Point-less: More Abstractive Summarization with Pointer-Generator Networks

Freek Boutkan
University of Amsterdam

Jorn Ranzijn
University of Amsterdam

David Rau
University of Amsterdam

Eelco van der Wel
University of Amsterdam

ABSTRACT

The Pointer-Generator architecture has shown to be a big improvement for abstractive summarization seq2seq models. However, the summaries produced by this model are largely extractive as over 30% of the generated sentences are copies from the source text. This work proposes a multihead attention mechanism, pointer dropout and two new loss functions to promote more abstractive summaries while maintaining similar ROUGE scores. Both the multihead attention and dropout do not improve N-gram novelty, however, the dropout acts as a regularizer which improves the ROUGE score. The new loss function achieve significantly higher novel N-grams and sentences, at the cost of a slightly lower ROUGE score.

1 INTRODUCTION

More data is becoming available on the web every day, for instance in the form of news articles and scientific publications, and extracting the most relevant information is becoming increasingly difficult. A well-written summary should be able to provide the gist of a text and can help to reduce the effort of obtaining all the relevant information. Automated text summarization has therefore received a lot of interest since the advent of deep learning techniques [1]. The process of summarization is often divided into extractive summarization and abstractive summarization. In extractive summarization, a summary is obtained by copying the relevant parts of the text. Abstractive summarization aims to distil the relevant information from the source text into a summary by paraphrasing and is therefore not limited to the use of the exact phrases of the source text.

See et al. [18] propose a new approach for abstractive summarization by combining a sequence-to-sequence model [21] with a Pointer Network [23]. Additionally, See et al. [18] incorporate a coverage mechanism that aims to tackle the problem of over-generation by penalizing attention to words in the source document that have already received attention in past timesteps.

One issue with pointer-generator networks is that, during test time, the model focuses mainly on the source text during summary generation and does not introduce many novel words. The resulting summaries therefore tend to be more extractive than abstractive [2, 18]. See et al. [18] state that this focus on the source text is likely caused by the word-by-word supervision during training time, which is not possible during test time. Another issue with the pointer-generator architecture is that the generator is undertrained as the network learns to use the pointer mechanism early in training and arrives at a local minimum. We hypothesise that these two factors are the main contributors to the over-reliance on the pointer mechanism during test time.

Another limitation is the used dataset and evaluation metric ROUGE. This is discussed in detail in section 2. In abstractive summarization there are many candidate solutions. However, the provided dataset rarely contains all of these and perfectly viable summaries are sometimes penalized. This is closely related to the problem with the ROUGE metric as this can also produce low score for viable summaries [18]. These problems are not specific to the pointer-generator model, and addressing them is less obvious.

The goal of this research to increase the number of novel N-grams while obtaining similar ROUGE scores, therefore improving abstraction in an end-to-end trainable text summarization model. Our contributions comprise of

- Multihead attention over the source text
- Dropout mechanism over the pointer
- Naive pointer regularization
- Pointer regularization based on word priors

2 RELATED WORK

Abstractive and extractive summarization. In extractive summarization, fragments of the source text are concatenated to generate a summary [1]. An advantage of this task is that it is relatively easy to obtain a summary with good fluency and factual correctness. In contrast, abstractive methods allow for the use of synonyms, generalization, and rephrasing of the source text. While in theory this can lead to results that are closer to human generated summaries Jing [10], it is a much more difficult task than extractive summarization. Common problems include factual and grammatical mistakes but also over/under generation of words [1].

In more recent work [7, 18], hybrid models are proposed to combine the strengths of both methods. These models can create abstractive summaries with extractive elements to promote factual correctness, and out-of-vocabulary (OOV) word generation.

Pointer networks. The incorporation of a copying mechanism to the sequence-to-sequence has proved to be a powerful addition for summarization tasks. Both the CopyNet [7] and Pointer-Generator

[18] propose adding such a mechanism to bypass the generator network, in order to generate words directly from the input document. While this is useful in many cases, both papers observe balancing the strength of the pointer mechanism and the generator is a difficult task. The pointer generator seems easier to train and, as a result, most of the generated summary is generated by directly copying from the source.

Weber et al. [24] confirmed the over-reliance on the pointer mechanism and introduced a penalty during beam decoding in order to increase the probability of generating a word from the generator distribution. However, no changes are made to the training process and the clear downside of this approach is that the over-reliance is not solved during training time but only afterwards.

Song et al. [19] add structural elements to the copy mechanism. They say a possible problem of the copy mechanism is that it only looks at semantic information, while structural information (such as grammatical structure) might be more important for generating good summaries.

Attention. A main difference between the Copynet architecture [7] and Pointer-Generator [18] is that Copynet uses a separate attention distribution for pointing and generating, while the Pointer-Generator only uses one. See et al. [18] pose that similar information is needed for both pointing and generating, and that decoupling the two distributions might lead to a loss in performance. However, a popular recent architecture proposed for machine translation takes an opposite approach. Vaswani et al. [22] propose a multi-head attention mechanism, which is able to learn multiple attention distribution over an input sequence. These attention mechanisms are merged and projected with a linear layer, and can theoretically encode a more varied representation of the input sequence compared to the regular attention mechanism.

Fan et al. [6] are the first to use multi-head attention with a comparable model architecture for abstractive summarization. They show that multi-head mechanisms are useful for summarization tasks and that different useful features are learnt by the different attention heads. This could be particularly useful for the pointer-generator, since the distribution used by the pointer, and the distribution for the generator are identical in the original architecture.

Model evaluation. Generated summaries will be compared against provided target summaries. The *ROUGE* score Lin [16] indicates the recall of overlapping N-grams between the generated and target summary. Using ROUGE as the evaluation metric is problematic as has been noted by Dohare et al. [4]. Not only because ROUGE scores do not correlate with human judgement but more fundamentally because ROUGE can not evaluate restructured sentences in a proper way. ROUGE matches overlap in complete words and in reconstructed sentences different word forms can be used which might lead to low ROUGE scores. See et al. [18] shows an example of a valid summary that has a ROUGE score of 0.

Krantz and Kalita [12] propose a new metric *VERT* that compares similarity scores of sentences. Since this method does not match exact word forms and it is to some extend robust to grammatical changes, word reordering and sentence reconstructions. Versatile Evaluation of Reduced Texts (VERT) is made up out of a similarity and dissimilarity sub-score. A sentence vector is created out of the reference and created summaries and the cosine similarity between these two vectors is measure of semantic similarity between the summaries. The dissimilarity sub-score is calculated using the word-mover-distance algorithm that indicates how much a created summary has to change in order to match a reference summary. This new metric correlates stronger with human judgement compared to the commonly used ROUGE metric.

Both ROUGE and VERT only measure the accuracy of the generated sentences with respect to the target summary but they provide no insight in the *abstractiveness*. To measure abstractiveness we use the proportion of new N-grams in the generated summary. A low proportion of higher order N-grams indicates that the model is copying long phrases from the input sequence and is thus acting in a more extractive way. Improving both ROUGE and novel N-grams seems like a contradiction since improving ROUGE will decrease the number of new N-grams if the generated summary is not rephrased in the same way as the reference summary.

Directly improving novel N-grams using policy learning. Kryściński et al. [13] optimize the ROUGE score directly. Since the ROUGE metric is not differentiable this can only be done by using reinforcement techniques such as policy improvement. The loss function combines the maximum likelihood and ROUGE objective. In addition an abstractive reward is added to the loss. This reward is defined as the proportion of novel N-grams in the generated summary. This metric has a bias towards very short summaries and needs to be normalised using the length ratio of the generated and ground-truth summaries. They achieve similar ROUGE scores as [18] but show that the number of new N-grams increases significantly and thus is less extractive.

3 METHODS

In this section we describe our dataset (3.1) and (3.2) the baseline pointer-generator network. Then we introduce our extensions over the baseline network which comprises our multi-head attention (3.4), pointing penalty losses (3.5) and pointer dropout mechanism (3.3).

