Quality Controlled Paraphrase Generation

Elron Bandel[1,2] **Ranit Aharonov**[1] **Michal Shmueli-Scheuer**[1]
Ilya Shnayderman[1] **Noam Slonim**[1] **Liat Ein-Dor**[1]
[1]IBM Research
[2]Computer Science Department, Bar Ilan University
elron.bandel@ibm.com
{shmueli,ilyashn,noams,liate}@il.ibm.com

Abstract

Paraphrase generation has been widely used in various downstream tasks. Most tasks benefit mainly from high quality paraphrases, namely those that are semantically similar to, yet linguistically diverse from, the original sentence. Generating high-quality paraphrases is challenging as it becomes increasingly hard to preserve meaning as linguistic diversity increases. Recent works achieve nice results by controlling specific aspects of the paraphrase, such as its syntactic tree. However, they do not allow to directly control the quality of the generated paraphrase, and suffer from low flexibility and scalability. Here we propose QCPG, a quality-guided controlled paraphrase generation model, that allows directly controlling the quality dimensions. Furthermore, we suggest a method that given a sentence, identifies points in the quality control space that are expected to yield optimal generated paraphrases. We show that our method is able to generate paraphrases which maintain the original meaning while achieving higher diversity than the uncontrolled baseline. The models, the code, and the data can be found in https://github.com/IBM/quality-controlled-paraphrase-generation.

1 Introduction

Paraphrase generation, namely rewriting a sentence using different words and/or syntax while preserving its meaning (Bhagat and Hovy, 2013), is an important technique in natural language processing, that has been widely used in various downstream tasks including question answering (Fader et al., 2014a; McCann et al., 2018), summarization (Rush et al., 2015), data augmentation (Yu et al., 2018) and adversarial learning (Iyyer et al., 2018). However, not all paraphrases are equally useful. For most real-world applications, paraphrases which are too similar to the original sentence are of limited value, while those with high linguistic diversity, i.e. with large syntactic/lexical differences between the paraphrase and the original sentence, are more beneficial to the robustness and accuracy of automatic text evaluation and classification, and can avoid the blandness caused by repetitive patterns (Qian et al., 2019). The quality of paraphrases is often evaluated using three dimensions, where high quality paraphrases are those with high semantic similarity as well as high lexical and/or syntactic diversity (McCarthy et al., 2009).

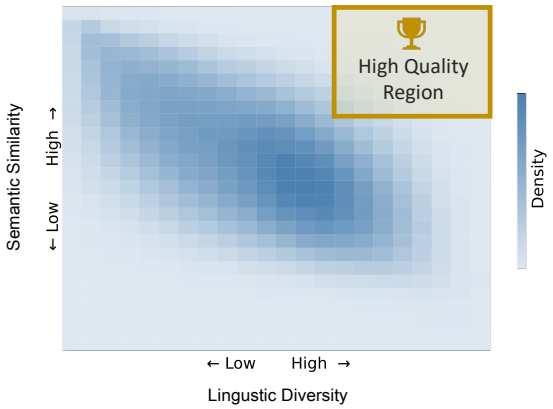

Figure 1: Density of paraphrases in WikiAnswers as a function of the semantic similarity and the linguistic diversity. The marked area, which contains high quality paraphrases, is very sparse (The measures used in the figure are described in Section 2.1) .

Generating high quality paraphrases can be challenging (for both humans and automatic models) since it is increasingly difficult to preserve meaning with increasing linguistic diversity. Indeed, when examining the quality of paraphrases among paraphrase generation datasets, one can find a wide range of paraphrase qualities, where the area of high quality is often very sparse (see Figure 1). This in turn results in scarcity of supervised data for high-quality paraphrase generation.

Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 596 - 609

A recent approach aiming to produce high quality paraphrases is controlled paraphrase generation, which exposes control mechanisms that can be manipulated to produce diversity. While the controlled generation approaches have yielded impressive results, they require providing the model with very specific information regarding the target sentence, such as its parse tree (Iyyer et al., 2018) or the list of keywords it needs to contain (Zeng et al., 2019). However, for most downstream applications, the important property of the paraphrase is its overall quality, rather than its specific syntactic or lexical form. The over-specificity of existing control-based methods not only complicates their usage and limits their scalability, but also hinders their coverage. Thus, it would be desirable to develop a paraphrase generation model, which uses a simple mechanism for directly controlling paraphrase quality, while avoiding unnecessary complications associated with fine-grained controls.

In this paper we propose QCPG, a Quality Controlled Paraphrase Generation model, that given an input sentence and quality constraints, represented by a three dimensional vector of semantic similarity, and syntactic and lexical distances, produces a target sentence that conforms to the quality constraints.

Our constraints are much simpler than previously suggested ones, such as parse trees or keyword lists, and leave the model the freedom to choose how to attain the desired quality levels.

Enabling the direct control of the three quality dimensions, allows flexibility with respect to the specific requirements of the task at hand, and opens a range of generation possibilities: paraphrases of various flavors (e.g. syntactically vs. lexically diverse), quasi-paraphrases (with lower semantic similarity), and even non-paraphrases which may be useful for downstream tasks (e.g. hard negative examples of sentences that are linguistically similar but have different meanings (Guo et al., 2018; Reimers and Gurevych, 2020)).

Our results show that the QCPG model indeed enables controlling paraphrase quality along the three quality dimensions.

Furthermore, even though the training data is of mixed quality, and exhibits scarcity in the high quality area (see Figure 1), our model is able to learn high quality paraphrasing behavior, i.e. it increases the linguistic diversity of the generated paraphrases without decreasing the semantic similarity compared to the uncontrolled baseline.

2 Method

In this section we provide a general description of our approach. We first explain how the different quality dimensions are measured. We then describe the controlled paraphrase generation model, QCPG, and finally we suggest a method that given the task requirements, detects the input control values which maximize the quality of the generated paraphrases. Figure 2 summarizes our proposed solution for generating controlled paraphrases, which is detailed in the rest of the section.

2.1 Quantifying Paraphrase Quality

The most common dimensions for measuring paraphrase quality are the semantic, syntactic and lexical dimensions. Several previous works used also a fluency evaluation metric (Siddique et al., 2020). However, since our focus is on the supervised setting, we rely on the gold paraphrases as fluency guidance for the model (McCarthy et al., 2009). Thus, given a sentence s and a paraphrase s', we define the paraphrase quality as a three dimensional vector $\mathbf{q}(s, s') = (q_{sem}(s, s'), q_{syn}(s, s'), q_{lex}(s, s'))$, where q_{sem} is a measure of semantic similarity, and q_{syn} and q_{lex} are measures of syntactic and lexical variation, respectively. For the syntactic score, inspired by Iyyer et al. (2018) we choose $q_{syn}(s, s')$ to be the normalized tree edit distance (Zhang and Shasha, 1989) between the third level constituency parse-trees of s and s', after removing the tokens - to increase the decoupling from the lexical distance metric. We define the lexical score $q_{lex}(s, s')$ to be the normalized character-level minimal edit distance between the bag of words. This measure is independent of word order, and hence increases the decoupling from syntactic measures. Additionally, calculating the token distances on the character level enables to capture tokens that share the same stem/lemma. Character-level distance is also more robust to typos that may be found in noisy data. As for the semantic score, several strong metrics have been recently proposed for measuring semantic similarity between sentences. In order to select $q_{sem}(s, s')$, we studied the agreement between the candidate metrics and human judgments, using only development data, and found Bleurt (Sellam et al., 2020) to have the highest correlation with human judgments (see Appendix A). Thus, we define

