v1/2020.acl-main.485. URL https://www.aclweb.org/anthology/2020.acl-main.485.

Blumer, A., Ehrenfeucht, A., Haussler, D., and Warmuth, M. K. Occam's razor. *Inf. Process. Lett.*, 24(6): 377–380, April 1987. ISSN 0020-0190. doi: 10.1016/0020-0190(87)90114-1. URL https://doi.org/10.1016/0020-0190(87)90114-1.

Bolukbasi, T., Chang, K.-W., Zou, J. Y., Saligrama, V., and Kalai, A. T. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., and Garnett, R. (eds.), *NeurIPS*, volume 29, pp. 4349–4357. Curran Associates, Inc., 2016. URL https://proceedings.neurips.cc/paper/2016/file/a486cd07e4ac3d270571622f4f316ec5-Paper.pdf.

Bowman, S. R., Angeli, G., Potts, C., and Manning, C. D. A large annotated corpus for learning natural language inference. In *EMNLP*, pp. 632–642, Lisbon, Portugal, September 2015. ACL. doi: 10.18653/v1/D15-1075. URL https://www.aclweb.org/anthology/D15-1075.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. Language models are few-shot learners, 2020. URL https://arxiv.org/abs/2005.14165.

Brunet, M.-E., Alkalay-Houlihan, C., Anderson, A., and Zemel, R. Understanding the origins of bias in word embeddings. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *ICML*, volume 97 of *PMLR*, pp. 803–811. PMLR, 09–15 Jun 2019. URL http://proceedings.mlr.press/v97/brunet19a.html.

Buolamwini, J. and Gebru, T. Gender shades: Intersectional accuracy disparities in commercial gender classification. In Friedler, S. A. and Wilson, C. (eds.), *Fairness, Accountability and Transparency*, volume 81 of *PMLR*, pp. 77–91, New York, NY, USA, 23–24 Feb 2018. PMLR. URL http://proceedings.mlr.press/v81/buolamwini18a.html.

Caliskan, A., Bryson, J., and Narayanan, A. Semantics derived automatically from language corpora contain human-like biases. *Science*, 356(6334):183–186, April 2017. ISSN 0036-8075. doi: 10.1126/science.aal4230.

Camburu, O.-M., Rocktäschel, T., Lukasiewicz, T., and Blunsom, P. e-snli: Natural language inference with natural language explanations. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *NeurIPS*, volume 31, pp. 9539–9549. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper/2018/file/4c7a167bb329bd92580a99ce422d6fa6-Paper.pdf.

Cer, D., Diab, M., Agirre, E., Lopez-Gazpio, I., and Specia, L. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *SemEval-2017*, pp. 1–14, Vancouver, Canada, August 2017. ACL. doi: 10.18653/v1/S17-2001. URL https://www.aclweb.org/anthology/S17-2001.

Chen, J. and Durrett, G. Understanding dataset design choices for multi-hop reasoning. In *NAACL, Volume 1 (Long and Short Papers)*, pp. 4026–4032, Minneapolis, Minnesota, June 2019. ACL. doi: 10.18653/v1/N19-1405. URL https://www.aclweb.org/anthology/N19-1405.

Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *EMNLP*, pp. 1724–1734, Doha, Qatar, October 2014. ACL. doi: 10.3115/v1/D14-1179. URL https://www.aclweb.org/anthology/D14-1179.

Conneau, A., Kruszewski, G., Lample, G., Barrault, L., and Baroni, M. What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties. In *ACL (Volume 1: Long Papers)*, pp. 2126–2136, Melbourne, Australia, July 2018. ACL. doi: 10.18653/v1/P18-1198. URL https://www.aclweb.org/anthology/P18-1198.

Cover, T. M. and Thomas, J. A. *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience, USA, 2006. ISBN 0471241954. URL http://staff.ustc.edu.cn/~cgong821/Wiley.Interscience.Elements.of.Information.Theory.Jul.2006.eBook-DDU.pdf.