3.1 Dataset

We use the CNN Dailymail dataset Hermann et al. [8]. We use the same preprocessing and training splits as See et al. [18], which in turn uses the method from Nallapati et al. [17]. The training set consists of approximately 287k training pairs, with a validation set of 13 thousand pairs and test set of 11 thousand examples. The average article length is 781 tokens and the summary length is on average 56 tokens (±3.75 sentences).

3.2 Pointer-Generator network

The baseline model is the pointer-generator network described by See et al. [18]. This model allows for copying of words from the source document using a pointing mechanism and also generation of novel words by selecting words from a fixed vocabulary. The main advantage of this approach over previous methods is that it allows the model to produce out of vocabulary words during summary generation.

The basic architecture is a sequence-to-sequence attention model. Words from the source document are fed sequentially into a single bidirectional LSTM resulting in a sequence of encoder hidden states. The decoder is a single layer LSTM that is initialised with the final hidden states of the encoder. More specifically, a linear layer maps the final bidirectional hidden states to a fixed size output that represent the initial values of the decoder at the first time step. During the decoding at time step t, an attention distribution is calculated over the source words:

$$e_i^t = v^T \tanh(W_h h_i + W_s s_{(t)} + W_c c_i^t + b_{att})$$

$$a^t = \text{softmax}(e^{(t)})$$

Here, $v^T, W_h, W_s, W_c, b_{att}$ are learnable parameters. s_t refers to the output of the decoder at time step t and h_i is the representation of the word at position i produced by the encoder. c_i^t is the coverage vector.

See et al. [18] include a coverage mechanism in their model to reduce the amount of repetition. By taking into account the amount of attention that has been given to words from the source text in previous time steps, they manage to significantly decrease repetition in the produced summaries. The coverage vector at time step t is just the sum of attention of the previous time steps:

$$c^t = \sum_{t'=0}^{t-1} a^{(t')}$$

The attention distribution indicates which words from the source text are relevant to produce the next word of the summary. This information is stored in a fixed size representation called the context vector, h_t^*, that is a weighted combination of the encoder hidden states:

$$h_{(t)}^* = \sum_i a_i^{(t)} h_i$$

Based on the context vector and decoder hidden state, a probability distribution is calculated over the fixed size vocabulary. The context vector and decoder hidden state are concatenated and subsequently fed through two linear layers and a softmax to obtain a valid probability distribution over the vocabulary words, called p_{vocab}.

$$p_{vocab} = \text{softmax}(V'(V[s_{(t)}, h_{(t)}^*] + b) + b')$$

Here, V', V, b' and b are the learnable parameters. A copy distribution over the source words is also required in order to select words from the source text during summary generation. See et al. [18] decided to recycle the corresponding attention distribution and also made it serve as the pointing distribution. The probabilities of the words that occurred multiple times in the source text were summed.

A trade-off has to be made between copying a word with the help of the attention distribution and generating a word by the p_{vocab} distribution. Therefore, a generation probability, p_{gen} was introduced that acts as a soft switch. A p_{gen} of 1 would mean that only words from the p_{vocab} distribution can be used and none from the pointing distribution while a p_{gen} of 0 has the opposite effect.

$$p_{gen}^{(t)} = \sigma(w_{h^*}^T h_{(t)}^* + w_s^T s_{(t)} + w_x^T x_{(t)} + b_{ptr})$$

Here, $w_{h^*}^T, w_s^T, w_x^T, b_{ptr}$ are learnable parameters. x_t refers to the input of the decoder at time step t and σ is the sigmoid function.

For every document, there is an extended vocabulary that is the union of the words in that document and all the words in fixed vocabulary. Now, the probability of a word in this extended vocabulary is defined as (where $p_{point} = 1 - p_{gen}$):

$$p(w) = p_{gen} \cdot p_{vocab}(w) + (p_{point}) \cdot \sum_{i:w_i=w} a_i^{(t)}$$

The loss function at time step t is defined as:

$$loss_t = -\log p(w_{(t)}^*) + \lambda \sum_i \min(a_i^{(t)}, c_i^{(t)})$$

where the first term is the negative log likelihood of target word w^* and the second term is the coverage loss. This coverage loss is introduced to penalize repeated attention to the same words and is reweighted by a hyperparameter λ. The final loss function is defined as the average loss over all time steps:

$$loss = \frac{1}{T} \sum_{t=0}^{T} loss^{(t)}$$

3.3 Dropout mechanism

We propose a dropout mechanism on the pointer network to make the model less dependent on the pointer mechanism. Generally, dropout is a simple method to prevent overfitting in neural networks [20], by dropping parts of the network during training. With a predefined probability weights are set to zero during training. This ensures that the model can not rely on hidden co-dependencies and generalises better. During evaluation the pointer-generator model tends to rely to much on the pointer mechanism. The contribution of the generator network to the final output probability is on average only 17%. Our pointer dropout method can be implemented by randomly, setting p_{gen} with probability 0.2 to 1 during training, where a value of 1 makes the output distribution of the model the same as the output of the generator. We expect the model to rely less on the pointing mechanism and use the copy mechanism only when necessary. Hopefully, this would result in a model that generates more abstractive summaries.

3.4 Multihead attention

In the original paper, the pointer and the generator make use of the exact same attention distribution. In our opinion this is problematic because pointer and generator carry out different functions that require different underlying features. For example, the generator might use syntactical features to create a correct sentence structure or point to multiple words to create a more abstract summary. In contrast, the pointer only attends to words that it wants to copy to the summary.

In order to both differentiate between pointer and generator attention distributions, but still supply all information of the pointer mechanism to the generator, we use a modification of the multi-head attention mechanism [22]. Figure 1 shows a schematic of our new pointer-generator multi-head attention mechanism where the first attention head is shared between the pointer and the generator, whereas the generator receives all attention heads. This way, by

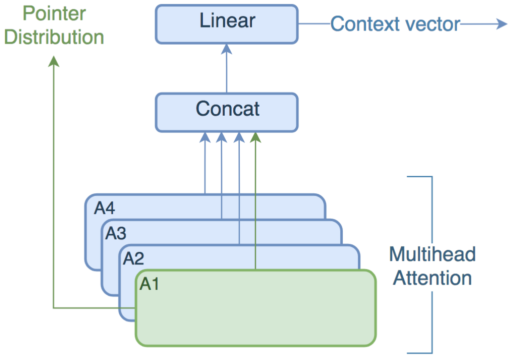


Figure 1: A schematic of the pointer-generator multi-head attention with four attention heads, where the first head is used as shared attention.

introducing regularizations to the pointer mechanism, we only affect the shared attention head while dedicating the rest of the attention heads to generator specific features.

3.5 Pointing losses

See et al. [18] show that the model exploits the pointing mechanism during evaluation. We hypothesise this is because pointing is easier than generating sentences. The model takes a shortcut by copying a lot of phrases and sentences in cases where this might not be necessary. To discourage the network to count on the pointer network we add a term to the loss called the naive pointing loss. We add the sum of all pointing probabilities and weigh it with a hyperparameter λ_p. This way the model can still use the pointer network but it will directly contribute to a higher loss. For readability, we define the pointer mechanism weight $p_{point} = 1 - p_{gen}$.

$$\mathcal{L}_{naive} = \lambda_p \sum_{t=0}^{T} \left(p_{point}^{(t)}\right)$$

A disadvantage of this relatively simple penalty term could be that every word gets the same penalty, even for words where pointing is desired. We propose a second pointer penalty term, the word prior pointing loss, that only penalizes the pointer when a word is common in the vocabulary.

$$\mathcal{L}_{WP}^{(t)} = \lambda_p \cdot p_{point}^{(t)} \cdot - \sum_{w \in X} p_w(w) \log(1 - p_a^{(t)}(w))$$

Where $p_a^{(t)}$ is the attention distribution used for the pointer mechanism and p_w is a pre-calculated word prior. As with the previous loss, λ_p is a hyper parameter to weigh the influence of the loss during training.