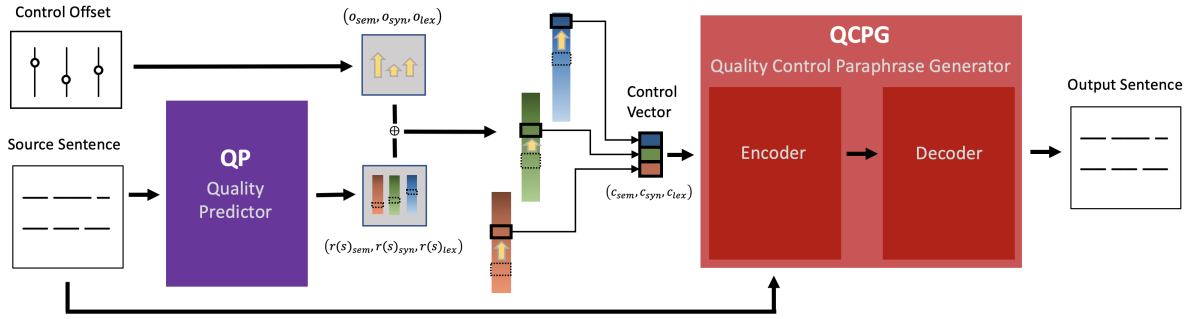

Figure 2: Solution Architecture. The input to the paraphrase generation model, QCPG, is composed of two elements: a sentence s, and a three-dimensional quality vector $c = (c_{sem}, c_{syn}, c_{lex})$, which controls the quality of the generated paraphrase. Selecting appropriate values of c is crucial for obtaining high-quality paraphrases. The quality predictor model, QP, helps select suitable input quality vectors, by predicting the typical quality, $r(s)$, of the paraphrases of s. The control vector c is the sum of $r(s)$, and an offset vector o, which indicates the extent to which the requested quality deviates from the typical value. Dev-set results can help the user in selecting suitable values of o, as shown in Figure 5

$q_{sem}(s, s')$ to be the Bleurt score, normalized using the sigmoid function to ensure a uniform range of values, $[0, 1]$, for all three quality dimensions. For ease of presentation all metrics are presented on a $0 - 100$ scale.

2.2 The QCPG Model

The main component of our solution is a quality controlled paraphrase generation model (QCPG), which is an encoder-decoder model trained on the task of controlled paraphrase generation. Given an input sentence s and a control vector $\mathbf{c} = (c_{sem}, c_{syn}, c_{lex})$, the goal of QCPG is to generate an output paraphrase $QCPG(s, \mathbf{c})$ that conforms to $\mathbf{c}$. We train QCPG using the training set pairs (s, t), by setting $\mathbf{c}$ to be $\mathbf{q}(s, t)$, and maximizing $P(t|s, \mathbf{c} = \mathbf{q}(s, t))$ over the training set via the autoregressive cross entropy loss.

2.3 Control Values Selection

A major challenge in the research of controlled paraphrase generation, is selecting appropriate input control values that can be achieved by the model (Goyal and Durrett, 2020). Clearly, given a sentence, not all paraphrase qualities are achievable. Some sentences are more amenable to paraphrasing than others. For example, named entities and numbers are much harder to be replaced while keeping sentence meaning, and hence, the potential lexical diversity of paraphrases involving such terms is relatively limited. Forcing QCPG to conform to quality control values that are too high with respect to the input sentence, may lead to suboptimal quality of the resultant paraphrases. Thus, for a more effective use of QCPG, the control values should be determined with respect to the input sentence.

Below we describe the second part of our solution, namely a method that given a sentence, predicts the input control values, $\mathbf{c}(s)$, that optimize the expected quality of the paraphrases generated by QCPG. For simplicity we assume that the quality distribution $p(\mathbf{q}|s)$ of all paraphrases of sentence s, is approximately normally distributed around a sentence dependent mean $\mathbf{q_0}(s)$, and that the variance is approximately sentence-independent. We further assume that given an input sentence s, the difficulty to generate a paraphrase of a given quality, $\mathbf{q}$, is dominated by $p(\mathbf{q}|s)$ rather than by the quality vector $\mathbf{q}$ itself.

Following our assumptions, the level of difficulty can be expressed by the offset, $\mathbf{o} = (o_{sem}, o_{syn}, o_{lex})$ of $\mathbf{q}$ from $\mathbf{q_0}(s)$. Thus, the input control, $\mathbf{c}(s)$, for QCPG, is the sum of $\mathbf{q_0}(s)$ and an offset $\mathbf{o}$.

Our aim is to analyze the model results for varying levels of difficulty, namely under different offsets, $\mathbf{o}$, from $\mathbf{q_0}(s)$.

The Quality Predictor (QP): Since $\mathbf{q_0}(s)$ is unknown, we introduce QP, a regressor whose output, termed the reference of s, $\mathbf{r}(s) = (r_{sem}(s), r_{syn}(s), r_{lex}(s))$, approximates $\mathbf{q_0}(s)$. During training, QP aims to predict $\mathbf{q}(s, t)$ given s, where (s, t) are the input-output pairs of the training data.

To summarize, we define *sentence-aware quality control* by decomposing the QCPG input control, $\mathbf{c}$, into a sum of a sentence dependent reference

point, $\mathbf{r}(s)$, and a sentence independent offset, $\mathbf{o}$.

3 Data and Implementation Details

3.1 Datasets

To test the ability of our model to learn high quality behavior from mixed quality data we use weakly annotated datasets. These datasets are large but noisy, and contain only a relatively small amount of high quality paraphrases.

MSCOCO: This dataset consists of 123K images, where each image contains at most five human-labeled captions (Lin et al., 2014). Similar to previous works we consider different captions of the same image as paraphrases.

WikiAnswers (WikiAns for short): The WikiAnswers corpus contains clusters of questions tagged by wiki-answers.com users as similar. There are $30,370,994$ clusters with 25 question in each on average. In total, the corpus contains over 70 million question pairs (Fader et al., 2014b).

ParaBank2.0: A dataset containing clusters of sentential paraphrases, produced from a bilingual corpus using negative constraints, inference sampling, and clustering (Hu et al., 2019). The dataset is composed of avarage of 5 paraphrases in every cluster and close to 100 million pairs in total.

To get comparable results across all datasets, we randomly sub-sampled ParaBank2.0 and WikiAns to the same size as MSCOCO, and split them to train, dev and test sets, of sizes $900K$, $14K$ and $14K$ respectively. We carefully made sure that there are no pairs from the same cluster in different splits of the data. The full data splits will be published with our code.

3.2 Implementation Details

All models are trained with batch size of 32 on 2 NVIDIA A100 GPUs for 6 epochs. Full details as well as train and dev results can be found in Appendix C.1.

