

A Task in a Suit and a Tie: Paraphrase Generation with Semantic Augmentation*

Su Wang,¹ Rahul Gupta,² Nancy Chang,² Jason Baldridge²

The University of Texas at Austin¹

Google AI Language²

shrekwang@utexas.edu; {grahul, ncchang, jasonbaldridge}@google.com

Abstract

Paraphrasing is rooted in semantics. We show the effectiveness of transformers (Vaswani et al. 2017) for paraphrase generation and further improvements by incorporating Prop-Bank labels via a multi-encoder. Evaluating on MSCOCO and WikiAnswers, we find that transformers are fast and effective, and that semantic augmentation for both transformers and LSTMs leads to sizable 2-3 point gains in BLEU, METEOR and TER. More importantly, we find surprisingly large gains on human evaluations compared to previous models. Nevertheless, manual inspection of generated paraphrases reveals ample room for improvement: even our best model produces human-acceptable paraphrases for only 28% of captions from the CHIA dataset (Sharma et al. 2018), and it fails spectacularly on sentences from Wikipedia. Overall, these results point to the potential for incorporating semantics in the task while highlighting the need for stronger evaluation.

Introduction

Paraphrasing is at its core a semantic task: restate one phrase s as another s' with approximately the same meaning. High-precision, domain-general paraphrase generation can benefit many natural language processing tasks (Madnani and Dorr 2010). For example, paraphrases can help diversify responses of dialogue assistants (Shah et al. 2018), augment machine translation training data (Fader, Zettlemoyer, and Etzioni 2014) and extend coverage of semantic parsers (Berant and Liang 2014). But what makes a good paraphrase? Whether two phrases have “the same” meaning leaves much room for interpretation. As Table 1 illustrates, paraphrases can involve lexical and syntactic variation with differing degrees of semantic fidelity. Should a paraphrase yield precisely the same inferences as the original? Should it refer to the same entities, in similar ways, with the same detail? How much lexical overlap between paraphrases is desirable?

Much prior work on paraphrasing addresses these questions directly by exploiting linguistic knowledge, including handcrafted rules (McKeown 1983), shallow linguistic features (Zhao et al. 2009) and syntactic and semantic information (Kozlowski, McCoy, and Vijay-Shanker 2003; Ellsworth and Janin 2007). These systems, while difficult to

Source	Sue sold her car to Bob.
A	Sue sold her auto to Bob. Sue sold Bob her car. Bob was sold a car by Sue.
B	Bob bought a vehicle from Sue. Bob bought Sue’s car (from her). Bob paid Sue for her automobile. Sue let Bob buy her auto.
C	Sue gave Bob a good deal on her Honda. Bob got a bargain on Sue’s Honda.

Table 1: Paraphrases of a source sentence with increasing syntactic and semantic variability: (A) mild lexical and syntactic variation; (B) moderate lexical and syntactic variation plus perspective shifting; (C) even more lexical and syntactic variation, with greater semantic license.

scale and typically restricted in domain, nonetheless produce paraphrases that are qualitatively close to what humans expect. For example, Ellsworth and Janin (2007)’s rule-based system uses frame semantic information (Fillmore 1982) to paraphrase *I want your opinion* as *Your opinion is desired*, where the two sentences evoke the DESIRING frame from different perspectives.

Many recent paraphrase generation systems sidestep explicit questions of meaning, focusing instead on implicitly learning representations by training deep neural networks (see next section) on datasets containing paraphrase pairs. This is essentially the same approach as modern sequence-to-sequence machine translation (MT), another intrinsically semantic task for which deep learning on large-scale end-to-end data has yielded large gains in quality. Unlike machine translation, however, the training data currently available is much smaller and more domain-specific for paraphrasing. Despite incremental improvements on automatically scored metrics, the paraphrasing quality of state-of-the-art systems fall far short of what’s needed for most applications.

This work investigates a combination of these two approaches. Can adding structured linguistic information to large-scale deep learning systems improve the quality and generalizability of paraphrase generation while remain-

*Work done while the first author was an intern at Google AI. Copyright © 2019, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

Dataset	Train	Test	Vocab
MSCOCO	331,163	162,023	30,332
WikiAnswers	1,826,492	500,000	86,334

Table 2: Data statistics

ing scalable and efficient?¹ We present a fast-converging and data-efficient paraphrase generator that exploits structured semantic knowledge within the Transformer framework (Vaswani et al. 2017; Xiong et al. 2018). Our system outperforms state-of-the-art systems (Prakash et al. 2016; Gupta et al. 2018) on automatic evaluation metrics and data benchmarks, with improvements of 2-3 BLEU, METEOR and TER points on both MSCOCO and WikiAnswers.

Despite these substantial gains, manual examination of our system’s output on out-of-domain sentences reveals serious problems. Models with high BLEU scores often paraphrased unfamiliar inputs as *a man in a suit and a tie...* (see Table 11). This striking pattern suggests we need more extensive evaluation to better characterize model behavior. We thus obtain human judgments on the outputs of several models to measure both overall quality and relative differences between the models. Interestingly, we find that some models with similar BLEU scores differ widely in human evaluations: e.g. an LSTM-based model with 41.1 BLEU score gets 36.4% acceptability, while a Transformer model with 41.0 BLEU score rates 45.6%. These and our other human evaluations and observations imply that automatic metrics are useful for model development for paraphrase generation research, but not sufficient for final comparison.