Dawid, A. P. Present position and potential developments: Some personal views: Statistical theory: The prequential approach. *Journal of the Royal Statistical Society. Series A (General)*, 147(2):278–292, 1984. ISSN 00359238. URL http://www.jstor.org/stable/2981683.

Desai, S. and Durrett, G. Calibration of pre-trained transformers. In *EMNLP*, pp. 295–302, Online, November 2020. ACL. doi: 10.18653/v1/2020.emnlp-main.21. URL https://www.aclweb.org/anthology/2020.emnlp-main.21.

Dinan, E., Fan, A., Williams, A., Urbanek, J., Kiela, D., and Weston, J. Queens are powerful too: Mitigating gender bias in dialogue generation. In *EMNLP*, pp. 8173–8188, Online, November 2020a. ACL. doi: 10.18653/v1/2020.emnlp-main.656. URL https://www.aclweb.org/anthology/2020.emnlp-main.656.

Dinan, E., Fan, A., Wu, L., Weston, J., Kiela, D., and Williams, A. Multi-dimensional gender bias classification. In *EMNLP*, pp. 314–331, Online, November 2020b. ACL. doi: 10.18653/v1/2020.emnlp-main.23. URL https://www.aclweb.org/anthology/2020.emnlp-main.23.

Dixon, L., Li, J., Sorensen, J., Thain, N., and Vasserman, L. Measuring and mitigating unintended bias in text classification. In *AIES*, 2018. URL https://dl.acm.org/doi/10.1145/3278721.3278729.

Dolan, W. B. and Brockett, C. Automatically constructing a corpus of sentential paraphrases. In *International Workshop on Paraphrasing (IWP2005)*, 2005. URL https://www.aclweb.org/anthology/I05-5002.

Falcon et al., W. Pytorch lightning. *GitHub. Note: https://github.com/PyTorchLightning/pytorch-lightning*, 2019.

Gebru, T., Morgenstern, J., Vecchione, B., Vaughan, J. W., Wallach, H. M., III, H. D., and Crawford, K. Datasheets for datasets. *CoRR*, abs/1803.09010, 2018. URL http://arxiv.org/abs/1803.09010.

Grünwald, P. A tutorial introduction to the minimum description length principle. *CoRR*, math.ST/0406077, 06 2004. URL https://arxiv.org/abs/math/0406077.

Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. On calibration of modern neural networks. In *ICML*, PMLR, pp. 1321–1330. JMLR.org, 2017. URL http://proceedings.mlr.press/v70/guo17a.html.

Gupta, A., Kvernadze, G., and Srikumar, V. Bert & family eat word salad: Experiments with text understanding, 2021. URL https://arxiv.org/abs/2101.03453.

Gururangan, S., Swayamdipta, S., Levy, O., Schwartz, R., Bowman, S., and Smith, N. A. Annotation artifacts in natural language inference data. In *NAACL, Volume 2 (Short Papers)*, pp. 107–112, New Orleans, Louisiana, June 2018. ACL. doi: 10.18653/v1/N18-2017. URL https://www.aclweb.org/anthology/N18-2017.

Guzmán, F., Chen, P.-J., Ott, M., Pino, J., Lample, G., Koehn, P., Chaudhary, V., and Ranzato, M. The FLORES evaluation datasets for low-resource machine translation: Nepali–English and Sinhala–English. In *EMNLP*, pp. 6098–6111, Hong Kong, China, November 2019. ACL. doi: 10.18653/v1/D19-1632. URL https://www.aclweb.org/anthology/D19-1632.

Hinton, G. E. and van Camp, D. Keeping the neural networks simple by minimizing the description length of the weights. In *COLT*, COLT '93, pp. 5–13, New York, NY, USA, 1993. ACL. ISBN 0897916115. doi: 10.1145/168304.168306. URL https://doi.org/10.1145/168304.168306.

Holtzman, A., Buys, J., Du, L., Forbes, M., and Choi, Y. The curious case of neural text degeneration. In *ICLR*, 2020. URL https://openreview.net/forum?id=rygGQyrFvH.

Honnibal, M. and Montani, I. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear, 2017. URL https://github.com/explosion/spaCy.