QCPG: We use the pre-trained T5-base (Raffel et al., 2020) as the encoder-decoder model. The control input vector to QCPG is quantized at every dimension into 20 equally spaced values ranging from 0 to 100. Each value is assigned to a special saved-token. The three tokens corresponding to the quantized values of the control vector $\mathbf{c}$, are concatenated to the head of the input sentence, and together used as input to the model. $\mathbf{r}(\mathbf{s})$ and $\mathbf{o}$ are also quantized in a similar way.

QP: An Electra base model (Clark et al., 2020) finetuned with MSE loss to predict the typical quality values (see Section 2.3).

Baseline Model (BL): A T5-base model finetuned on the training data.

For all the models, we adopt the experimental setup used in (Devlin et al., 2019), i.e. we train the model with several learning rates and choose the one that achieves the highest dev set performance (see appendix C.1).

4 Results

4.1 Controlling the Quality Dimensions

The aim of the following analysis is to study the level of control achieved by QCPG. To this end, we measure the model response to changes in the input offsets. We compute the expected difference in paraphrase quality, as a result of applying an input offset $\mathbf{o}$ compared to zero offset as a reference. More formally, we define the 3-dimensional responsiveness vector of QCPG at an offset $\mathbf{o}$, $\mathbf{R}(\mathbf{o})$ as $\mathbf{Q}(\mathbf{o}) - \mathbf{Q}((0,0,0))$, where $\mathbf{Q}(\mathbf{o})$ is the expected quality of the paraphrases generated by QCPG at an offset $\mathbf{o}$. We estimate $\mathbf{Q}(\mathbf{o})$ by averaging $\mathbf{q}(QCPG(s, \mathbf{r}(s) + \mathbf{o}))$ over the input sentences s of the dev set, and denote this estimate by $\tilde{\mathbf{Q}}(\mathbf{o}) = (\tilde{\mathbf{Q}}_{\mathbf{sem}}(\mathbf{o}), \tilde{\mathbf{Q}}_{\mathbf{syn}}(\mathbf{o}), \tilde{\mathbf{Q}}_{\mathbf{lex}}(\mathbf{o}))$, and the corresponding estimate of $\mathbf{R}(\mathbf{o})$ by $\tilde{\mathbf{R}}(\mathbf{o})$.

Specifically, in the following analysis we are interested in studying the model response to each of the dimensions separately, i.e. how changing the input offset along a given quality dimension dim – the *controlled* dimension – while keeping the two other dimensions constant, affects the responsiveness in each of the three dimensions. A good control mechanism would imply that increasing the input offset in one dimension will result in a monotonically increasing responsiveness in that dimension, with relatively small responsiveness in the other two dimensions.

Figure 3 shows, for each of the three datasets, the responsiveness in the three quality dimensions, when changing the input offset along each of the three dimensions, while fixing the input offsets in the other two dimensions at 0. Examining the actual values of quality in the paraphrases of the dev sets, reveals that the standard deviation is different in each dimension. Hence, for clarity of presentation, we present the input offset values and the responsiveness in units of standard deviation as measured in the respective dimension and dev set.

For the range of offsets displayed in Figure 3, the responsiveness in the controlled dimension increases monotonically with the input offsets across all datasets and dimensions. As expected, the responsiveness in the uncontrolled dimensions does not zeros due to the inherent coupling between the dimensions. For example, many changes that increase syntactic diversity, also increase lexical diversity (e.g. a move from passive to active voice). Still, our control mechanism is able to increase the responsiveness in the controlled dimension with relative low responsiveness in the uncontrolled dimensions. Specifically, focusing on the relation between semantic similarity and expression diversity, the figure shows that there is a minor decrease in semantic similarity in response to an increase in lexical and syntactic diversity. In the next section, we will show that this does not prevent our model from generating paraphrases that are not only more lexically and syntactically diverse, but also more semantically similar to the source sentences, compared to the paraphrases generated by the uncontrolled baseline.

Figure 3 focused on small to moderate input offsets, i.e. offsets up to 2 stds from the reference point. However, as we speculated before, with increasing offsets, i.e. the more the requested control value deviates from the typical value, it becomes increasingly difficult to generate a paraphrase that conforms to the requested control value. Figure 4 depicts the responsiveness in the syntactic and lexical dimensions for a larger range of offset values. For the semantic dimension, the typical values are too high to allow large positive offsets, which for most sentences result in exceeding the upper limit of the semantic score. Indeed, as can be seen in Figure 4, when moving to high offset values, the responsiveness in the syntactic and lexical dimensions starts to decrease. This behavior is in line with our aforementioned hypothesis, and reflects the detrimental effect of feeding QCPG with input control values that are too far from the typical paraphrase qualities of the input sentence. The non-monotonic behavior of the responsiveness implies that the input offsets should be selected carefully in order to optimize the quality of the resultant paraphrases. In Section 4.2 we suggest a method for identifying these optimal offsets.

4.2 Selecting Optimal Input Control Values

In this section, we suggest a method that given task requirements, selects the input offsets that are expected to yield the desired quality of paraphrases. The idea is to compute the estimated expected quality, $\tilde{\mathbf{Q}}(\mathbf{o})$, for each input offset $\mathbf{o}$, using the dev set as described in Section 4.1, and then search the 3D grid of input offsets to find the point for which $\tilde{\mathbf{Q}}(\mathbf{o})$ is best suited for the user's requirements. We envision this analysis as a preliminary step in which the user chooses the input control parameters that best achieve his desired paraphrasing operation point, and then uses the chosen values at inference – which is why we use the dev set.

We study the behavior of $\tilde{\mathbf{Q}}(\mathbf{o})$ as a function of the 3D grid of offset points in the relevant range, i.e every $\mathbf{o}$ where o_{sem}, o_{syn} and o_{lex} in $0, 5, 10...50$. Figure 5 depicts $\tilde{\mathbf{Q}}(\mathbf{o})$ for WikiAns, on a slice of the full offset grid. The results for the full grid on all datasets are shown in Figure 6. The right-hand-side map depicts the estimated linguistic diversity (the average of $\tilde{\mathbf{Q}}_{\mathbf{syn}}(\mathbf{o})$ and $\tilde{\mathbf{Q}}_{\mathbf{lex}}(\mathbf{o})$) and the left-hand-side depicts the semantic similarity, $\tilde{\mathbf{Q}}_{\mathbf{sem}}(\mathbf{o})$). The maps are presented for $o_{sem} = 50$, and for different values of o_{syn} and o_{lex}. As expected, the two measures are anti-correlated, where areas with increased semantic similarity are characterized by decreased linguistic diversity. The QCPG results are compared to two reference points, which are invariant to $\mathbf{o}$ and are marked on the colorbars with black squares: 'Dataset' is the semantic-similarity/linguistic-diversity average value over the corresponding dev set paraphrases, and 'Baseline' is the average semantic-similarity/linguistic-diversity of the uncontrolled baseline over the corresponding dev set. Notice that the average diversity level achieved by the uncontrolled baseline is lower than that of the dev set mean, reflecting the difficulty of this model to generate diverse paraphrases. QCPG on the other hand, with suitable input offset values, is able to generate paraphrases which are on average higher than the baseline both in their linguistic diversity and in their semantic similarity, and in fact even higher in many cases than the values of the ground truth paraphrases in the dev-set.