Our results also indicate considerable headroom for granting semantics a greater role throughout—in representation, architecture design and task evaluation. This emphasis is in line with other recent and related work, including entailment in paraphrasing (Pavlick et al. 2015), paraphrase identification with Abstract Meaning Representations (Issa et al. 2018) and using semantic role labeling in machine translation (Marcheggiani, Bastings, and Titov 2018). It should be noted that this work is only a first step toward paraphrasing at the quality exemplified in Table 1. We believe that achieving that level requires much more fundamental work to define appropriate tasks, metrics and optimization objectives.

Data and Evaluation

Data. We use MSCOCO (Lin et al. 2014) and WikiAnswers (Fader, Zettlemoyer, and Etzioni 2013) as our main datasets for training and evaluation, as prepared by Prakash et al. (2016). MSCOCO contains 500K+ paraphrases pairs created by crowdsourcing captions for a set of ~120K images (e.g., a horse grazing in a meadow). Each image is captioned by 5 people with a (single/multi-clause) sentence. WikiAnswers has 2.3M+ questions pairs marked by users of the

¹While our models are resource-dependent, we believe it is reasonable to utilize existing resources to introduce semantic structure. As we will show, this approach holds great promise when combined with a general-purpose semantic parser.

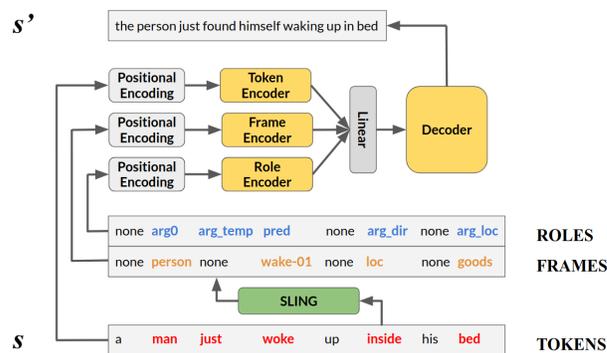


Figure 1: Multi-encoder Transformer (TRANSFORMER-PB, PB denotes PropBank-style semantic annotations) with SLING. Each token is annotated with a frame and role label, resulting in three input channels. Each channel is fed into a separate TRANSFORMER encoder, and results from the three encoders are merged with a linear layer for the decoder.

WikiAnswers website² as similar or duplicate (e.g. *where is Mexico City?* and *in which country is Mexico City located?*). See Table 2 for basic statistics of both datasets.

Evaluation. Paraphrasing systems have been evaluated using purely human methods (Barzilay and McKeown 2001; Bannard and Callison-Burch 2005), human-in-the-loop semi-automatic methods (Callison-Burch, Cohn, and Lapata 2008, ParaMetric), resource-dependent automatic methods (Liu, Dahlmeier, and Ng 2010, PEM), and cost-efficient human evaluation aided by automatic methods (Chaganty, Mussmann, and Liang 2018).

Recent neural paraphrasing systems (Prakash et al. 2016; Gupta et al. 2018) adopt automatic evaluation measures commonly used in MT, citing good correlation with human judgment (Madnani and Tetreault 2010; Wubben, van den Bosch, and Kraehmer 2010): BLEU (Papineni et al. 2002) encourages exact match between source and prediction by n-gram overlap; METEOR (Lavie and Agarwal 2007) also uses WordNet stems and synonyms; and TER (Snover et al. 2006) includes the number of edits between source and prediction.

Such automatic metrics enable fast development cycles, but they are not sufficient for final quality assessment. For example, Chen and Dolan (2011) point out that MT metrics reward homogeneous predictions to the training target, which conflicts with a qualitative goal of good human-level paraphrasing: variation in wording. Chaganty, Mussmann, and Liang (2018) show that predictions receiving low scores from these metrics are not necessarily poor quality according to human evaluation. We thus complement these automated metrics with crowdsourced human evaluation.

Models

Our core contribution is showing a straightforward way to inject semantic frames and roles into the encoders of seq2seq

²wiki.answers.com

models. We also show that Transformers in particular are fast and effective for paraphrase generation.

Transformer. The TRANSFORMER attention-only seq2seq model proposed in Vaswani et al. (2017) has been shown to be a data-efficient, fast-converging architecture. It circumvents token-by-token encoding with a parallelized encoding step that uses token position information. Experimentally its sophisticated self-attention and target-to-source attention mechanism work robustly for long sequences (see the experimentation section). The basic building block of the TRANSFORMER encoder is a multi-head attention layer followed by a feedforward layer, where both have residual links and layer norm (Ba, Kiros, and Hinton 2016):

$$\mathbf{h} = \text{LayerNorm}(\text{FFNN}(\mathbf{m})) + \mathbf{m} \quad (1)$$

$$\mathbf{m} = \text{LayerNorm}(\text{MultiAttn}(\mathbf{x})) + \mathbf{x} \quad (2)$$

$$\mathbf{x} = \text{PositionalEncoding}(s) \quad (3)$$

where \mathbf{h} denotes the encoding output of the block, \mathbf{m} the encoding by the multi-attention layer, and \mathbf{x} the positionally encoded word embeddings of the input sequence. N encoding blocks are cloned to produce the final encoding outputs \mathbf{h}_N , where $\mathbf{h}_n = g_{n-1}(\mathbf{h}_{n-1})$, where g_{n-1} is the previous encoding block. The decoding block is almost identical to the encoding block, with the addition of one more multi-attention layer before the feedforward layer, where the decoder attends to the encoding outputs \mathbf{h}_N . The decoder also has a number of clones of the decoding blocks, which is not necessarily equal to N . The final output of the decoder is projected through a linear layer followed by a softmax.