Intuitively, the cross entropy between the prior p_w and $(1 - p_a^{(t)})$ expresses the surprisal of not pointing to a word given the word prior. If the prior is high, and the word has a high weight in the attention distribution, this loss term will be high. In any other case, the loss term will be small.

The desired behaviour of this loss term is that p_{point} gets minimized when attending to words with a high prior. However, when naively implemented, the model could also minimize the cross entropy when p_{point} is high. This would cause the model to attend to uncommon words while pointing, which is not what the loss is supposed to achieve. To prevent this, the gradient of this loss term is only back-propagated to p_{point} during training.

4 EXPERIMENTS

This section describes the experimental setup (4.1) that is shared between all of the models and also the different model variations (4.2) that are tested.

4.1 Experimental setup

All the experiments follow the same setup as described by See et al. [18]. A fixed size vocabulary of 50.000 words is used for both the source and target words. All models use 128-dimensional word embeddings that are learned during training time and hidden states for the pointer and generator are kept at a fixed size of 256. The input summaries are truncated to 400 tokens during training and test time. The reason for this is that the most important words for the summary appear in the beginning of the articles and keeping longer source document even decreases performance [18]. Summary lengths are limited to 120 tokens during training time and 100 during test time in order to speed up training. Summaries are generated using beam search and use a beam size of 4 at test time. All the model parameters are optimised with Adagrad [5] using a learning rate of 0.15 and accumulator value of 0.1 as this proved to work best for See et al. [18]. Gradients are clipped with a maximum norm value of 2 and no further regularisation methods are used.

4.2 Model variations

The baseline model is the pointer-generator network described by See et al. [18]. The baseline is trained with and without coverage where the coverage mechanism is trained separately after 13 epochs for 3000 iterations as including this from the beginning turns out to decrease performance.

The multi-head attention mechanism is tested with four heads. Every head produces a context vector that is $\frac{1}{4}$th of the size of the context vector of the baseline. Next, these context vector are concatenated into a single vector resulting in a vector of the same size that is independent on the number of heads. This is similar to the approach taken by Vaswani et al. [22].

The probability for dropping out the pointer mechanism is set to 0.2. The decision to drop out the pointer holds for all the words during summary generation. This means that for some summaries, the model can not rely on the pointing mechanism at all.

The pointing losses are added at the end of training and for the same amount of iterations as the coverage loss. For both losses experiments were conducted with different scalars (1, 0.4, 0.2, 0.05). As best performing scalars 0.05 for NLoss and 0.2 for WPLoss were found. Prior probabilities of the words are calculated on the occurrence in the entire training set. Also, only the prior probabilities for words that occur in the generation vocab are calculated. For the other words, this probability is set to zero. Every adaptation to the baseline model that is proposed in this work is tested as a separate addition to the baseline. This gives a clear estimation of the influence of each adaptation although it leaves out the influence of possible interactions that might occur. Due to the time intensive training we use the same hyperparameters for all models.

In order to test our results for **statistical significance** we perform Wilcoxon signed-rank tests Wilcoxon [25]. All models without coverage are compared to the baseline whereas models with coverage are tested against the baseline + coverage model. Statistical tests for ROUGE 1, ROUGE 2 and ROUGE L are conducted. A *p-value* of 0.01 is used to determine statistical significance. Since the VERT score correlates strongly with the ROUGE scores separate tests are not needed. The differences in novel N-grams are generally much bigger and have lower variance than ROUGE scores so statistical tests are not needed for a large test set (11k examples).

5 RESULTS

Table 1: Mean ROUGE F_1 and VERT scores of the tested models (11k examples in testset). All models were trained from epoch 13 on with coverage for 3000 steps. Here NLoss corresponds to naive pointing loss and WPLoss to the word prior pointing loss. Scores with a star are not significantly different from the baseline. Best results are marked in bold.

Attention heads	Model extensions	ROUGE 1	ROUGE 2	ROUGE L	VERT
1 (*baseline*)		38.14	15.82	33.47	0.706
4		38.15*	15.76*	33.47*	0.707
1	dropout	**38.35**	**15.94**	**33.62**	**0.708**
4	dropout	38.09*	15.75*	33.51*	0.706
1	NLoss	37.63	15.64	33.02	0.702
4	NLoss	37.35	15.41	32.74	0.699
1	WPLoss	36.48	14.93	31.55	0.690
4	WPLoss	36.81	14.95	32.13	0.693
1 *(baseline See et al. [18])*		39.53	17.28	36.38	-

In Table 1 the average ROUGE scores are reported on all models, and Table 2 shows the amount of novel N-grams and sentences. These models are all trained with coverage, results without coverage are included in Appendix A.

The obtained baseline ROUGE scores are on average 3 points lower than reported in See et al. [18] paper. However, we use a pytorch re-implementation [1] and did not perform any hyperparameter tuning to optimize this score. As a reference to compare our models to, we therefore use the *baseline* that we trained ourselves.

Table 2: Percentage of novel N-grams and sentences that are produced for each of the tested models. Best results are marked in bold.

Attention heads	Model extensions	1-grams	2-grams	3-grams	4-grams	Sentences
1 (*baseline*)		0.17	3.24	8.12	12.80	79.52
4		0.12	2.90	7.56	12.12	78.38
1	Dropout	0.12	2.67	7.13	11.51	75.94
4	Dropout	0.23	3.20	8.07	12.77	78.30
1	NLoss	0.31	5.21	12.04	18.67	86.57
4	NLoss	0.30	5.06	11.92	18.17	86.20
1	WPLoss	**1.44**	**8.86**	**17.96**	25.50	**92.00**
4	WPLoss	0.95	8.24	**17.96**	**26.07**	91.05
Target summaries		16.95	52.48	72.36	81.94	98.97

On average, multi-head obtains slightly worse ROUGE scores. This can be a simple case of undertraining and sub-optimal choice of hyperparameters. However, we notice that in case of the Word Prior model, the multi-head architecture performs better. A possible explanation is that the decrease of weights in the pointer head might hurt the ROUGE score if the model mostly relies on the pointer, and when the generator gets a more important task the multi-head might be beneficial. This hypothesis needs more extensive training and tuning to prove, and is not in the scope of this research.

Dropout on the single head model achieves slightly higher ROUGE scores and does increase N-gram novelty. The multi-head dropout model is not significantly different from the baseline.

Both proposed losses greatly improve the number of novel N-grams and sentences. This is especially noticeable in case of the word prior loss; The number of novel N-grams is more than double. However, in both cases this increase in novel N-grams decreases the ROUGE score.

In example 4, we can clearly observe that the model favours generating over pointing when predicting simple words like articles or prepositions. On less common words, like names and uncommon nouns, the pointer is still used.

5.1 Is the model pointing less?

Table 3 shows the average p_{gen} during train and test time, which show how much the model uses the pointer mechanism on average. The baseline has a p_{gen} of 0.18 on average, which is in line with the findings of See et al. [18]. The Multihead does not change this behaviour. While the Dropout model uses the generator significantly more during training, during test time it falls back to the same value as the baseline.

Both loss functions greatly increase the amount the generator is used. This is to be expected: When actively penalizing the pointer mechanism, the pointer mechanism is used less at test time.

Table 3: Average value of p_{gen} during the end of training and test time.

Model	train p_{gen}	test p_{gen}
Baseline	0.54	0.18
4 Heads	0.51	0.17
Dropout	0.67	0.17
NLoss	0.77	0.32
WPLoss	0.86	0.36

The examples in Appendix B show the p_{gen} value on each generated word for the baseline, and two new losses. The model trained word prior loss shows that it achieves a much higher average p_{gen} on common words, such as articles and verbs. In the first example only three fragments have a low p_{gen}: "Ellie Meredith", "Down Syndrome", and "let". The first two fragments are cases where we want the model to point, whereas the third fragment is less clear. On inspection of the source article, this fragment starts a direct quote from the article: "Let's party like it's 1989".