In general, the estimates of the expected quality achieved by QCPG at different input offsets, enable a user to generate paraphrases at different operation points, by manipulating the input offset control $\mathbf{o}$ to meet her desired quality values. Con-

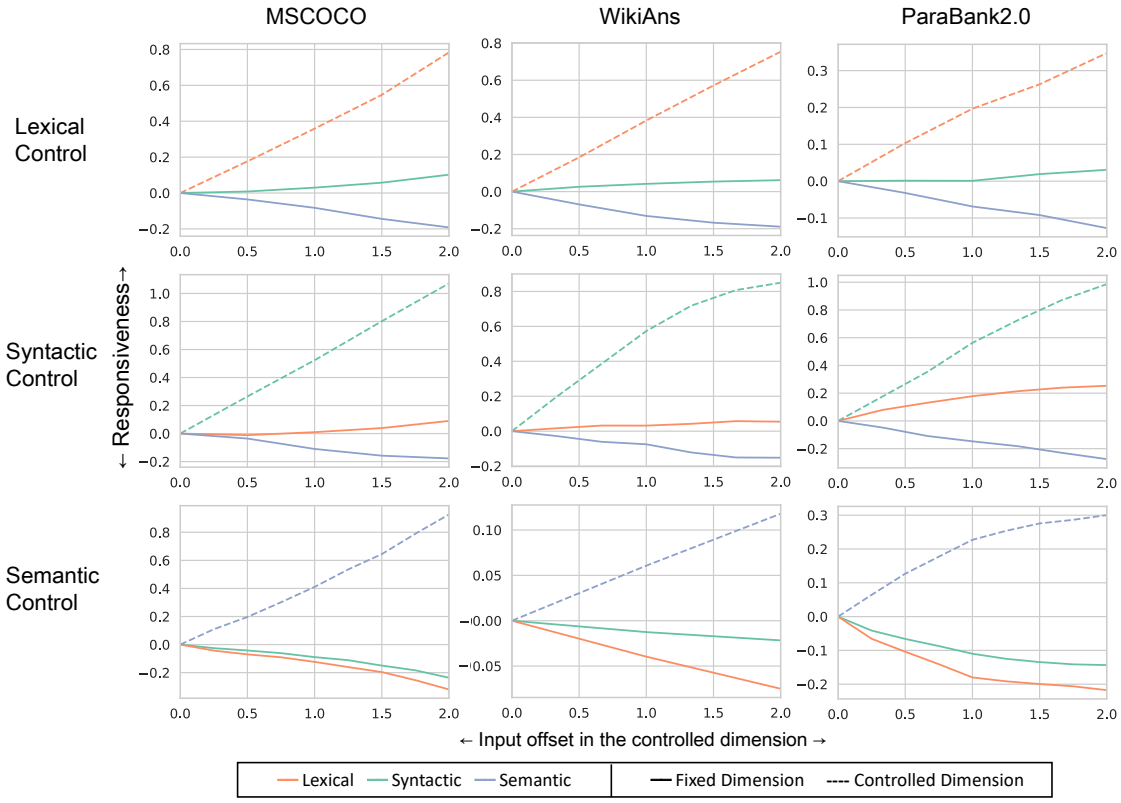

Figure 3: Controllability of QCPG. The responsiveness of QCPG to changes in the input quality vector. In each graph only one dimension of the input is changed (the control dimension), where the other two dimensions are fixed at zero offset. The control dimensions in the top middle and bottom rows are the lexical syntactic and semantic dimensions respectively. Each color represents a different quality dimension of the generated paraphrases. The responsiveness in the control dimension is plotted in a dashed line. Responsiveness and offsets are shown in standard deviation units.

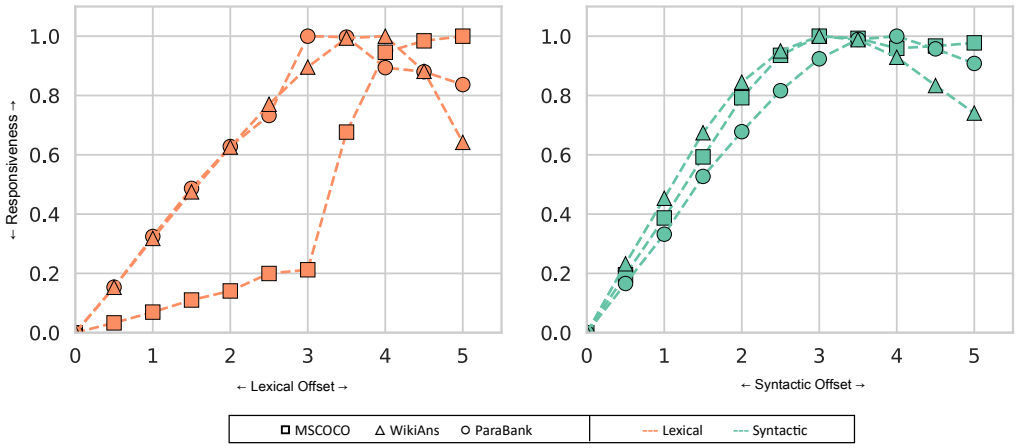

Figure 4: Monotonicity break in the responsiveness of QCPG. The responsiveness of QCPG in the controlled quality dimension vs. the input offset in that dimension in the three datasets. Left: Lexial control. Right: Syntactic control. Each curve is normalized by its maximal value, to create a uniform y-axis range across all datasets. Offsets are shown in standard deviation units.

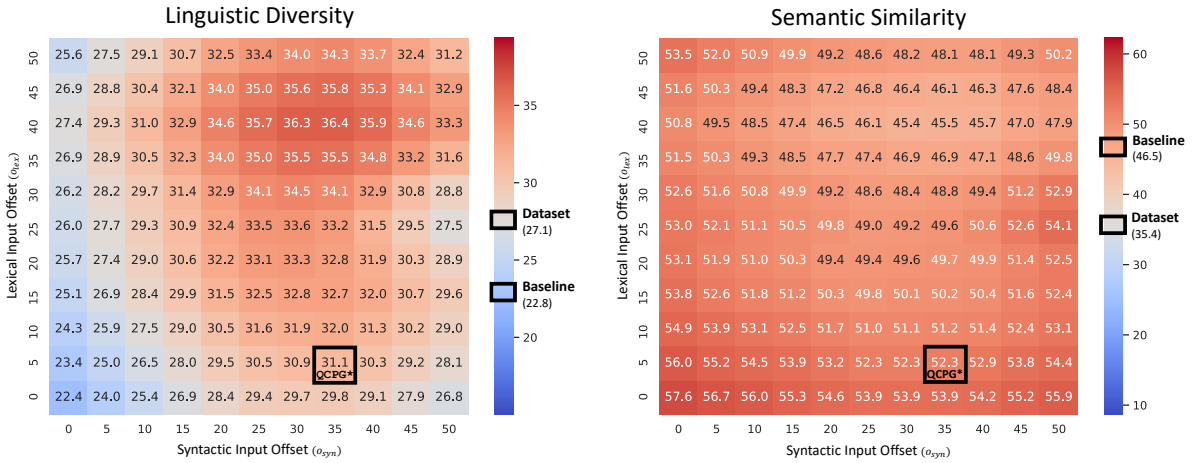

Figure 5: Estimated Quality at different offset values for WikiAns. Average of linguistic diversity (left) and semantic similarity (right) of the paraphrases generated for the dev-set sentences, as a function of o_{syn} and o_{lex}, for fixed $o_{sem} = 50$. The average quality of the gold-label paraphrases, and the average values achieved by the uncontrolled baseline, are marked on the color bars. Red/blue shades correspond to above/below the dev-set mean.

sider for example a typical use case, of aiming to maximize linguistic diversity under a constraint on semantic similarity. An example of such a case is an operation point, denoted by $QCPG^{\star}$, which aims to exemplify the advantage of QCPG over the baseline, by maximizing linguistic diversity under the constraint that the semantic similarity is at least 5 points higher than the baseline. The input offset values to obtain this operation point depend on the dataset, and can be found using heatmaps such as in Figure 5. For WikiAns the input offset for the $QCPG^{\star}$ operation point values are $(50, 35, 5)$ (entry marked by the black square).