Encoding semantics. We predict structured semantic representations with an off-the-shelf general-purpose frame-semantic parser SLING (Ringgaard, Gupta, and Pereira 2017). SLING is a neural transition-based semantic graph generator that incrementally generates a frame graph representing the meaning of its input text. SLING’s frame graph representation can homogeneously encode common semantic annotations such as entity annotations, semantic role labels (SRL), and more general frame and inter-frame annotations; the released SLING model³ provides entity, measurement, and PropBank-style SRL annotations out of the box. For example, for the sentence *a man just woke up inside his bed*, the spans *man* and *woke* evoke frames of type *person* and */propbank/wake-01*, respectively, and the latter frame links to the former via the role *arg0*, denoting that the *person* frame is the subject of the predicate. Other spans in the sentence similarly evoke their corresponding frames and link with the appropriate roles (see Figure 1). The end output of SLING is an inter-linked graph of frames.

Given an input sentence s , SLING first embeds and encodes the tokens of s through a BiLSTM using lexical features. A feedforward Transition-Based Recurrent Unit (TBRU) then processes the token vectors one at a time, left to right. At each step, it shifts to the next token or makes an edit operation (or *transition*) to the frame graph under construction (initially empty). Examples of such transitions are (a) evoking a new frame from a span starting at the current

token, (b) linking two existing frames with a role, and (c) revoking an old frame for a new span (e.g. as in coreference resolution). A full set of such operations is listed in Ringgaard, Gupta, and Pereira (2017). To decide which operation to perform, SLING maintains and exploits a bounded priority queue of frames predicted so far, ordered by those most recently used/evoked. Since frame-graph construction boils down to a sequence of fairly general transitions, the model and architecture of SLING are independent of the annotation schema. The same model architecture can thus be trained and applied flexibly to diverse tasks, such as FrameNet parsing (Schneider and Wooters 2018) or coreference resolution.

SLING offers two major benefits for our paraphrase generation task. First, since everything is cast in terms of frames and roles, multiple heterogeneously annotations can be homogeneously obtained in one model using SLING’s frame API. We use to access entity, measurements, and SRL frames output from a pre-trained SLING model. Second, the frame graph representation is powerful enough to capture many semantic tasks, so we can eventually try other ways of capturing semantics by changing the annotation schema, e.g. QA-SRL (He, Lewis, and Zettlemoyer 2015), FrameNet annotations, open-domain facts, or coreference clusters.

Here, we incorporate SLING’s entity, measurement, and PropBank-SRL frames using a multi-encoder method. Taking TRANSFORMER for example (Figure 1), we first use SLING to annotate the input sentence s with frame and role labels, and transfer these labels to tokens, resulting in three aligned vectors for s : tokens, frames and roles. The three channels each have a separate TRANSFORMER encoder. The encoders produce three sets of outputs which are merged with a linear layer before decoding. The multi-encoder TRANSFORMER with the PropBank-style SLING annotations will be listed as TRANSFORMER-PB.

Benchmarks. We consider two state-of-the-art neural paraphrase generators: (1) a stacked residual LSTM by Prakash et al. (2016) (listed as SR-LSTM), and (2) a nested variational LSTM by Gupta et al. (2018) (listed as NV-LSTM).

Let $s = \{w_1, \dots, w_K\}$ be an input sentence, where w_k denotes a word embedding. The SR-LSTM encodes, with a bidirectional LSTM (Bi-LSTM), the sentence as a context vector c that encapsulates its representation. At each time-step at layer l , the LSTM cell takes input from the previous state, as well as a residual linking (He et al. 2015) from the previous layer $l - 1$:

$$\mathbf{c} = \text{Bi-LSTM}(s) \quad (4)$$

$$\mathbf{h}_t^l = \text{LSTM-Cell}^l(\mathbf{h}_t^{l-1}, \mathbf{h}_{t-1}^l) + \mathbf{w}_t \quad (5)$$

where \mathbf{h}_t^l is the hidden state at layer l and time-step t . Prakash et al. (2016) find the best balance between model capacity and generalization with 2-layered stacking. The decoder is initialized with the context vector c and has the same architecture as the encoder. Finally, the decoder maintains a standard attention (Bahdanau, Cho, and Bengio 2015) over the output states of the encoder.

For NV-LSTM, the encoder consists of a sequence of two nested LSTMs: Let s and s' be a paraphrase input entry (source and target). The second nested LSTM takes s' and

³<https://github.com/google/sling>

Model	MSCOCO			Wikianswers			#Tokens/sec (GPU)
	BLEU↑	METEOR↑	TER↓	BLEU↑	METEOR↑	TER↓	
SR-LSTM (Prakash et al. 2016)	36.7	27.3	52.3	37.0	32.2	27.0	-
NV-LSTM (Gupta et al. 2018)	41.7	31.0	40.8	-	-	-	-
SR-LSTM (ours)†	36.5	26.8	51.4	36.9	34.7	27.0	458
NV-LSTM (ours)†	41.1	31.2	41.1	39.2	36.1	22.9	417
TRANSFORMER (Vaswani et al. 2017)	41.0	32.8	40.5	41.9	35.8	22.5	2,875
SR-LSTM-PB (ours)	40.8	32.3	47.0	42.1	37.9	21.2	173
TRANSFORMER-PB (ours)	44.0	34.7	37.1	43.9	38.7	19.4	2,650