[1] https://github.com/lipiji/neural-summ-cnndm-pytorch

Table 4: Most frequent novel words for the best model (4 heads, coverage and word prior loss).

says	beat	diagnosed	unk	taken
scored	>	say	premier	boss
been	<	:	year	since
he	has	found	said	by

5.2 Novel Words

To investigate the novel words that are used further the most frequent new words for the most abstractive model (multihead, coverage and WPLoss) is calculated (Tab. 4). It can be noted that the majority of the words are verbs. When a sentence is rephrased it can happen that the same root of a verb is used but with another suffix. Similarly, when the tense of a verb changes this can introduce new words. This suggests that the newly introduced words are valid rephrases and not random words. Note that the tokens < and > are the result of incorrect parsing of <unk> and <s> tokens.

The average reference summary length is 56 tokens. The length of the generated summaries is approximately 46 for models with a single attention head and word prior loss and around 55 for all other models (including the baseline). For none of the models the generated summary was on average longer than the target summaries. The maximum allowed length for generating summaries is 120. The new N-grams are thus not a result of simply generating more text because the average length does not change. Instead the novel N-grams replace non novel N-grams. This observation (and the type of newly generated words, as can be seen in table 4) suggests that the novel N-grams are valid rephrasings and not random words or model artefacts.

6 DISCUSSION

6.1 Multi-head Evaluation

The multi-head attention mechanism improves the results of the new loss function in our measurements. The ROUGE L score increases by 0.7 (NLoss) and 0.6 (WPLoss) but the novel N-grams drop slightly for both losses. This shows that penalizing the pointer mechanism when the attention between pointer and generator is shared can reduce the overall quality of summaries, which indicates the multi-head mechanism is working as intended: by splitting the pointer and generator attentions, penalizing the pointer affects the generator less.

Figure 2 shows the KL divergence between each head of the multi-head attention, where cell (i, j) corresponds to $D_{KL}(head_i || head_j)$. In the last column the KL divergence between the attention distributions of the multi-head, and the attention distribution of the same model with just one attention head is shown. We can read from the plot that the head used for the pointer ('head 1') is on average very different from the other heads in the multi-head. This means that on average they attend to different words in the source text, and perform a different function when generating words.

On the other hand, the pointer head is most similar to the attention distribution in the single head model. Additionally, the other heads in the multi-head are more similar to the single head than to eachother. This result indicates that the single head model attempts to incorporate information needed for both pointing and generating, but that it can be desirable to split this information into multiple heads.

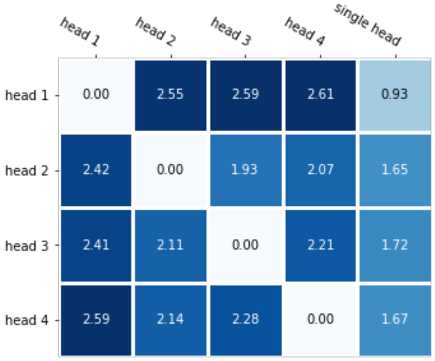


Figure 2: The average KL divergence between each pair of heads of the multi-head model, and between each head of the multi-head and the single head model (last column).

6.2 Dropout

Models that used the dropout mechanism show an increased ROUGE score while having lowered amounts of novel N-grams. This is the exact opposite of what was expected. A possible explanation is that dropout does not produce gradients that indicate that the pointing mechanism is wrong but that it only slows its training. Still, it does force the generator to be more reliable and this leads to better scores in general. This is in line with the idea of See et al. [18] that the generator might not be optimally trained because of over-reliance on the pointer. Using dropout is therefore very similar to the baseline model expect that dropout does give the generator the opportunity to be better trained.

6.3 Coverage ablation studies

To understand the relationship between the coverage loss and our proposed losses we trained all models without the coverage mechanism. Tables 6 and 7 in the appendix show that the Rouge-L score decreases about 2 points for all models without using the coverage loss. Further, it can be seen that the number of novel N-grams decreases significantly. This matches our expectations for N-grams > 1 which reflect the order of words. However, we can also observe a decrease in novel 1-grams which suggests that the coverage mechanism favours extractive summarization which stands in contrast to our goal towards a more abstractive model. This claim is supported by table 5 which shows that introducing the coverage mechanism reduces the amount of repetition whereas both WPLoss and NLoss result in more duplicate N-grams. The losses thus interfere with coverage. Coverage reduces repetition at the expense of more extractive summaries. Penalizing the pointing mechanism introduces new N-grams but also increases the problem of overgeneration.

Table 5: Duplicated N-grams within summaries for multi-head models where cov stands for models that have been trained with coverage.

Model extensions		1-grams	2-grams	3-grams	4-grams	Sentences
baseline		0.32	0.20	0.19	0.18	0.02
NLoss		0.34	0.24	0.22	0.20	0.03
WPLoss		0.40	0.29	0.27	0.25	0.00
baseline	cov	0.22	0.07	0.06	0.05	0.00
NLoss	cov	0.24	0.11	0.09	0.08	0.01
WPLoss	cov	0.29	0.16	0.13	0.12	0.01

6.4 Pointer and generator distributions

When examining the examples produced by the model, we notice that even with a high p_{gen}, the model is copying full sentences. This effect can be seen in Figure 6 (Appendix B): The last sentence is mostly made with the generator, but is a copy of line 11 in the source article. An issue of the pointing distribution is that it has a much lower cardinality compared to the generator distribution, in our case it is two orders of magnitude smaller (401 for the pointer, 50k for the generator). This means that there is generally a bias towards words from the source text, even with high values of p_{gen}. We can conclude that the pointer mechanism by definition decreases the novel 1-gram score, and makes the model less abstractive.

Instead of learning the soft switch p_{gen}, which introduces this bias, a hard pointing mechanism could be learnt by reinforcement learning or with a Gumbel-Softmax approximation [9] to diminish this bias.

6.5 ROUGE metric and dataset

The aforementioned problems of ROUGE are a serious limitation to the evaluation of abstractive summarization models. The idea of abstractive summarization is that valid summaries can be created in many ways. However, ROUGE measures the exact overlap between a set of references summaries and summaries created by our models. This means that either the set of reference summaries should be representative of all valid ways in which abstractive summaries can be produced or that there is a need for a new metric that does not rely on exact overlap but rather on semantic similarity.

7 CONCLUSION

In this work, we investigated several additions to the Pointer-Generator framework in order to improve abstraction while maintaining similar summary quality.

While the multi-head mechanism learns different features for pointing and generating, it does not improve the ROUGE score. When dropout is used on the pointer mechanism, the multi-head attention does promote novel N-grams in the produced summary, but in other models the results are very similar.

The dropout mechanism does the opposite of what was expected as the amount of novel N-grams decreased while the ROUGE scores increased. It seems likely that dropout partly removes the over-reliance on the pointer and therefore gives improved performance compared to the baseline. The two introduced loss functions improve the generation of novel N-grams significantly. For the Word Prior loss, we observe an improvement of 12.5% more novel sentences compared to the baseline model. However, in both cases we did not manage to maintain the same ROUGE scores. This might be a problem with the loss functions, but also with the training process.

The VERT metric looked like a promising metric to measure the semantic similarity between generated and target summaries. However, the resulting VERT scores are completely correlated with the ROUGE metric and do not produce any new insights.

As our goal was to train a model that is more abstractive we used the number of novel N-grams with respect to the reference summary as a measure. The novel N-grams score can easily be increased by adding random words to the summary. This does not lead to more abstractive or higher quality summaries. Since there is no metric to evaluate abstractive summarization tasks effectively it is not possible to claim that the summaries are more abstractive based on just the increase of new N-grams. However we have shown that the summary length does not increase which suggests that new words replace other words instead of adding more words. We have also shown that the novel words are plausible words for rephrasing/summarization tasks.