4.3 Quality Evaluation on the Test Set

In the previous section we saw, using estimates based on the dev sets, that there are many operation points which generate paraphrases with higher quality than those achieved by the uncontrolled baseline. We now turn to evaluate one such operation point, namely $QCPG^{\star}$, using the source sentences of the *test* sets which were not used in the selection of the input offset values.

Automatic Evaluation We use four quality measures to evaluate different aspects of generated paraphrases. The three quality measures used in the control of QCPG (Section 2.1) and Self-BLEU (Zhu et al., 2018) as adapted in Li et al. (2019); Liu et al. (2020a), which aims to measure the linguistic diversity in the generated paraphrases by penalizing copying from input sentences. As can be seen in Table 1, $QCPG^{\star}$ outperforms the baseline in all metrics across all datasets, as predicted using the dev-set heatmaps. A clear advantage is obtained even for Self-BLEU, which was not part of the metrics used as input controls. Importantly, the quality of the paraphrases generated by our model is comparable to, or at times better than the quality of the paraphrases in the ground truth of the datasets. Examples of paraphrases generated by $QCPG^{\star}$ compared to the ground truth paraphrases appear in Table 10. This is an important step towards the goal of obtaining paraphrases in the sparse area of high quality (recall the top right corner of Figure 1).

Additionally, we examined QCPG from another perspective: the effect of the quality guidance on the model's ability to predict the ground truth paraphrases. Tables 5 and 6 show the BLEU scores (Papineni et al., 2002) obtained by QCPG and the uncontrolled baseline respectively. The results verify that the input quality vectors induced by the target sentences are effectively utilized by QCPG to achieve better prediction performance.

Human Evaluation While linguistic diversity can be automatically measured by reliable metrics such as Self-BLEU, measuring semantic similarity is more challenging. We therefore rely on automatic metrics for evaluating the lexical and syntactic diversity, but use human annotation for validating the semantic evaluation. To this end, we selected a sample of 50 source sentences from each test set, and generated one paraphrase using the uncontrolled baseline and one using $QCPG^{\star}$. The

	MSCOCO				WikiAns				ParaBank2			
	$q_{sem}\uparrow$	$q_{syn}\uparrow$	$q_{lex}\uparrow$	Self-BLEU↓	$q_{sem}\uparrow$	$q_{syn}\uparrow$	$q_{lex}\uparrow$	Self-BLEU↓	$q_{sem}\uparrow$	$q_{syn}\uparrow$	$q_{lex}\uparrow$	Self-BLEU↓
Gold	29.9	34.5	28.0	8.7	34.6	30.7	24.4	16.4	75.0	18.5	20.9	23.9
BL	50.0	27.8	23.0	18.8	46.6	24.7	20.9	23.4	77.8	16.8	18.6	29.4
$QCPG^{\star}$	**56.6**	**29.6**	**42.4**	**18.0**	**48.5**	**41.5**	**24.8**	**21.4**	**81.4**	**18.9**	**19.6**	**27.1**

Table 1: Automatic evaluation of the QCPG model on the test set. The semantic similarity (q_{sem}), syntactic diversity (q_{syn}) and lexical diversity (q_{lex}), are measured using Bleurt, Tree edit distance, and character-level edit distance respectively, as described in Section 2. Self-BLUE is an external measure of linguistic diversity (see text for details). BL: uncontrolled baseline. Gold: the test set ground truth paraphrases. $QCPG^{\star}$ is the QCPG model in the operation point defined in Section 4.2. Best performance amongst the compared models is highlighted in **bold**. Best results amongst the models and the gold labels are underlined.

	Votes			Agreement
	$QCPG^{\star}$	BL	(Tie)	Cohen's Kappa
MSCOCO	**.56**	.36	(.08)	.38
WikiAns	**.48**	.36	(.16)	.47
ParaBank2	**.30**	.26	(.44)	.57

Table 2: Human evaluation of semantic similarity. The numbers represent the proportion of annotators that voted for each method. $QCPG^{\star}$: the QCPG model in the operation point defined in Section 4.2. BL: Uncontrolled Baseline.

annotators were shown the source sentence, along with the two generated paraphrases (randomly ordered), and were asked which of the two better preserves the semantic meaning of the source sentence (ties are also allowed). In total, 150 triplets were evaluated by 5 judges. Table 2 demonstrates an advantage for $QCPG^{\star}$ in all datasets, with a large margin in MSCOCO and WikiAns. This advantage is statistically significant ($p-value < 0.05$) as obtained by applying the Wilcoxon signed-rank test to the difference between the number of annotators that voted for $QCPG^{\star}$ and those voted for the baseline, across all datasets. Thus, the human evaluation is in line with the results of the automatic semantic similarity measure. We also verified, that the results of this sample, in terms of linguistic diversity, are very similar to those shown in Table 1.

For examples of paraphrases generated by $QCPG^{\star}$ see Table 10 in the Appendix.

5 Related Work

Many recent works on paraphrase generation have been focused on attempting to achieve high-quality paraphrases. These works can be divided into supervised and unsupervised approaches.

Supervised Approaches To achieve diversity, some works focused on diverse decoding using heuristics such as Hamming distance or distinct n-grams to preserve diverse options during beam search (Vijayakumar et al., 2018). Other works generate multiple outputs by perturbing latent representations (Gupta et al., 2018; Park et al., 2019). or by using distinct generators (Qian et al., 2019). These methods achieve some diversity, but do not control generation in an interpretable manner.

The works that are most similar to ours strive to gain diversity using controlled-paraphrase generation, by exposing control mechanisms that are manipulated to produce either lexically (Zeng et al., 2019; Thompson and Post, 2020) or syntactically (Chen et al., 2019; Goyal and Durrett, 2020) diverse paraphrases. One approach is to use an exemplar sentence for guiding the syntax of the generated paraphrase (Chen et al., 2019; Bao et al., 2019; Hosking and Lapata, 2021). An alternative is to directly employ constituency tree as the syntax guidance (Iyyer et al., 2018; Li and Choi, 2020). Goyal and Durrett (2020) promote syntactic diversity by conditioning over possible syntactic rearrangements of the input. Zeng et al. (2019) use keywords as lexical guidance for the generation process. Here we introduce a simple model for jointly controlling the lexical, syntactic and semantic aspects of the generated paraphrases.