Table 3: Results on MSCOCO and WIKIANSWERS with length-15 sentence truncation. The arrows ↑ and ↓ indicate how the scores are interpreted: for BLEU and METEOR, the higher the better, for TER, the lower the better. The best result in a column is in boldface. The #tokens/sec statistics: batch size 32.(†: We emphasize that these results are based on our reimplementations of the models described in the cited papers.)

the encoding of s by the first nested LSTM. This results in the context vector \mathbf{c} :

$$\mathbf{c} = \text{LSTM}_2^{\text{enc}}(s', \text{LSTM}_1^{\text{enc}}(s)) \quad (6)$$

The context vector is then fed through a standard variational reparameterization layer (Kingma and Welling 2014; Bowman et al. 2015) to produce encoding \mathbf{z} :

$$\mathbf{z} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\sigma}) \quad (7)$$

$$\boldsymbol{\mu} = \text{Linear}_{\boldsymbol{\mu}}(\mathbf{h}) \quad (8)$$

$$\boldsymbol{\sigma} = \text{Linear}_{\boldsymbol{\sigma}}(\mathbf{h}) \quad (9)$$

The decoder also comprises two nested LSTMs: the first again encodes s and produces its final state for the second LSTM for final decoding, which also conditions on \mathbf{z} :

$$\hat{s}' = \text{LSTM}_2^{\text{dec}}(\text{LSTM}_1^{\text{dec}}(s) | \mathbf{z}) \quad (10)$$

where \hat{s}' is the predicted sequence.

To separate the contributions from architecture and semantics, we also implement a similar multi-encoder for the baseline SR-LSTM (listed as SR-LSTM-PB), where we again work with three copies of encoder Bi-LSTMs:

$$\mathbf{c} = \text{Linear}(\text{Concat}(\mathbf{c}_s, \mathbf{c}_f, \mathbf{c}_r)) \quad (11)$$

$$\mathbf{c}_s = \text{Bi-LSTM}(s) \quad (12)$$

$$\mathbf{c}_f = \text{Bi-LSTM}(f) \quad (13)$$

$$\mathbf{c}_r = \text{Bi-LSTM}(r) \quad (14)$$

$$f, r = \text{SLING}(s) \quad (15)$$

where f and r are the same frame and role annotations from SLING as used in TRANSFORMER-PB; and $\mathbf{c}_s, \mathbf{c}_f, \mathbf{c}_r$ are the context vectors produced through the token, frame, and role channels, respectively.

Automatic Evaluation

We first use easily computed automatic metrics to evaluate Transformers and semantic augmentation of both Transformers and SR-LSTM. We train and assess the models on MSCOCO and WikiAnswers with the same setup as in previous work. In particular, for a fair and normalized comparison, we control factors such as computational machinery (including hardware and software), random seeding,

Model	MSCOCO	WikiAnswers
SR-LSTM (ours)	11.6	11.3
NV-LSTM (ours)	10.9	12.1
TRANSFORMER	17.7	16.8
SR-LSTM-PB (ours)	11.5	11.6
TRANSFORMER-PB (ours)	17.9	16.4

Table 4: BLEU scores on long sentences (>20 words). Similar patterns hold for METEOR and TER.

hyperparameter tuning, etc. by implementing all the models in the same toolkits.⁴ Specifically, we build all models with PyTorch 0.3.1 (hyperparameters tuned to their best performance with the same random seeds) and run all experiments on a Titan Xp GPU. All models are tuned to their best performing configuration, and the average over 20 runs is recorded. For all models we apply greedy decoding.⁵ All word embeddings are initialized with 300D GloVe vectors (Pennington, Socher, and Manning 2014). For SR-LSTM, NV-LSTM and TRANSFORMER respectively: the learning rates are 1e-4/1e-5/1e-5; the dropout rates are 0.5/0.3/0.1. For the TRANSFORMER, we set the weight decay factor at 0.99, with the warmup at 500.

For the first experiment, following Prakash et al. (2016), we restrict complexity by truncating all sentences to 15 words. We also gauge model speed as *tokens per second*. As shown in Table 3, TRANSFORMER performs comparably with the current state-of-the-art NV-LSTM—and is nearly 7x faster with GPU. Also, while SR-LSTM generally underperforms NV-LSTM, our version enhanced with SLING-parsed semantic information (SR-LSTM-PB) pulls ahead in all metrics except speed. This slowdown relative to other models is expected, since the multi-encoder increases model size. But TRANSFORMER-PB parallelizes the added computations while retaining the relative performance gain, scoring the highest on all metrics. This experiment sets a new state-of-

⁴Crane (2018) argues convincingly that factors such as hardware (e.g. GPU type and configuration), software (framework and version) and random seeding have nontrivial impacts on results.

⁵Rather than beam search, which shows little performance variation from greedy search, as observed in our experiments and reported in the original works (Prakash et al. 2016; Gupta et al. 2018).