7.1 Future work

It appears that the newly introduced losses interfere with the coverage mechanism and increase the over-generation problem. It could be the case that by introducing the coverage and new loss at the same point in training produces gradients that are too different from training without these losses, which would interfere with the convergence of the network. In future work, the interaction between these new loss function and the coverage can be investigated.

The difference in pointing behaviour between training and inference could be reduced by using scheduled teacher-forcing [3] , which gradually decreases the frequency the model receives the ground truth as input to the generator. This reduces the difference between training and inference, which could result in higher values of p_{gen}.

REFERENCES

[1] Mehdi Allahyari, Seyedamin Pouriyeh, Mehdi Assefi, Saeid Safaei, Elizabeth D Trippe, Juan B Gutierrez, and Krys Kochut. 2017. Text summarization techniques: a brief survey. *arXiv preprint arXiv:1707.02268* (2017).

[2] Tal Baumel, Matan Eyal, and Michael Elhadad. 2018. Query Focused Abstractive Summarization: Incorporating Query Relevance, Multi-Document Coverage, and Summary Length Constraints into seq2seq Models. *arXiv preprint arXiv:1801.07704* (2018).

[3] Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. 2015. Scheduled sampling for sequence prediction with recurrent neural networks. In *Advances in Neural Information Processing Systems*. 1171–1179.

[4] Shibhansh Dohare, Harish Karnick, and Vivek Gupta. 2017. Text summarization using abstract meaning representation. *arXiv preprint arXiv:1706.01678* (2017).

[5] John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. *J. Mach. Learn. Res.* 12 (July 2011), 2121–2159. http://dl.acm.org/citation.cfm?id=1953048.2021068

[6] Lisa Fan, Dong Yu, and Lu Wang. 2018. Robust Neural Abstractive Summarization Systems and Evaluation against Adversarial Information. *arXiv preprint arXiv:1810.06065* (2018).

[7] Jiatao Gu, Zhengdong Lu, Hang Li, and Victor OK Li. 2016. Incorporating copying mechanism in sequence-to-sequence learning. *arXiv preprint arXiv:1603.06393* (2016).

[8] Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*. 1693–1701.

[9] Eric Jang, Shixiang Gu, and Ben Poole. 2016. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144* (2016).

[10] Hongyan Jing. 2002. Using hidden Markov modeling to decompose human-written summaries. *Computational linguistics* 28, 4 (2002), 527–543.

[11] Diederik P. Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *CoRR* abs/1412.6980 (2014). arXiv:1412.6980 http://arxiv.org/abs/1412.6980

[12] Jacob Krantz and Jugal Kalita. 2018. Abstractive Summarization Using Attentive Neural Techniques. *arXiv preprint arXiv:1810.08838* (2018).

[13] Wojciech Kryściński, Romain Paulus, Caiming Xiong, and Richard Socher. 2018. Improving abstraction in text summarization. *arXiv preprint arXiv:1808.07913* (2018).

[14] Alex M Lamb, Anirudh Goyal ALIAS PARTH GOYAL, Ying Zhang, Saizheng Zhang, Aaron C Courville, and Yoshua Bengio. 2016. Professor forcing: A new algorithm for training recurrent networks. In *Advances In Neural Information Processing Systems*. 4601–4609.

[15] Piji Li, Wai Lam, Lidong Bing, and Zihao Wang. 2017. Deep recurrent generative decoder for abstractive text summarization. *arXiv preprint arXiv:1708.00625* (2017).

[16] Chin-Yew Lin. 2004. ROUGE: a Package for Automatic Evaluation of Summaries. https://www.microsoft.com/en-us/research/publication/rouge-a-package-for-automatic-evaluation-of-summaries/

[17] Ramesh Nallapati, Bowen Zhou, Caglar Gulcehre, Bing Xiang, et al. 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. *arXiv preprint arXiv:1602.06023* (2016).

[18] Abigail See, Peter J Liu, and Christopher D Manning. 2017. Get to the point: Summarization with pointer-generator networks. *arXiv preprint arXiv:1704.04368* (2017).

[19] Kaiqiang Song, Lin Zhao, and Fei Liu. 2018. Structure-Infused Copy Mechanisms for Abstractive Summarization. *arXiv preprint arXiv:1806.05658* (2018).

[20] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research* 15, 1 (2014), 1929–1958.

[21] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*. 3104–3112.

[22] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*. 5998–6008.

[23] Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. 2015. Pointer networks. In *Advances in Neural Information Processing Systems*. 2692–2700.

[24] Noah Weber, Leena Shekhar, Niranjan Balasubramanian, and Kyunghyun Cho. 2018. Controlling Decoding for More Abstractive Summaries with Copy-Based Networks. *arXiv preprint arXiv:1803.07038* (2018).

[25] Frank Wilcoxon. 1945. Individual comparisons by ranking methods. *Biometrics bulletin* 1, 6 (1945), 80–83.

Appendices

A SCORES WITHOUT COVERAGE

Table 6: Mean rouge F_1 and VERT scores of the tested models (11k examples in testset). All models were trained without coverage. Here NLoss corresponds to naive pointing loss and WPLoss to the word prior pointing loss.

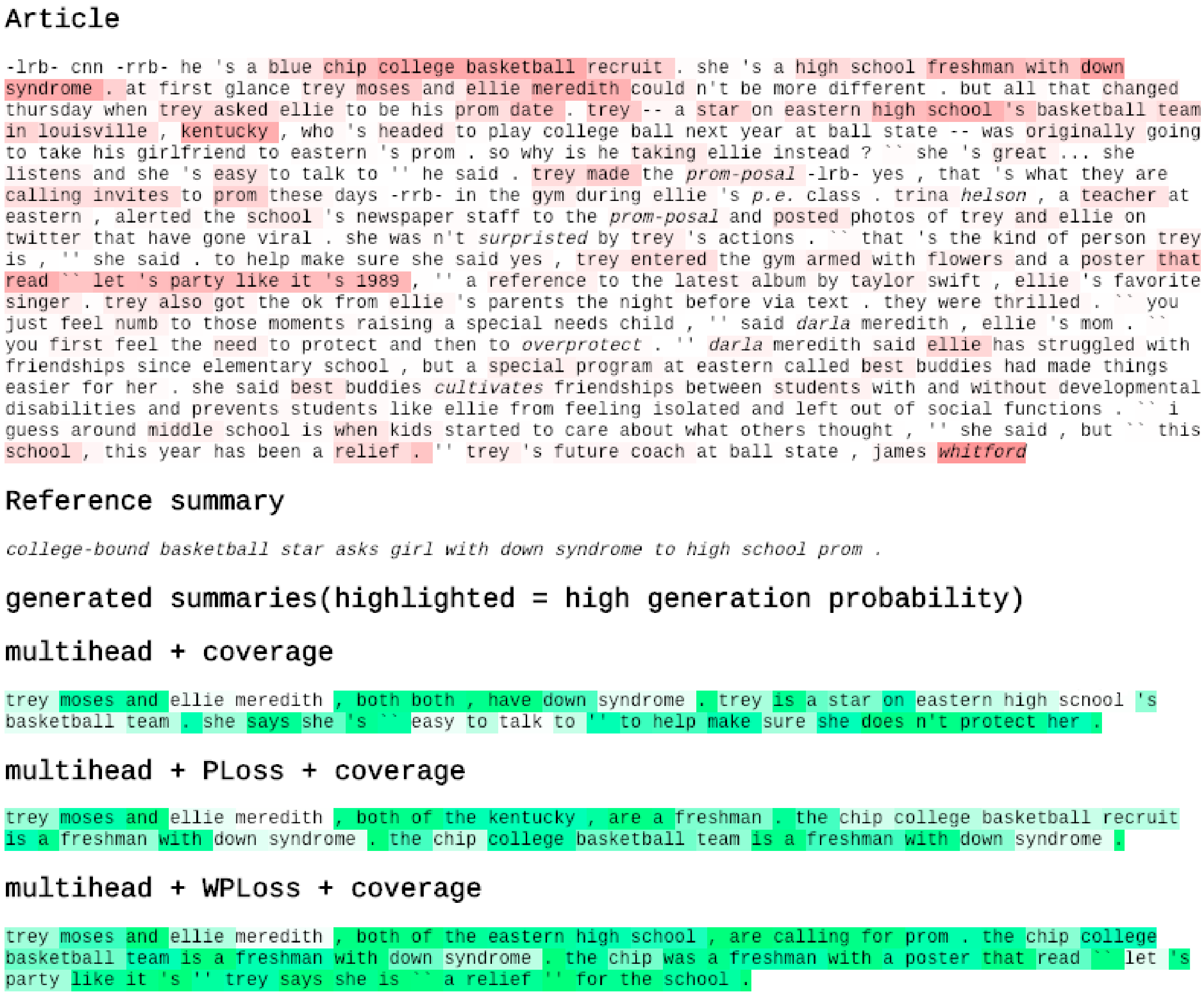
Attention heads	Model extensions		Rouge 1	Rouge 2	Rouge L	VERT
1			36.65	15.30	31.53	0.691
4			36.50	15.15	31.43	0.691
1	dropout		36.37	15.20	31.13	0.688
4	dropout		36.27	15.11	31.19	0.689
1		NLoss	35.87	14.96	30.81	0.692
4		NLoss	35.55	14.58	30.37	0.683
1		WPLoss	35.44	14.49	30.13	0.680
4		WPLoss	35.12	14.26	30.01	0.678
1 (*baseline See et al. [18]*)			36.44	15.66	34.42	-