Unsupervised Approaches Niu et al. (2020) rely on neural models to generate high quality paraphrases, using a decoding method that enforces diversity by preventing repetitive copying of the input tokens. Liu et al. (2020b) optimize a quality oriented objective by casting paraphrase generation as an optimization problem, and searching the sentence space to find the optimal point. Garg et al. (2021) and Siddique et al. (2020) use reinforcement learning with quality-oriented reward combining textual entailment, semantic similarity, expression

diversity and fluency. In this work, we employ similar metrics for guiding the generation of paraphrases within the *supervised* framework.

6 Discussion

In this paper, we propose a novel controlled paraphrase generation model, that leverages measures of paraphrase quality for encouraging the generation of paraphrases with desired quality. We demonstrate the high level of control achieved by the model, and suggest a method for coping with the challenging problem of finding suitable control values.

Aside from offering a simple and effective way for controlling models' output quality, the quality control paradigm enables a holistic view of the data, the training process and the final model analysis. Namely: (I) Examination of the training data through the lens of data quality enables to characterize the data at hand, its strengths and limitations. (II) A quality-aware training process can be viewed as multi-task learning, where each quality level is a separate task with its own accurate supervision, as opposed to the standard quality-agnostic approach, where low quality data is in fact used as a poor supervision for a model which aims at generating higher quality output. (III) Analyzing the model behavior under different quality controls, allows finer understanding of the different model behaviors and the trade-offs between their output qualities. Better understanding the expected output quality of neural NLG models, for different input quality controls, can increase the trust in their output.

Finally, our model analysis consistently shows that although the models generally follow the quality requirements, there is still room for improvement. A possible direction for future research is exploring methods, such as reinforcement learning, for further improving the ability of the model to satisfy the quality requirements.

References

Yu Bao, Hao Zhou, Shujian Huang, Lei Li, Lili Mou, Olga Vechtomova, Xin-yu Dai, and Jiajun Chen. 2019. Generating sentences from disentangled syntactic and semantic spaces. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6008–6019, Florence, Italy. Association for Computational Linguistics.

Rahul Bhagat and Eduard Hovy. 2013. Squibs: What is a paraphrase? *Computational Linguistics*, 39(3):463–472.

Mingda Chen, Qingming Tang, Sam Wiseman, and Kevin Gimpel. 2019. Controllable paraphrase generation with a syntactic exemplar. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5972–5984, Florence, Italy. Association for Computational Linguistics.

Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*.

Wayne W Daniel. 1990. Applied nonparametric statistics pws.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Anthony Fader, Luke Zettlemoyer, and Oren Etzioni. 2014a. Open question answering over curated and extracted knowledge bases. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, page 1156–1165.

Anthony Fader, Luke Zettlemoyer, and Oren Etzioni. 2014b. Open Question Answering Over Curated and Extracted Knowledge Bases. In *KDD*.

Sonal Garg, Sumanth Prabhu, Hemant Misra, and G. Srinivasaraghavan. 2021. Unsupervised contextual paraphrase generation using lexical control and reinforcement learning.

Tanya Goyal and Greg Durrett. 2020. Neural syntactic preordering for controlled paraphrase generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 238–252, Online. Association for Computational Linguistics.

Mandy Guo, Qinlan Shen, Yinfei Yang, Heming Ge, Daniel Cer, Gustavo Hernandez Abrego, Keith Stevens, Noah Constant, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2018. Effective parallel corpus mining using bilingual sentence embeddings. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 165–176, Brussels, Belgium. Association for Computational Linguistics.

Ankush Gupta, Arvind Agarwal, Prawaan Singh, and Piyush Rai. 2018. A deep generative framework for paraphrase generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Tom Hosking and Mirella Lapata. 2021. Factorising meaning and form for intent-preserving paraphrasing. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the*

11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 1405–1418, Online. Association for Computational Linguistics.

J. Edward Hu, Abhinav Singh, Nils Holzenberger, Matt Post, and Benjamin Van Durme. 2019. Large-scale, diverse, paraphrastic bitexts via sampling and clustering. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 44–54, Hong Kong, China. Association for Computational Linguistics.

Mohit Iyyer, John Wieting, Kevin Gimpel, and Luke Zettlemoyer. 2018. Adversarial example generation with syntactically controlled paraphrase networks. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1875–1885, New Orleans, Louisiana. Association for Computational Linguistics.

Changmao Li and Jinho D. Choi. 2020. Transformers to learn hierarchical contexts in multiparty dialogue for span-based question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5709–5714, Online. Association for Computational Linguistics.

Zichao Li, Xin Jiang, Lifeng Shang, and Qun Liu. 2019. Decomposable neural paraphrase generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3403–3414, Florence, Italy. Association for Computational Linguistics.

Tsung-Yi Lin, M. Maire, Serge J. Belongie, James Hays, P. Perona, D. Ramanan, Piotr Dollár, and C. L. Zitnick. 2014. Microsoft coco: Common objects in context. In *ECCV*.

Xianggen Liu, Lili Mou, Fandong Meng, Hao Zhou, Jie Zhou, and Sen Song. 2020a. Unsupervised paraphrasing by simulated annealing. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 302–312, Online. Association for Computational Linguistics.

Xianggen Liu, Lili Mou, Fandong Meng, Hao Zhou, Jie Zhou, and Sen Song. 2020b. Unsupervised paraphrasing by simulated annealing. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 302–312, Online. Association for Computational Linguistics.

Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. 2018. The natural language decathlon: Multitask learning as question answering. *CoRR*, abs/1806.08730.

Philip M. McCarthy, Rebekah H. Guess, and D. McNamara. 2009. The components of paraphrase evaluations. *Behavior Research Methods*, 41:682–690.

Tong Niu, Semih Yavuz, Yingbo Zhou, Huan Wang, Nitish Shirish Keskar, and Caiming Xiong. 2020. Unsupervised paraphrase generation via dynamic blocking.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Sunghyun Park, Seung-won Hwang, Fuxiang Chen, Jaegul Choo, Jung-Woo Ha, Sunghun Kim, and Jinyeong Yim. 2019. Paraphrase diversification using counterfactual debiasing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6883–6891.

Lihua Qian, Lin Qiu, Weinan Zhang, Xin Jiang, and Yong Yu. 2019. Exploring diverse expressions for paraphrase generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3173–3182, Hong Kong, China. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4512–4525, Online. Association for Computational Linguistics.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389, Lisbon, Portugal. Association for Computational Linguistics.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.

AB Siddique, Samet Oymak, and Vagelis Hristidis. 2020. Unsupervised paraphrasing via deep reinforcement learning. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1800–1809.

Brian Thompson and Matt Post. 2020. Paraphrase generation as zero-shot multilingual translation: Disentangling semantic similarity from lexical and syntactic diversity. In *Proceedings of the Fifth Conference on Machine Translation*, pages 561–570, Online. Association for Computational Linguistics.