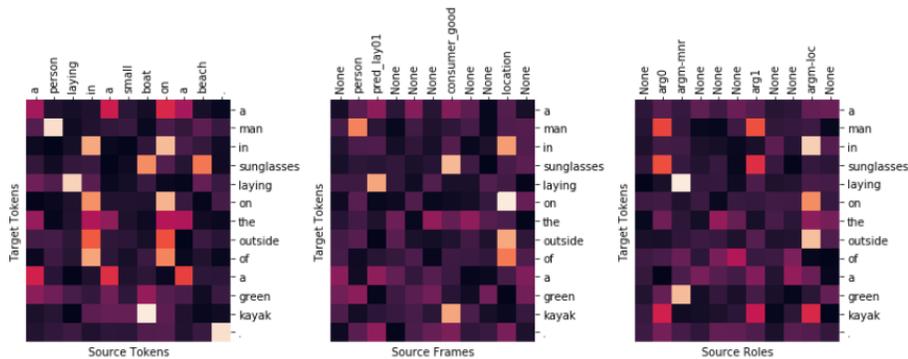


Figure 2: Example for target→source attention (tokens, frames and roles). Each role is an attention weight distribution over source tokens/frames/roles (normalized). Attention weights taken from the first layer of the TRANSFORMER-PB.

Model	MSCOCO Train Size			
	50K	100K	200K	331K (original)
SR-LSTM	9.7	18.3	29.3	36.5
NV-LSTM	5.3	15.4	25.5	41.1
TRANSFORMER	9.9	24.5	33.9	41.0

Table 5: Experimenting with lower data sizes: performance (BLEU) of three models with randomly sampled training data (MSCOCO) in three sizes. Note that this evaluation is for all examples, with no length restrictions.

Data	None	Frame-only	Role-only	Both
MSCOCO	41.0	42.5	41.8	44.0
WikiAnswers	41.9	43.2	43.0	43.9

Table 6: Ablation study (BLEU) with the best model TRANSFORMER-PB. Note that this evaluation is for all examples, with no length restrictions.

the-art for these benchmarks with about 6-7x speed-up.

We also measured BLEU scores for models trained on all sentences but evaluated only on sentences with >20 words (Table 4). The Transformer models prove especially effective for longer sentences, with a 4-6 BLEU margin over the LSTM models. Adding semantics did not provide further benefit on these metrics. This is likely in part because the quality of SLING’s SRL annotations goes down on longer sentences: SLING’s SPAN and ROLE F1 scores were respectively 1.5% and 9.8% lower on long sentences. Though SLING’s lower performance could be due partly to its in-built one-pass greedy decoder, long-range dependencies pose a general problem for all semantic role labeling systems. They employ mechanisms ranging from tweaking long sentences (Vickrey and Koller 2008) to incorporating syntax (Strubell et al. 2018, Sec. A.1). Exploring such approaches is a possible avenue for future work.

As mentioned earlier, a key limitation of the seq2seq approach for paraphrasing is that it has orders of magnitude less data than typically available for training MT systems using the same model architecture. To see how train-

ing set size impacts performance, we trained and evaluated the three base systems on three randomly selected subsets of MSCOCO with 50K, 100K, and 200K examples each, plus using all 331K examples. Results in Table 5 show that successively doubling the number of examples gives large improvements in BLEU. The TRANSFORMER model is particularly able to exploit additional data, attesting to its overall representational capacity given sufficient training examples. Note also that using all 331k examples in MSCOCO produces a BLEU score of 41.0—a strong indication that more data will likely further improve model quality.

Table 6 shows the results of ablating semantic information in the TRANSFORMER model, with four conditions: None (i.e. TRANSFORMER), Frame-only, Role-only and Both (i.e. TRANSFORMER-PB). Frame and role information each improve results by 1-2 points on their own, but combining them yields the largest overall gain for both datasets.

We can also examine the TRANSFORMER-PB’s attention alignments for indirect evidence of the contribution of semantics. Figure 2 shows a cherry-picked example of target-to-source attention weights in the first layer of the multi-encoder.⁶ For token-to-token attention, words are aligned, more or less as expected, with distributional lexical similarity (Luong, Pham, and Manning 2016, for instance). For token-to-frame and token-to-role alignment, word tokens attend to frames and roles with interpretable relations. For example, *man* has heavy attention weights for the frame *person* and the roles *arg0* and *arg1*; *sunglasses* for the frame *consumer_good*; *laying* for *argm-mnr* (manner).

Human Evaluation

While adding semantic information significantly boosts performance on automated evaluation metrics, it is unclear whether the gains actually translate into higher-quality paraphrases. As noted above, MT metrics like BLEU reward fidelity to the input phrase, unlike human judgments that value lexical and syntactic diversity. This systematic divergence means that reasonable paraphrases can receive low BLEU scores (Table 7); and conversely that high BLEU scores can

⁶We did not observe clear patterns for other layers.

Type	Target	Prediction
Low-BLEU/Good	a man in sunglasses laying on a green kayak. a girl in a jacket and boots with a black umbrella.	the man laying on a boat in the water. a little girl holding a umbrella.
Low-BLEU/Bad	people on a gold course enjoy a few games a tall building and people walking	a group of people walking. a large building with a clock on the top
High-BLEU	a picture of someone taking a picture of herself. a batter swinging a bat at a baseball.	a woman taking a picture with a cell phone. a baseball player swinging a bat at a ball.

Table 7: Examples from sample study. Pairs with Low-BLEU score <10 in BLEU, and those with High-BLEU score >25 . In general, high BLEU correlates with good paraphrasing, but low BLEU includes both relatively good and bad paraphrasing.