Table 7: Percentage of novel N-grams and sentences that are produced for each of the tested models.

Attention heads	Model extensions		1-grams	2-grams	3-grams	4-grams	Sentences
1			0.36	4.31	10.51	16.25	82.50
4			0.31	3.97	9.84	15.41	81.40
1	Dropout		0.28	4.28	10.74	16.80	83.31
4	Dropout		0.60	4.37	10.42	16.08	82.66
1		NLoss	0.28	5.11	12.68	19.29	86.54
4		NLoss	0.31	5.28	12.87	19.73	87.22
1		WPLoss	1.43	8.90	18.49	26.46	92.19
4		WPLoss	0.90	8.19	18.40	26.97	91.35
Target summaries			16.95	52.48	72.36	81.94	98.97

B EXAMPLES

The following pages contain randomly chosen examples from the Multihead model with coverage, combined with the new losses. The words highlighted in red reflect the overall attention the model paid to a word during the constructing the summary. Italic words denote out-of-vocabulary words. The green shading intensity represents the value of the generation probability p_gen.



Article

-lrb- cnn -rrb- he 's a blue chip college basketball recruit . she 's a high school freshman with down syndrome . at first glance trey moses and ellie meredith could n't be more different . but all that changed thursday when trey asked ellie to be his prom date . trey -- a star on eastern high school 's basketball team in louisville , kentucky , who 's headed to play college ball next year at ball state -- was originally going to take his girlfriend to eastern 's prom . so why is he taking ellie instead ? `` she 's great ... she listens and she 's easy to talk to '' he said . trey made the *prom-posal* -lrb- yes , that 's what they are calling invites to prom these days -rrb- in the gym during ellie 's *p.e.* class . trina *helson* , a teacher at eastern , alerted the school 's newspaper staff to the *prom-posal* and posted photos of trey and ellie on twitter that have gone viral . she was n't *surpristed* by trey 's actions . `` that 's the kind of person trey is , '' she said . to help make sure she said yes , trey entered the gym armed with flowers and a poster that read `` let 's party like it 's 1989 , '' a reference to the latest album by taylor swift , ellie 's favorite singer . trey also got the ok from ellie 's parents the night before via text . they were thrilled . `` you just feel numb to those moments raising a special needs child , '' said *darla* meredith , ellie 's mom . `` you first feel the need to protect and then to *overprotect* . '' *darla* meredith said ellie has struggled with friendships since elementary school , but a special program at eastern called best buddies had made things easier for her . she said best buddies *cultivates* friendships between students with and without developmental disabilities and prevents students like ellie from feeling isolated and left out of social functions . `` i guess around middle school is when kids started to care about what others thought , '' she said , but `` this school , this year has been a relief . '' trey 's future coach at ball state , james *whitford*

Reference summary

college-bound basketball star asks girl with down syndrome to high school prom .

generated summaries(highlighted = high generation probability)

multihead + coverage

trey moses and ellie meredith , both both , have down syndrome . trey is a star on eastern high school 's basketball team . she says she 's `` easy to talk to '' to help make sure she does n't protect her .

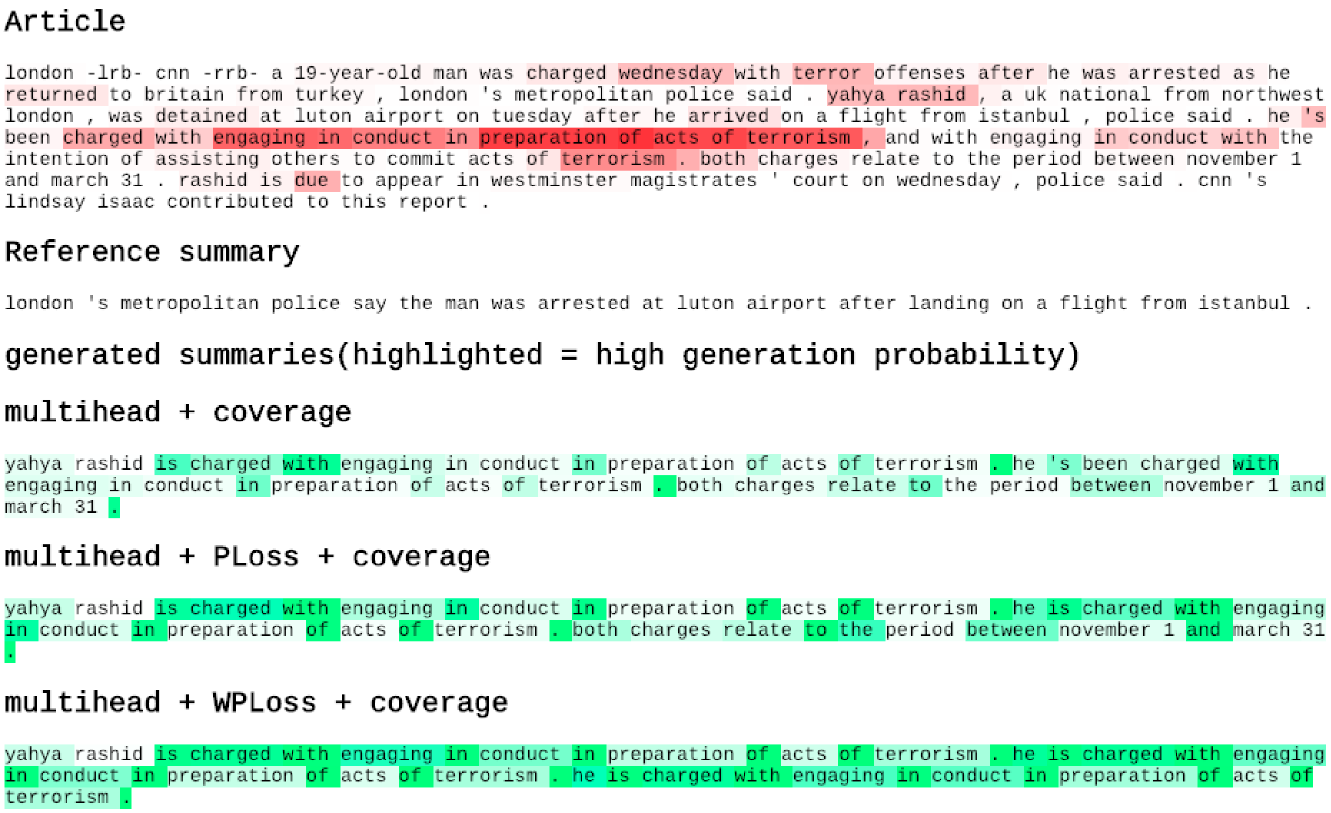
multihead + PLoss + coverage

trey moses and ellie meredith , both of the kentucky , are a freshman . the chip college basketball recruit is a freshman with down syndrome . the chip college basketball team is a freshman with down syndrome .

multihead + WPLoss + coverage

trey moses and ellie meredith , both of the eastern high school , are calling for prom . the chip college basketball team is a freshman with down syndrome . the chip was a freshman with a poster that read `` let 's party like it 's '' trey says she is `` a relief '' for the school .