Hopkins, M. and May, J. Models of translation competitions. In *ACL (Volume 1: Long Papers)*, pp. 1416–1424, Sofia, Bulgaria, August 2013. ACL. URL https://www.aclweb.org/anthology/P13-1139.

Jia, R. and Liang, P. Adversarial examples for evaluating reading comprehension systems. In *EMNLP*, pp. 2021–2031, Copenhagen, Denmark, September 2017. ACL. doi: 10.18653/v1/D17-1215. URL https://www.aclweb.org/anthology/D17-1215.

Jiang, Y. and Bansal, M. Avoiding reasoning shortcuts: Adversarial evaluation, training, and model development for multi-hop QA. In *ACL*, pp. 2726–2736, Florence, Italy, July 2019. ACL. doi: 10.18653/v1/P19-1262. URL https://www.aclweb.org/anthology/P19-1262.

Johnson, J., Hariharan, B., van der Maaten, L., Fei-Fei, L., Zitnick, C. L., and Girshick, R. B. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. *CVPR*, pp. 1988–1997, 2017. URL https://arxiv.org/abs/1612.06890.

Jørgensen, A., Hovy, D., and Søgaard, A. Challenges of studying and processing dialects in social media. In *Workshop on Noisy User-generated Text*, pp. 9–18, Beijing, China, July 2015. ACL. doi: 10.18653/v1/W15-4302. URL https://www.aclweb.org/anthology/W15-4302.

Joulin, A., Grave, E., Bojanowski, P., and Mikolov, T. Bag of tricks for efficient text classification. In *EACL: Volume 2, Short Papers*, pp. 427–431. ACL, April 2017. URL https://arxiv.org/abs/1607.01759.

Kim, N., Patel, R., Poliak, A., Xia, P., Wang, A., McCoy, T., Tenney, I., Ross, A., Linzen, T., Van Durme, B., Bowman, S. R., and Pavlick, E. Probing what different NLP tasks teach machines about function word comprehension. In *Proceedings of the Eighth Joint Conference on Lexical and Computational Semantics (*SEM 2019)*, pp. 235–249, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/S19-1026. URL https://www.aclweb.org/anthology/S19-1026.

Koh, P. W. and Liang, P. Understanding black-box predictions via influence functions. In Precup, D. and Teh, Y. W. (eds.), *ICML*, volume 70 of *PMLR*, pp. 1885–1894, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR. URL http://proceedings.mlr.press/v70/koh17a.html.

Kolmogorov, A. N. Three approaches to the quantitative definition of information. *IJCM*, 2(1-4):157–168, 1968. URL https://www.tandfonline.com/doi/abs/10.1080/00207166808803030.

Lake, B. M., Ullman, T. D., Tenenbaum, J. B., and Gershman, S. J. Building machines that learn and think like people. *Behavioral and Brain Sciences*, 40:

e253, 2017. doi: 10.1017/S0140525X16001837. URL https://arxiv.org/abs/1604.00289.

Lalor, J. P., Wu, H., and Yu, H. Learning latent parameters without human response patterns: Item response theory with artificial crowds. In *EMNLP*, pp. 4249–4259, Hong Kong, China, November 2019. ACL. doi: 10.18653/v1/D19-1434. URL https://www.aclweb.org/anthology/D19-1434.

Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., and Soricut, R. Albert: A lite bert for self-supervised learning of language representations. In *ICLR*, 2020. URL https://openreview.net/forum?id=H1eA7AEtvS.

Levesque, H. J., Davis, E., and Morgenstern, L. The winograd schema challenge. In *KR*, KR'12, pp. 552–561. AAAI Press, 2012. ISBN 9781577355601. URL https://cs.nyu.edu/faculty/davise/papers/WSKR2012.pdf.

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., and Zettlemoyer, L. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *ACL*, pp. 7871–7880, Online, July 2020. ACL. doi: 10.18653/v1/2020.acl-main.703. URL https://www.aclweb.org/anthology/2020.acl-main.703.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692, 2019. URL http://arxiv.org/abs/1907.11692.