Target	Low-BLEU paraphrases	High-BLEU paraphrases
several surfers are heading out into the waves.	some guys are running towards the ocean. [BLEU = 6.0]	some surfers heading into the waves. [BLEU = 34.2]
a kitchen with wooden cabinets and tile flooring	wood closet in a kitchen with tile on floor [BLEU = 17.7]	a kitchen with cabinets and flooring [BLEU = 27.0]

Table 8: Examples illustrating disconnect between standard BLEU metric and human intuition: low-scoring paraphrases are more diverse than high-scoring ones, which tend toward parroting.

Model	Task 1 (source,pred)	Task 2 (source,target,pred)
NV-LSTM	19.0	36.4
TRANSFORMER	31.1	45.6
TRANSFORMER-PB	36.4	66.5

Table 9: Human acceptability scores for models presented with (target, prediction); and (target, source, prediction).

TRANSFORMER	TRANSFORMER-PB	CHIA
18.0	28.0	78.2

Table 10: Similar-domain comparison on 1000 sampled CHIA images by 3 human evaluators, paired with predictions from the Transformer with and without semantic augmentation and with the gold CHIA captions.

be obtained by parroting of the input sentence (Table 8).

A related question is: how can we be sure our models can generalize to other datasets and domains? MSCOCO’s caption-based data grants considerable semantic license since the “paraphrase” caption can include more or less or different detail than the original. Paraphrases involve several kinds of variation, and overall paraphrase acceptability is a matter of degree. Are our models learning robust generalizations about what kinds of variation preserve meaning?

To investigate these issues, we turned to crowdsourced human judgments. We ran several experiments that varied in dataset used (same domain, similar domain, out-of-domain) and paraphrase decision (roughly, what kinds of inputs are presented). All cases require a binary judgment, averaged over multiple human evaluators. Each is described below.

Task 1. Same domain, standard paraphrase: Decide whether s_1 and s_2 have approximately the same meaning.

This task compares NV-LSTM (the previous state-of-the-art) with our TRANSFORMER and TRANSFORMER-PB mod-

els on the standard paraphrase task. We randomly sampled 100 sentences from MSCOCO, paired them with their predicted paraphrases from each model, and presented each to 5 human evaluators. Results in Table 9(a) demonstrate that our models significantly outperformed the state-of-the-art, with the TRANSFORMER-PB hitting 36.4% paraphrase acceptability, compared to NV-LSTM’s 19.0%.

Given the relatively low bar, however, it is worth considering how well the task as presented fits the data. As noted earlier, MSCOCO “paraphrases” are based on multiple descriptions of the same image; it thus includes many pairs that differ significantly in details or focus. For example, *a boy standing on a sidewalk by a skateboard* and *there is a young boy that is standing between some trees* differ in details to raise doubt about whether they mean “the same thing”—but they are nonetheless compatible descriptions of the same scene. A judgment better matched to the annotated data could provide more insight into model performance.

Task 2. Same domain, pair-based paraphrase: Assuming s_1 and s_2 have approximately the same meaning, decide whether s_3 also has approximately the same meaning.

This task adapts the standard paraphrase judgment to better handle the inherent variability of caption-based training data. Rather than explicitly define what is meant by “approximately the same meaning”, we give raters additional context by providing both source and target (i.e. a gold training pair) as input to the task. This approach flexibly accommodates a wide range of differences between source and target, but (implicitly) requires them to be mutually consistent.

As shown in Table 9(b), all systems score much higher for this evaluation over standard paraphrase task. Interestingly, we see even greater gains of TRANSFORMER-PB (66.5%) over TRANSFORMER (45.6%) for this task.

Task 3. Similar domain, image-based paraphrase: Given image i and its caption s_1 , decide whether a paraphrase s_2 generated from s_1 is an acceptable caption for i .

To address the key issue of domain generalization, we

1-O: The figure is illuminated by four footlights in the base and its proper left arm is raised.

1-T: Baseball players in a field playing baseball on a field.

2-O: One day, while going to bed, Tara tells Ramesh that the paw works as she'd wished to solve the problem.

2-T: Photograph of a man in a suit and tie with a hat on his head.

3-O: According to free people, democracy will not work without the element of the sensitive, conscience of an individual.

3-T: Photograph of a man in a suit and tie looking at a cell phone.

Table 11: Paraphrase predictions on out-of-domain (Wikipedia) sentences (O=Original, T=Transformer), demonstrating (1) hallucination of a baseball scene; and (2 and 3) bias toward predicting *photograph of a man in a suit and tie* for novel input. Note that n-grams in the bias sequence appear frequently in the corpus (e.g. *in a suit and tie* ranks 345 out of over 21M unique 5-grams), accounting for the failure mode of “backing off” to high-frequency sequences when testing on out-of-domain input.

evaluated model performance on a very near neighbor: CHIA, a collection of 3.3M image caption pairs of a specific caption with a more general (paraphrase) caption (Sharma et al. 2018).⁷ We sampled 1,000 CHIA images and presented image-caption pairs to 3 human raters, where captions come from our TRANSFORMER, TRANSFORMER-PB models and the gold CHIA paraphrase.

Table 10 shows a large gain for the semantically enhanced TRANSFORMER-PB (28.0%) over the basic TRANSFORMER (18.0%)—but both fall far short of CHIA (78.2%).