Ashwin K Vijayakumar, Michael Cogswell, Ramprasaath R Selvaraju, Qing Sun, Stefan Lee, David Crandall, and Dhruv Batra. 2018. Diverse beam search for improved description of complex scenes. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Adams Wei Yu, David Dohan, Minh-Thang Luong, Rui Zhao, Kai Chen, Mohammad Norouzi, and Quoc V. Le. 2018. Qanet: Combining local convolution with global self-attention for reading comprehension.

Daojian Zeng, Haoran Zhang, Lingyun Xiang, Jin Wang, and Guoliang Ji. 2019. User-oriented paraphrase generation with keywords controlled network. *IEEE Access*, 7:80542–80551.

Kaizhong Zhang and Dennis Shasha. 1989. Simple fast algorithms for the editing distance between trees and related problems. *SIAM journal on computing*, 18(6):1245–1262.

Yaoming Zhu, Sidi Lu, Lei Zheng, Jiaxian Guo, Weinan Zhang, Jun Wang, and Yong Yu. 2018. Texygen: A benchmarking platform for text generation models. *SIGIR*.

A Selecting the semantic similarity measure

Recently, several strong metrics have been proposed for measuring semantic similarity between sentences (Reimers and Gurevych, 2019; **?**; Sellam et al., 2020). In order to select the semantic similarity metric for QCPG, we performed a small experiment over the three dev sets, with the aim of measuring the agreement of the candidate metrics with human judgments. To this end, we leveraged two properties that characterize weakly labeled datasets, the underlying clusters of sentences, and the high variability of semantic similarity. Given a dataset, we randomly selected 100 clusters, and picked three sentences at random from each cluster. For each triplet of sentences $t = (t_1, t_2, t_3)$ we asked 5 human annotators to choose which of the two sentences, t_2 or t_3, better preserves the semantic meaning of t_1. In order to find the candidate similarity measure with the highest agreement with human judgments, we first computed, for each triplet, the difference between the number of annotators voted for t_2 and those voted for t_3. We then computed for each candidate measure, the difference between the similarity of t_2 to t_1 and and of t_3 to t_1. We then measured Kendall's Tau correlation (Daniel, 1990) between the difference vector of the human judgments and that of the judgments of each of the candidate measures. Table 3 shows the resultant correlations. The highest correlations are obtained for SBERT (Reimers and Gurevych, 2019), but since it was trained on WikiAns and MSCOCO, we could not use it in our study. We selected Bleurt due to its highest correlation with human judgments over the three datasets (among the methods that were not exposed to the considered datasets). We normalize Bleurt score using the sigmoid function to ensure a uniform range of values, $[0, 1]$, for the three quality dimensions.

	MSCOCO	WikiAns	ParaBank2
SBERT	.52	.43	.41
BERTSCORE	.38	.3	.31
BLEURT	**.45**	**.4**	**.36**

Table 3: Correlation of different semantic similarity models with human evaluations.

B Correlation of semantic similarity measures with linguistic diversity

We study the coupling between the different semantic similarity measures and the linguistic diversity. We assume that the level of coupling of a good similarity measure will resemble that of humans, and will be less sensitive to lexical and syntactic properties of the paraphrase. Table 4 presents the Kendall tau correlation between the different similarity measures and the linguistic diversity. Results for human judgments are also shown for a reference. The correlation calculation is performed between the vectors of differences as described in section A). The results show that Bleurt demonstrates the lowest coupling with linguistic diversity among the automatic measures (aside from SBERT which, as mentioned before, was trained with MSCOCO and WikiAns). The comparison to human judgments shows that Bleurt is more influenced by linguistic features, indicating that automatic measures need to be further improved to reach the decoupling level achieved by humans.

	MSCOCO	WikiAns	ParaBank2
Human	-0.17	-0.19	-0.25
SBERT	-0.17	-0.37	-0.29
BERTSCORE	-0.39	-0.48	-0.51
BLEURT	**-0.25**	**-0.36**	**-0.39**

Table 4: Correlation of different semantic similarity models with linguistic diversity.

Dataset	LR	Dev BLEU ↑	Dev Loss ↓	Train Loss
MSCOCO	1e-3	10.19	2.10	1.52
	1e-4	**10.94**	**1.89**	1.65
	5e-3	0.00	2.23	2.76
	5e-4	10.53	2.07	1.51
ParaBank2	1e-3	27.28	1.38	0.65
	1e-4	**30.22**	**1.15**	0.69
	5e-3	0.00	3.45	3.88
	5e-4	28.40	1.37	0.61
WikiAns	1e-3	13.09	2.24	1.46
	1e-4	**15.22**	**1.95**	1.59
	5e-3	0.00	3.62	4.03
	5e-4	13.51	2.17	1.43

Table 5: Training and dev set loss of the finetuned T5 baseline.

C Models Details and Training Results

The learning rates for the QCPG and the Baseline models were selected in the following way. For a given dataset, we finetuned the models with 4 learning rates ($1e$-3, $1e$-4, $5e$-3, $5e$-4) (The training results of the baseline presented in Table 5 and the results of QCPG presented in Table 6.). For the baseline we selected the one which yielded the best BLEU score (Papineni et al., 2002) on the corresponding dev set The best learning rate for every dataset was chosen based on the Dev set BLEU score. For the QCPG we chose the model that best conforms to the control input as measured by the MSE between the input control vector and the output quality vector (see Table 9). The QP model is an Electra-Base model finetuned with 4 different learning rates ($1.5e$-4, $1e$-4, $3e$-5, $5e$-5). We choose the learning rate the yields the minimal MSE on the dev set (For full results see Table 8)

C.1 Full Heatmaps

The full heatmaps can be found in Figure 6.

Dataset	LR	Dev BLEU	Dev Loss	Train Loss
MSCOCO	1e-3	11.14	2.01	1.47
	1e-4	11.24	1.80	1.61
	5e-3	0.00	2.29	2.89
	5e-4	10.86	1.98	1.46
ParaBank2	1e-3	32.03	1.28	0.60
	1e-4	34.28	1.05	0.65
	5e-3	0.00	3.37	3.86
	5e-4	32.77	1.25	0.56
WikiAns	1e-3	17.29	2.08	1.40
	1e-4	19.48	1.81	1.52
	5e-3	0.00	3.57	4.01
	5e-4	18.21	1.99	1.36

Table 6: Training and dev set loss of the QCPG.

Dataset	Diversity	Lexical	Syntactic	Semantic
MSCOCO	25.4	23.0	27.8	50.0
ParaBank2	17.7	18.6	16.8	77.8
WikiAns	22.8	20.9	24.7	46.6

Table 7: Automatic evaluation of the chosen finetuned T5 baseline.

Dataset	LR	Dev MSE ↓	Train MSE
MSCOCO	1.5e-4	0.0242	0.0240
	1e-4	0.0242	0.0240
	3e-5	0.0206	0.0161
	5e-5	**0.0205**	0.0164
ParaBank2	1.5e-4	0.0260	0.0239
	1e-4	0.0248	0.0239
	3e-5	**0.0169**	0.0124
	5e-5	0.0170	0.0126
WikiAns	1.5e-4	0.0402	0.0374
	1e-4	0.0404	0.0374
	3e-5	**0.0317**	0.0200
	5e-5	0.0445	0.0372

Table 8: Training results of the QP models.