Lovering, C., Jha, R., Linzen, T., and Pavlick, E. Predicting inductive biases of pre-trained models. In *ICLR*, 2021. URL https://openreview.net/forum?id=mNtmhaDkAr.

Martínez-Plumed, F., Prudêncio, R. B., Martínez-Usó, A., and Hernández-Orallo, J. Item response theory in ai: Analysing machine learning classifiers at the instance level. *Artificial Intelligence*, 271:18 – 42, 2019. ISSN 0004-3702. doi: https://doi.org/10.1016/j.artint.2018.09.004. URL http://www.sciencedirect.com/science/article/pii/S0004370219300220.

McCoy, T., Pavlick, E., and Linzen, T. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *ACL*, pp. 3428–3448, Florence, Italy, July 2019. ACL. doi: 10.18653/v1/P19-1334. URL https://www.aclweb.org/anthology/P19-1334.

Micikevicius, P., Narang, S., Alben, J., Diamos, G., Elsen, E., Garcia, D., Ginsburg, B., Houston, M., Kuchaiev, O., Venkatesh, G., and Wu, H. Mixed precision training. In *ICLR*, 2018. URL https://openreview.net/forum?id=r1gs9JgRZ.

Min, S., Wallace, E., Singh, S., Gardner, M., Hajishirzi, H., and Zettlemoyer, L. Compositional questions do not necessitate multi-hop reasoning. In *ACL*, pp. 4249–4257, Florence, Italy, July 2019a. ACL. doi: 10.18653/v1/P19-1416. URL https://www.aclweb.org/anthology/P19-1416.

Min, S., Zhong, V., Zettlemoyer, L., and Hajishirzi, H. Multi-hop reading comprehension through question decomposition and rescoring. In *ACL*, pp. 6097–6109, Florence, Italy, July 2019b. ACL. doi: 10.18653/v1/P19-1613. URL https://www.aclweb.org/anthology/P19-1613.

Nie, Y., Williams, A., Dinan, E., Bansal, M., Weston, J., and Kiela, D. Adversarial NLI: A new benchmark for natural language understanding. In *ACL*, pp. 4885–4901, Online, July 2020. ACL. doi: 10.18653/v1/2020.acl-main.441. URL https://www.aclweb.org/anthology/2020.acl-main.441.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. Bleu: a method for automatic evaluation of machine translation. In *ACL*, pp. 311–318, Philadelphia, Pennsylvania, USA, July 2002. ACL. doi: 10.3115/1073083.1073135. URL https://www.aclweb.org/anthology/P02-1040.

Perez, E., Strub, F., de Vries, H., Dumoulin, V., and Courville, A. Film: Visual reasoning with a general conditioning layer. *AAAI*, 32(1), Apr. 2018. URL https://ojs.aaai.org/index.php/AAAI/article/view/11671.

Perez, E., Karamcheti, S., Fergus, R., Weston, J., Kiela, D., and Cho, K. Finding generalizable evidence by learning to convince Q&A models. In *EMNLP*, pp. 2402–2411, Hong Kong, China, November 2019. ACL. doi: 10.18653/v1/D19-1244. URL https://www.aclweb.org/anthology/D19-1244.

Perez, E., Lewis, P., Yih, W.-t., Cho, K., and Kiela, D. Unsupervised question decomposition for question answering. In *EMNLP*, pp. 8864–8880, Online, November 2020. ACL. doi: 10.18653/v1/2020.emnlp-main.713. URL https://www.aclweb.org/anthology/2020.emnlp-main.713.

Pham, T. M., Bui, T., Mai, L., and Nguyen, A. Out of order: How important is the sequential order of words in a sentence in natural language understanding tasks?, 2020. URL https://arxiv.org/abs/2012.15180.

Phillips, P. J., Hyeonjoon Moon, Rizvi, S. A., and Rauss, P. J. The feret evaluation methodology for face-recognition algorithms. *TPAMI*, 22(10):1090–1104, 2000. doi: 10.1109/34.879790.