Out-of-domain, standard paraphrase. The situation grows more dire as we venture further out of domain to tackle Wikipedia data. Our models fail so miserably that we did not bother with human evaluation, with most predictions hallucinatory at best (Table 11). A striking behavior of the basic TRANSFORMER model was that it frequently resorts to mentioning men in suits and ties when presented with novel inputs (sentences 2 and 3).

There are some qualitative indications that the TRANSFORMER-PB makes somewhat more thematically relevant predictions. For example:

- ORIGINAL: *In this fire, he lost many of his old notes and books including a series on Indian birds that he had since the age of 3*
- TRANSFORMER: *bird sitting on a table with a laptop computer on it 's side of the road and a sign on the wall*
- TRANSFORMER-PB: *yellow fire hydrant near a box of birds on a table with books on it nearby table.*

However, neither model’s attempt is even close to adequate.

Taken as a whole, human evaluations (both quantitative and qualitative) provide a more nuanced picture of how our models perform. Across all tasks, we find consistent improvements for models that include semantic information. But none of the models demonstrates much ability to generalize to out-of-domain examples.

Conclusion

We have shown substantial improvements in paraphrase generation through a simple semantic augmentation strategy

⁷CHIA captions are automatically modified versions of the original alt-text associated with the images, e.g. *Musician Justin Timberlake performs at the 2017 Pilgrimage Music & Cultural Festival on September 23, 2017 in Franklin, Tennessee* is generalized to *pop artist performs at the festival in a city* as the CHIA caption.

that works equally well with both LSTMs and Transformers. Much scope for future work remains. Since SLING is a neural model, it associates both semantic frames and roles with continuous vectors, and the parser state upon adding frames/roles to the graph can serve as a continuous internal representation of the input’s incrementally constructed meaning. In future work we plan to integrate this continuous representation with the symbolic (final) frame graph used here, as well as train SLING for other tasks.

Our evaluations also suggest a fundamental mismatch between automated metrics borrowed from MT, which are optimized for surface similarity, and the more nuanced factors that affect human judgment of paraphrase quality. Automated metrics, while far more scalable in speed and cost, cannot (yet) substitute for human evaluation. Paraphrase generation research would benefit from better use of existing datasets (Iyer, Dandekar, and Csernai 2017; Xu, Callison-Burch, and Dolan; Lan et al. 2017), creation of new datasets, and evaluation of metrics similar to that done by Toutanova et al. (2016) for abstractive compression.

In particular, crowdsourcing tasks that provide more evidence about what is meant by “meaning”—such as additional input instances, or the source image—may better accommodate diverse sources of paraphrase data. Future work can explore how robustly models perform given different amounts and types of data from a variety of tasks.

References

- Ba, J. L.; Kiros, J. R.; and Hinton, G. E. 2016. Layer Normalization. In *CoRR*.
- Bahdanau, D.; Cho, K.; and Bengio, Y. 2015. Neural Machine Translation by Jointly Learning to Align and Translate. In *Proceedings of ICLR*.
- Bannard, C., and Callison-Burch, C. 2005. Paraphrasing with bilingual parallel corpora. In *Proceedings of ACL*.
- Barzilay, R., and McKeown, K. 2001. Extracting paraphrases from a parallel corpus. In *Proceedings of ACL*.
- Berant, J., and Liang, P. 2014. Semantic parsing via paraphrasing.
- Bowman, S. R.; Vilnis, L.; Vinyals, O.; Dai, A. M.; Jozefowicz, R.; and Bengio, S. 2015. Generating Sentences from a Continuous Space. In *CoRR*.
- Callison-Burch, C.; Cohn, T.; and Lapata, M. 2008. Parametric: An automatic evaluation metric for paraphrasing. In *Proceedings of COLING*.
- Chaganty, A. T.; Mussmann, S.; and Liang, P. 2018. The Price of