Figure 3: In this example, the average p_{gen} is much higher in the word prior loss model, except on words with a low prior ('ellie meredith', 'down syndrome').



Article

london -lrb- cnn -rrb- a 19-year-old man was charged wednesday with terror offenses after he was arrested as he returned to britain from turkey , london 's metropolitan police said . yahya rashid , a uk national from northwest london , was detained at luton airport on tuesday after he arrived on a flight from istanbul , police said . he 's been charged with engaging in conduct in preparation of acts of terrorism , and with engaging in conduct with the intention of assisting others to commit acts of terrorism . both charges relate to the period between november 1 and march 31 . rashid is due to appear in westminster magistrates ' court on wednesday , police said . cnn 's lindsay isaac contributed to this report .

Reference summary

london 's metropolitan police say the man was arrested at luton airport after landing on a flight from istanbul .

generated summaries(highlighted = high generation probability)

multihead + coverage

yahya rashid is charged with engaging in conduct in preparation of acts of terrorism . he 's been charged with engaging in conduct in preparation of acts of terrorism . both charges relate to the period between november 1 and march 31 .

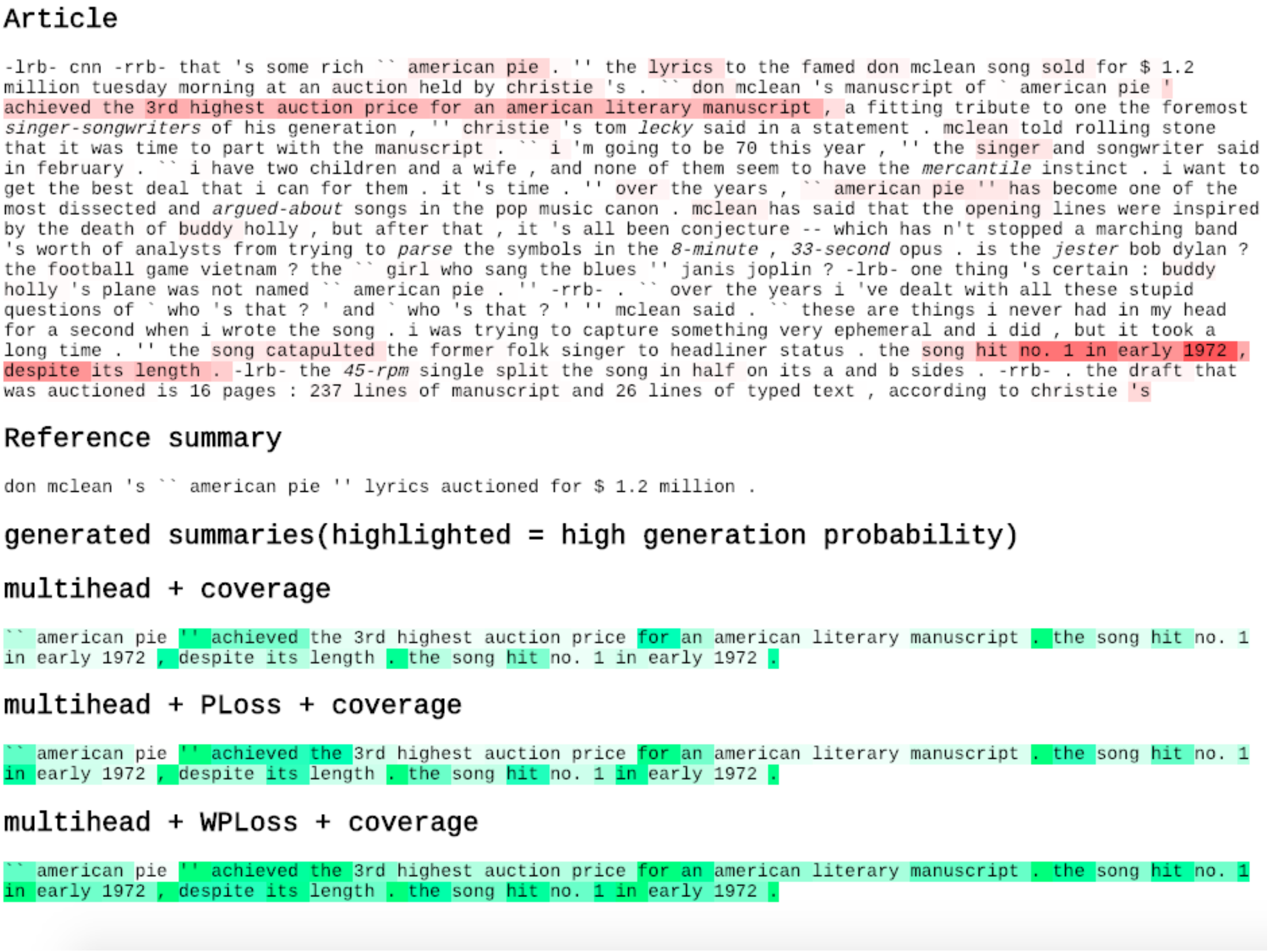
multihead + PLoss + coverage

yahya rashid is charged with engaging in conduct in preparation of acts of terrorism . he is charged with engaging in conduct in preparation of acts of terrorism . both charges relate to the period between november 1 and march 31 .

multihead + WPLoss + coverage

yahya rashid is charged with engaging in conduct in preparation of acts of terrorism . he is charged with engaging in conduct in preparation of acts of terrorism . he is charged with engaging in conduct in preparation of acts of terrorism .

Figure 4: This example shows clearly that the new losses re-introduce the over generation problems that the coverage loss aimed to solve.



Article

-lrb- cnn -rrb- that 's some rich `` american pie . '' the lyrics to the famed don mclean song sold for $ 1.2 million tuesday morning at an auction held by christie 's . `` don mclean 's manuscript of ` american pie ' achieved the 3rd highest auction price for an american literary manuscript , a fitting tribute to one the foremost *singer-songwriters* of his generation , '' christie 's tom *lecky* said in a statement . mclean told rolling stone that it was time to part with the manuscript . `` i 'm going to be 70 this year , '' the singer and songwriter said in february . `` i have two children and a wife , and none of them seem to have the *mercantile* instinct . i want to get the best deal that i can for them . it 's time . '' over the years , `` american pie '' has become one of the most dissected and *argued-about* songs in the pop music canon . mclean has said that the opening lines were inspired by the death of buddy holly , but after that , it 's all been conjecture -- which has n't stopped a marching band 's worth of analysts from trying to *parse* the symbols in the *8-minute* , *33-second* opus . is the *jester* bob dylan ? the football game vietnam ? the `` girl who sang the blues '' janis joplin ? -lrb- one thing 's certain : buddy holly 's plane was not named `` american pie . '' -rrb- . `` over the years i 've dealt with all these stupid questions of ` who 's that ? ' and ` who 's that ? ' '' mclean said . `` these are things i never had in my head for a second when i wrote the song . i was trying to capture something very ephemeral and i did , but it took a long time . '' the song catapulted the former folk singer to headliner status . the song hit no. 1 in early 1972 , despite its length . -lrb- the *45-rpm* single split the song in half on its a and b sides . -rrb- . the draft that was auctioned is 16 pages : 237 lines of manuscript and 26 lines of typed text , according to christie 's

Reference summary

don mclean 's `` american pie '' lyrics auctioned for $ 1.2 million .

generated summaries(highlighted = high generation probability)

multihead + coverage

`` american pie '' achieved the 3rd highest auction price for an american literary manuscript . the song hit no. 1 in early 1972 , despite its length . the song hit no. 1 in early 1972 .