Pimentel, T., Valvoda, J., Hall Maudslay, R., Zmigrod, R., Williams, A., and Cotterell, R. Information-theoretic probing for linguistic structure. In *ACL*, pp. 4609–4622, Online, July 2020. ACL. doi: 10.18653/v1/2020.acl-main.420. URL https://www.aclweb.org/anthology/2020.acl-main.420.

Poliak, A., Naradowsky, J., Haldar, A., Rudinger, R., and Van Durme, B. Hypothesis only baselines in natural language inference. In *Joint Conference on Lexical and Computational Semantics*, pp. 180–191, New Orleans, Louisiana, June 2018. ACL. doi: 10.18653/v1/S18-2023. URL https://www.aclweb.org/anthology/S18-2023.

Pruthi, D., Dhingra, B., Soares, L. B., Collins, M., Lipton, Z., Neubig, G., and Cohen, W. W. Evaluating explanations: How much do explanations from the teacher aid students? *ArXiv*, abs/2012.00893, 2020.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. Language models are unsupervised multitask learners, 2019. URL https://d4mucfpksywv.cloudfront.net/better-language-models/language_models_are_unsupervised_multitask_learners.pdf.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. *JMLR*, 21(140):1–67, 2020. URL http://jmlr.org/papers/v21/20-074.html.

Rajani, N. F., McCann, B., Xiong, C., and Socher, R. Explain yourself! leveraging language models for commonsense reasoning. In *ACL*, pp. 4932–4942, Florence, Italy, July 2019. ACL. doi: 10.18653/v1/P19-1487. URL https://www.aclweb.org/anthology/P19-1487.

Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. SQuAD: 100,000+ questions for machine comprehension of text. In *EMNLP*, pp. 2383–2392, Austin, Texas, November 2016. ACL. doi: 10.18653/v1/D16-1264. URL https://www.aclweb.org/anthology/D16-1264.

Rissanen, J. Modeling by shortest data description. *Automatica*, 14(5):465 – 471, 1978. ISSN 0005-1098. doi: https://doi.org/10.1016/0005-1098(78)90005-5. URL http://www.sciencedirect.com/science/article/pii/0005109878900055.

Rissanen, J. Universal coding, information, prediction, and estimation. *IEEE Transactions on Information Theory*, 30(4):629–636, 1984. doi: 10.1109/TIT.1984.1056936.

Rudinger, R., May, C., and Van Durme, B. Social bias in elicited natural language inferences. In *ACL Workshop on Ethics in NLP*, pp. 74–79, Valencia, Spain, April 2017. ACL. doi: 10.18653/v1/W17-1609. URL https://www.aclweb.org/anthology/W17-1609.

Sanh, V., Debut, L., Chaumond, J., and Wolf, T. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *ArXiv*, abs/1910.01108, 2019. URL https://arxiv.org/abs/1910.01108.

Shannon, C. E. A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423, 1948. doi: 10.1002/j.1538-7305.1948.tb01338.x.

Shi, X., Padhi, I., and Knight, K. Does string-based neural MT learn source syntax? In *EMNLP*, pp. 1526–1534, Austin, Texas, November 2016. ACL. doi: 10.18653/v1/D16-1159. URL https://www.aclweb.org/anthology/D16-1159.

Sinha, K., Parthasarathi, P., Pineau, J., and Williams, A. Unnatural language inference, 2020.

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A., and Potts, C. Recursive deep models for semantic compositionality over a sentiment treebank. In *EMNLP*, pp. 1631–1642, Seattle, Washington, USA, October 2013. ACL. URL https://www.aclweb.org/anthology/D13-1170.

Sugawara, S., Stenetorp, P., Inui, K., and Aizawa, A. Assessing the benchmarking capacity of machine reading comprehension datasets. *AAAI*, 34(05): 8918–8927, Apr. 2020. doi: 10.1609/aaai.v34i05.6422. URL https://ojs.aaai.org/index.php/AAAI/article/view/6422.

Swayamdipta, S., Schwartz, R., Lourie, N., Wang, Y., Hajishirzi, H., Smith, N. A., and Choi, Y. Dataset cartography: Mapping and diagnosing datasets with training dynamics. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 9275–9293, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.746. URL https://www.aclweb.org/anthology/2020.emnlp-main.746.