- Debiasing Automatic Metrics in Natural Language Processing. In *Proceedings of ACL*.
- Chen, D. L., and Dolan, W. B. 2011. Collecting Highly Parallel Data for Paraphrase Evaluation. In *Proceedings of ACL*.
- Crane, M. 2018. Questionable Answers in Question Answering Research: Reproducibility and Variability of Published Results. In *Proceedings of TACL*.
- Ellsworth, M., and Janin, A. 2007. Mutaphrase: Paraphrase with FrameNet. In *Proceedings of ACL*.
- Fader, A.; Zettlemoyer, L.; and Etzioni, O. 2013. Paraphrase-Driven Learning for Open Question Answering. In *Proceedings of ACL*.
- Fader, A.; Zettlemoyer, L.; and Etzioni, O. 2014. Open Question Answering Over Curated and Extracted Knowledge Bases. In *Proceedings of KDD*.
- Fillmore, C. J. 1982. *Frame semantics*. Seoul, South Korea: Hanshin Publishing Co. 111–137.
- Gupta, A.; Agarwal, A.; Singh, P.; and Rai, P. 2018. A Deep Generative Framework for Paraphrase Generation. In *Proceedings of AAAI*.
- He, K.; Zhang, X.; Ren, S.; and Sun, J. 2015. Deep Residual Learning for Image Recognition. In *CoRR*.
- He, L.; Lewis, M.; and Zettlemoyer, L. S. 2015. Question-answer driven semantic role labeling: Using natural language to annotate natural language. In *EMNLP*.
- Issa, F.; Damonte, M.; Cohen, S. B.; Yan, X.; and Chang, Y. 2018. Abstract meaning representation for paraphrase detection. In *Proceedings of NAACL-HLT18*, 486–492.
- Iyer, S.; Dandekar, N.; and Csernai, K. 2017. First Quora Dataset Release: Question Pairs. In data.quora.com.
- Kingma, D. P., and Welling, M. 2014. Auto-encoding Variational Bayes. In *Proceedings of ICLR*.
- Kozłowski, R.; McCoy, K. F.; and Vijay-Shanker, K. 2003. Generation of Single-sentence Paraphrases from Predicate/Argument Structure Using Lexico-Grammatical Resources. In *Proceedings of IWP*.
- Lan, W.; Qiu, S.; He, H.; and Xu, W. 2017. A continuously growing dataset of sentential paraphrases. In *Proceedings of The 2017 Conference on Empirical Methods on Natural Language Processing (EMNLP)*.
- Lavie, A., and Agarwal, A. 2007. Meteor: An Automatic Metric for MT Evaluation with High Levels of Correlation with Human Judgments. In *Proceedings of SMT*.
- Lin, T.-Y.; Maire, M.; Belongie, S.; Bourdev, L.; Girshick, R.; Hays, J.; Perona, P.; Ramanan, D.; Zitnick, C. L.; and Dollár, P. 2014. Microsoft COCO: Common Objects in Context. In *CoRR*.
- Liu, C.; Dahlmeier, D.; and Ng, H. T. 2010. PEM: A paraphrase evaluation metric exploiting parallel texts. In *Proceedings of EMNLP*.
- Luong, M.-T.; Pham, H.; and Manning, C. D. 2016. Effective Approaches to Attention-based Neural Machine Translation. In *Proceedings of EMNLP*.
- Madnani, N., and Dorr, B. J. 2010. Generating Phrasal and Sentential Paraphrases: A Survey of Data-driven Methods. In *Computational Linguistics*, 36(3).
- Madnani, N., and Tetreault, J. 2010. Re-examining Machine Translation Metrics for Paraphrase Identification. In *Proceedings of NAACL*.
- Marcheggiani, D.; Bastings, J.; and Titov, I. 2018. Exploiting semantics in neural machine translation with graph convolutional networks. In *Proceedings of NAACL-HLT18*, 486–492.
- McKeown, K. R. 1983. Paraphrasing Questions Using Given and New Information. In *American Journal of Computational Linguistics*.
- Papineni, K.; Roukos, S.; Ward, T.; and Zhu, W.-J. 2002. BLEU: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of ACL*.
- Pavlick, E.; Bos, J.; Nissim, M.; Beller, C.; Van Durme, B.; and Callison-Burch, C. 2015. Adding semantics to data-driven paraphrasing.
- Pennington, J.; Socher, R.; and Manning, C. D. 2014. GloVe: Global Vectors for Word Representation. In *Proceedings of EMNLP*.
- Prakash, A.; ; Hasan, S. A.; Lee, K.; Datla, V.; Qadir, A.; Liu, J.; and Farri, O. 2016. Neural Paraphrase Generation with Stacked Residual LSTM Networks. In *Proceedings of COLING*.
- Ringgaard, M.; Gupta, R.; and Pereira, F. C. N. 2017. SLING: A Framework for Frame Semantic Parsing. In *CoRR*.
- Schneider, N., and Wooters, C. 2018. The NLTK FrameNet API. In *Proceedings of NAACL*.
- Shah, P.; Hakkani-Tur, D.; Tür, G.; Rastogi, A.; Bapna, A.; Nayak, N.; and Heck, L. 2018. Building a Conversational Agent Overnight with Dialogue Self-Play. In *CoRR*.
- Sharma, P.; Ding, N.; Goodman, S.; and Soricut, R. 2018. Conceptual Captions: A Cleaned, Hypernymed, Image Alt-text Dataset For Automatic Image Captioning. In *Proceedings of ACL*.
- Snover, M.; Dorr, B.; Schwartz, R.; Micciulla, L.; and Makhoul, J. 2006. A Study of Translation Edit Rate with Targeted Human Annotation. In *Proceedings of AMT*.
- Strubell, E.; Verga, P.; Andor, D.; Weiss, D.; and McCallum, A. 2018. Linguistically-informed self-attention for semantic role labeling. In *Proceedings of EMNLP*.
- Toutanova, K.; Brockett, C.; Tran, K. M.; and Amershi, S. 2016. A dataset and evaluation metrics for abstractive compression of sentences and short paragraphs. In *Proceedings of EMNLP*.
- Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention is All You Need. In *Proceedings of NIPS*.
- Vickrey, D., and Koller, D. 2008. Sentence simplification for semantic role labeling. In *Proceedings of ACL*.
- Wubben, S.; van den Bosch, A.; and Kraehmer, E. 2010. Paraphrase Generation as Monolingual Translation: Data and Evaluation. In *Proceedings of INLG*.
- Xiong, H.; He, Z.; Hu, X.; and Wu, H. 2018. Multi-channel Encoder for Neural Machine Translation. In *Proceedings of AAAI*.
- Xu, W.; Callison-Burch, C.; and Dolan, B. Semeval-2015 task 1: Paraphrase and semantic similarity in twitter (pit). In *Proceedings of SemEval 2015*.
- Zhao, S.; Lan, X.; Liu, T.; and Li, S. 2009. Application-driven Statistical Paraphrase Generation. In *Proceedings of ACL-IJCNLP*.