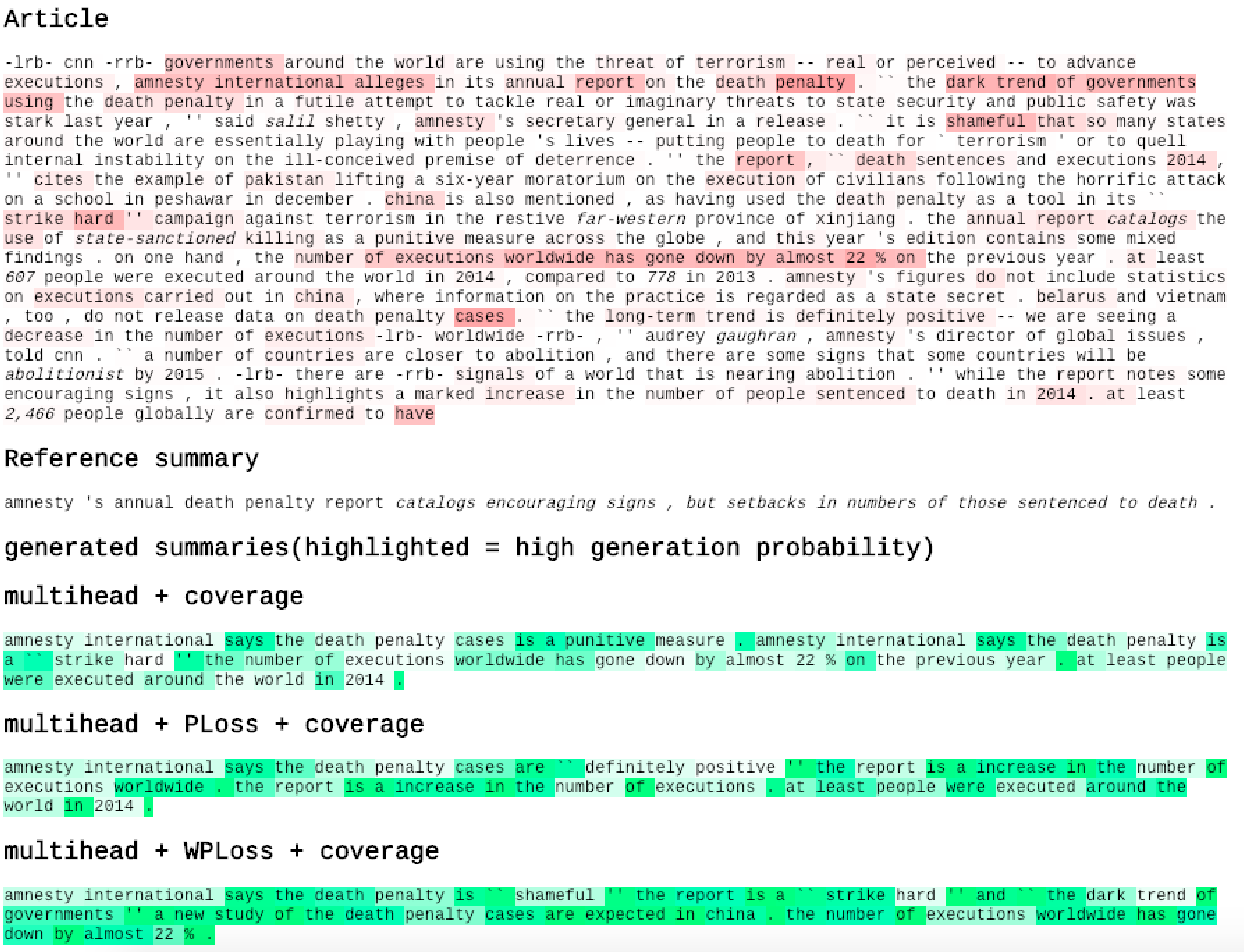
multihead + PLoss + coverage

`` american pie '' achieved the 3rd highest auction price for an american literary manuscript . the song hit no. 1 in early 1972 , despite its length . the song hit no. 1 in early 1972 .

multihead + WPLoss + coverage

`` american pie '' achieved the 3rd highest auction price for an american literary manuscript . the song hit no. 1 in early 1972 , despite its length . the song hit no. 1 in early 1972 .

Figure 5: Another example. While the average p_{gen} of each word is significantly higher in the WPLoss model, the generated summaries are the same.



Article

-lrb- cnn -rrb- governments around the world are using the threat of terrorism -- real or perceived -- to advance executions , amnesty international alleges in its annual report on the death penalty . `` the dark trend of governments using the death penalty in a futile attempt to tackle real or imaginary threats to state security and public safety was stark last year , '' said *salil* shetty , amnesty 's secretary general in a release . `` it is shameful that so many states around the world are essentially playing with people 's lives -- putting people to death for ` terrorism ' or to quell internal instability on the ill-conceived premise of deterrence . '' the report , `` death sentences and executions 2014 , '' cites the example of pakistan lifting a six-year moratorium on the execution of civilians following the horrific attack on a school in peshawar in december . china is also mentioned , as having used the death penalty as a tool in its `` strike hard '' campaign against terrorism in the restive *far-western* province of xinjiang . the annual report *catalogs* the use of *state-sanctioned* killing as a punitive measure across the globe , and this year 's edition contains some mixed findings . on one hand , the number of executions worldwide has gone down by almost 22 % on the previous year . at least *607* people were executed around the world in 2014 , compared to *778* in 2013 . amnesty 's figures do not include statistics on executions carried out in china , where information on the practice is regarded as a state secret . belarus and vietnam , too , do not release data on death penalty cases . `` the long-term trend is definitely positive -- we are seeing a decrease in the number of executions -lrb- worldwide -rrb- , '' audrey *gaughran* , amnesty 's director of global issues , told cnn . `` a number of countries are closer to abolition , and there are some signs that some countries will be *abolitionist* by 2015 . -lrb- there are -rrb- signals of a world that is nearing abolition . '' while the report notes some encouraging signs , it also highlights a marked increase in the number of people sentenced to death in 2014 . at least *2,466* people globally are confirmed to have

Reference summary

amnesty 's annual death penalty report *catalogs encouraging signs , but setbacks in numbers of those sentenced to death .*

generated summaries(highlighted = high generation probability)

multihead + coverage

amnesty international says the death penalty cases is a punitive measure . amnesty international says the death penalty is a `` strike hard '' the number of executions worldwide has gone down by almost 22 % on the previous year . at least people were executed around the world in 2014 .

multihead + PLoss + coverage

amnesty international says the death penalty cases are `` definitely positive '' the report is a increase in the number of executions worldwide . the report is a increase in the number of executions . at least people were executed around the world in 2014 .

multihead + WPLoss + coverage

amnesty international says the death penalty is `` shameful '' the report is a `` strike hard '' and `` the dark trend of governments '' a new study of the death penalty cases are expected in china . the number of executions worldwide has gone down by almost 22 % .

Figure 6: The last sentence in the WPLoss example on average mostly uses the generator, but is an exact copy of line 11 in the source article.