Tatman, R. Gender and dialect bias in YouTube's automatic captions. In *ACL Workshop on Ethics in NLP*, pp. 53–59, Valencia, Spain, April 2017. ACL. doi: 10.18653/v1/W17-1606. URL https://www.aclweb.org/anthology/W17-1606.

Tsuchiya, M. Performance impact caused by hidden bias of training data for recognizing textual entailment. In *LREC*, Miyazaki, Japan, May 2018. European Language Resources Association (ELRA). URL https://www.aclweb.org/anthology/L18-1239.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is all you need. *CoRR*, abs/1706.03762, 2017. URL http://arxiv.org/abs/1706.03762.

Voita, E. and Titov, I. Information-theoretic probing with minimum description length. In *EMNLP*, pp. 183–196, Online, November 2020. ACL. doi: 10.18653/v1/2020.emnlp-main.14. URL https://www.aclweb.org/anthology/2020.emnlp-main.14.

Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., and Bowman, S. R. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *ICLR*, 2019. URL https://openreview.net/forum?id=rJ4km2R5t7.

Warstadt, A., Singh, A., and Bowman, S. R. Neural network acceptability judgments. *TACL*, 7:625–641, March 2019. doi: 10.1162/tacl_a_00290. URL https://www.aclweb.org/anthology/Q19-1040.

Whitney, W. F., Song, M. J., Brandfonbrener, D., Altosaar, J., and Cho, K. Evaluating representations by the complexity of learning low-loss predictors, 2020.

Wiegreffe, S., Marasovic, A., and Smith, N. A. Measuring association between labels and free-text rationales, 2020.

Williams, A., Nangia, N., and Bowman, S. A broad-coverage challenge corpus for sentence understanding through inference. In *NAACL, Volume 1 (Long Papers)*, pp. 1112–1122, New Orleans, Louisiana, June 2018. ACL. doi: 10.18653/v1/N18-1101. URL https://www.aclweb.org/anthology/N18-1101.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Le Scao, T., Gugger, S., Drame, M., Lhoest, Q., and Rush, A. Transformers: State-of-the-art natural language processing. In *EMNLP: System Demonstrations*, pp. 38–45, Online, October 2020. ACL. doi: 10.18653/v1/2020.emnlp-demos.6. URL https://www.aclweb.org/anthology/2020.emnlp-demos.6.

Yang, Z., Qi, P., Zhang, S., Bengio, Y., Cohen, W., Salakhutdinov, R., and Manning, C. D. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *EMNLP*, pp. 2369–2380, Brussels, Belgium, October-November 2018. ACL. doi: 10.18653/v1/D18-1259. URL https://www.aclweb.org/anthology/D18-1259.

Yogatama, D., de Masson d'Autume, C., Connor, J., Kociský, T., Chrzanowski, M., Kong, L., Lazaridou, A., Ling, W., Yu, L., Dyer, C., and Blunsom, P. Learning and evaluating general linguistic intelligence. *CoRR*, abs/1901.11373, 2019. URL http://arxiv.org/abs/1901.11373.

Zhang, P., Goyal, Y., Summers-Stay, D., Batra, D., and Parikh, D. Yin and Yang: Balancing and answering binary visual questions. In *CVPR*, 2016a. URL https://arxiv.org/abs/1511.05099.

Zhang, Y., Marshall, I., and Wallace, B. C. Rationale-augmented convolutional neural networks for text classification. In *EMNLP*, pp. 795–804, Austin, Texas, November 2016b. ACL. doi: 10.18653/v1/D16-1076. URL https://www.aclweb.org/anthology/D16-1076.

Zhao, J., Wang, T., Yatskar, M., Ordonez, V., and Chang, K.-W. Men also like shopping: Reducing gender bias amplification using corpus-level constraints. In *EMNLP*, pp. 2979–2989, Copenhagen, Denmark, September 2017. ACL. doi: 10.18653/v1/D17-1323. URL https://www.aclweb.org/anthology/D17-1323.