Dataset	LR	MSE ↓
MSCOCO	1e-3	0.0124
	1e-4	0.0119
	5e-3	0.2943
	5e-4	**0.0118**
ParaBank2	1e-3	0.0140
	1e-4	0.0129
	5e-4	**0.0125**
WikiAns	1e-3	0.0166
	1e-4	**0.0153**
	5e-3	0.3091
	5e-4	0.0155

Table 9: MSE between the required control and the evaluations of the outputs of the QCPG models.

MSCOCO

Source	Ground-truth	$QCPG^{\star}$
A table filled with assorted prepared foods in a buffet fashion.	Fresh fruits, vegetables, and other foods are spread out on the table.	A table with food on it in a buffet line.
Ornately decorated assortment of vases displayed on shelf.	A display of pottery in a glass case	A decorated shelf with vases on display
Group of people seated at a long table eating pizza	A group of people are sitting around a wooden table.	A group of people sitting at a long table with pizza.
A building with a clock and weather vane is outlined against the blue sky.	a building with a clock inside of it	A clock and weather vane on a blue sky.
A knitted teddy bear hanging off an afghan	A blue crocheted teddy bear hanging off of a crocheted blanket	A knitted teddy bear hanging from a quilt
Two men pose next to a huge vase with an owl painted on it.	a big vase sits in the middle of a couple of people	Two men standing next to a large vase with an owl on it.

WikiAns

Source	Ground-truth	$QCPG^{\star}$
What did the cheyennes indians do for a living?	Cheyenne indians live in the derest?	What kind of jobs did the Cheyenne Indians have?
What temperature scale do you use in australia?	Temperature scale used for scientific work?	What is the temperature scale for Australia?
Are there any other names for tay sachs disease?	Who is warren tay and bernard sachs?	Other names for tay sachs disease?
What should you give to your elder sister on her birthday?	What should you get your little sister for her 9th birthday?	Your older sister's birthday what to give?
How changes in the respiration rate affect blood pH?	How does Increase in respiration of water affect pH?	Explain how the respiration rate affects the pH?
What is the value of a dollar bill signed by joseph w barr?	What is the value of a dollar bill 1963 signed by joseph barr?	Joseph W Barr dollar bill value?
What are the three meninges that cover the brain and spinal cord?	The three memebranous coverings that protect the brain and spinal cord?	What three meninges cover the brain and spinal cord?

ParaBank2

Source	Ground-truth	$QCPG^{\star}$
We're having trouble with Roger.	I've got issues on Roger.	We have a problem with Roger.
Everything on schedule.	All on schedule.	All in the plan.
The internet no longer maked the distance matter: the world may indeed be our classroom.	Because of the Internet, distance doesn't matter anymore: the world may indeed be an our classroom.	The Internet doesn't matter: the world could be our school.
Article 2 deals with the scope of application of a directive extending cooperation between Member States to include taxes of whatever type.	Article 2 concerns an area which is covered by a Directive which broadens cooperation among Member States so as that it covers taxes of any kind.	Article 2 concerns the scope of the directive extending the cooperation between the Member States to include taxation of any kind.
You're free to move forward.	You're free to move on.	You can go on.

Table 10: Paraphrases generated by $QCPG^{\star}$ compared to ground-truth paraphrases.

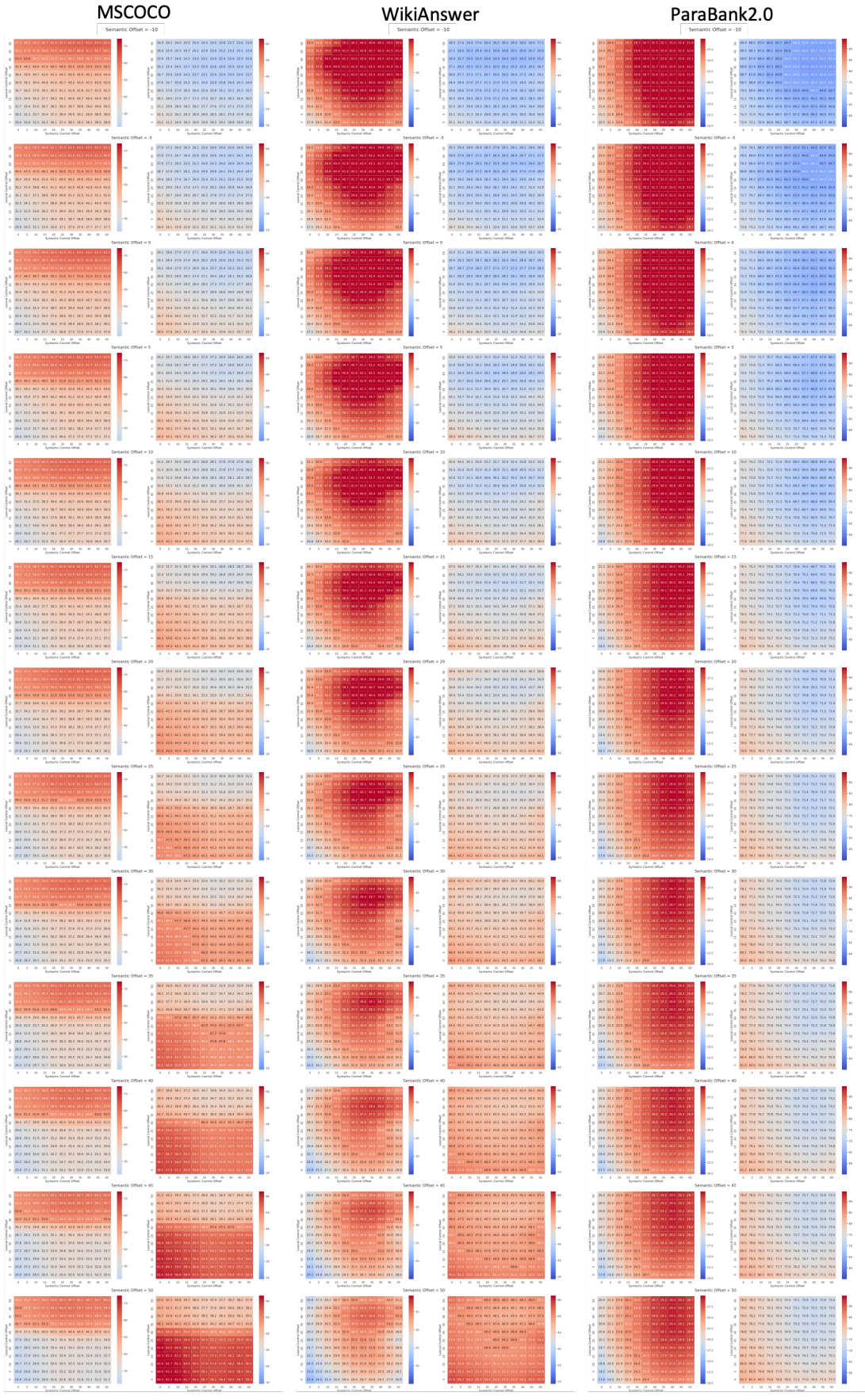

Figure 6: Heatmaps of linugstic diversity (left column) and semantic similarty (right column) as a function of input control offsets for the datasets.