Journal of Artificial Intelligence Research 71 (2021) 371-399 Submitted 11/2020; published 07/2021

Multi-Document Summarization with Determinantal Point Process Attention

Laura Perez-Beltrachini LPEREZ@ED.AC.UK
Mirella Lapata MLAP@ED.AC.UK
Institute for Language, Cognition and Computation,
School of Informatics, University of Edinburgh,
10 Crichton Street, Edinburgh, Scotland

Abstract

The ability to convey relevant and diverse information is critical in multi-document summarization and yet remains elusive for neural seq-to-seq models whose outputs are often redundant and fail to correctly cover important details. In this work, we propose an attention mechanism which encourages greater focus on *relevance* and *diversity*. Attention weights are computed based on (proportional) probabilities given by Determinantal Point Processes (DPPs) defined on the set of content units to be summarized. DPPs have been successfully used in extractive summarisation, here we use them to select relevant and diverse content for neural abstractive summarisation. We integrate DPP-based attention with various seq-to-seq architectures ranging from CNNs to LSTMs, and Transformers. Experimental evaluation shows that our attention mechanism consistently improves summarization and delivers performance comparable with the state-of-the-art on the MultiNews dataset.

1. Introduction

Text summarization has achieved significant progress in recent years thanks to neural encoder-decoder models (Bahdanau, Cho, & Bengio, 2014; Sutskever, Vinyals, & Le, 2014) and their ability to produce highly fluent texts. In this formulation, source content is encoded with a neural architecture, while the decoder autoregressively produces a token at each output position based on (a) its internal state, (b) attention mechanisms (Bahdanau et al., 2014; Luong, Pham, & Manning, 2015) focusing on relevant parts of the input (which in theory should be different at different time steps), and (c) the representation of the source.

Despite recent success, neural summarization models are prone to hallucination, i.e., generating text that does not preserve the meaning of the input (Song, Zhao, & Liu, 2018a), repetition and redundancy (Li, Xiao, Lyu, & Wang, 2018; Suzuki & Nagata, 2017), and often struggle to identify which content units are salient and should be summarized (Tan, Wan, & Xiao, 2017a). These phenomena are amplified when generating multi-document summaries from clusters of thematically related texts which are not only long but also naturally redundant (Liu & Lapata, 2019; Perez-Beltrachini, Liu, & Lapata, 2019). Especially in this setting, we argue that making each decoding step explicitly aware of previously focused content units and encouraging diversity with respect to these will lead to more informative summaries with less repetition.

The majority of previous work has focused on the encoder module and how to improve the input document representations. Some models encode documents hierarchically (Celikyilmaz, Bosselut, He, & Choi, 2018; Liu & Lapata, 2019), while others enrich representations with topic features (Narayan, Cohen, & Lapata, 2018), model graph connections between document elements to better capture salience (Tan, Wan, & Xiao, 2017b; Liu & Lapata, 2019), or use information from a separate content selection step to decide on the relevant aspects of the input (Gehrmann, Deng, & Rush, 2018). In contrast, we focus on the decoder module and propose a mechanism based on Determinantal Point Processes (DPPs, Macchi, 1975; Borodin, 2009) which at each time step encourages attention on a subset of input representations which are both relevant and diverse.

DPPs define a probability distribution over all possible subsets of a set. This probability is captured in terms of dissimilarities and thus subsets with relevant and diverse elements are assigned higher probability. Kulesza and Taskar (2012) applied DPPs to *extractive* **M**ulti-**D**ocument **S**ummarization (MDS), under the assumption that a cluster of documents constitutes a set of sentences. Specifically, they defined a decomposition in which subset probabilities are computed based on *diversity* (aka dissimilarity between a pair of sentences) and *relevance* (aka importance of sentences to the summary). In this work, we use DPPs to model content subset selection in *abstractive* MDS. We define the decoder-encoder attention as a DPP over the set of input content units. The proposed mechanism is not architecture specific and can be flexibly integrated with various sequence-to-sequence models ranging from Recurrent Neural Networks (RNNs) to Convolutional Neural Networks (CNNs, Gehring, Auli, Grangier, Yarats, & Dauphin, 2017) and Transformers (Vaswani, Shazeer, Parmar, Uszkoreit, Jones, Gomez, Kaiser, & Polosukhin, 2017).

To track which content has been covered, some neural approaches impose constraints on the attention mechanism based on the hypothesis that repeatedly attending to the same input positions will promote redundancy in the output. For example, Nallapati, Zhou, Gulcehre, Xiang, et al. (2016) introduce a temporal attention mechanism where attention scores are decreased at each step by scores accumulated in previous steps. See, Liu, and Manning (2017) feed attention weights from previous steps into the computation of the current attention step and use an additional loss term penalizing similar subsequent attention distributions. However, there are some important differences. First, our approach *explicitly* models of diversity. At each step, our attention mechanism is influenced by a past memory (Chun & Turk-Browne, 2007), i.e., the subset of content units attended so far, and computes diversity weights for the input in order to decide where to focus next. In other words, it is not only able to discard already selected content but also knows which content elements to select next on account of their diversity. Second, instead of accumulating attention weights at different input positions, the past memory encodes a content representation of the input elements attended so far. In particular, in a Multi-Document Summarisation (MDS) scenario where content can be repeated in different source documents, the DPP mechanism is more effective in promoting content diversity.

We evaluate our approach on two large-scale datasets which pose different challenges for abstractive MDS. WikiCatSum (Perez-Beltrachini et al., 2019) is an automatically constructed dataset which consists of Wikipedia abstracts, each corresponding to a set of webpages related to an entity (e.g., a film or animal), while MultiNews (Fabbri, Li, She, Li, & Radev, 2019) consists of professionally written summaries, each corresponding to two or

more news articles covering the same topic (e.g., the nutritional value of chicken nuggets). In WikiCatSum, the input webpages are very noisy, the training set is loosely aligned, and the summaries are short, while in MultiNews summaries are very long, and the input news articles considerably less noisy and possibly more redundant. We experimentally show that our mechanism helps with content selection across datasets and neural architectures.

Our contributions in this work are three-fold: we propose a novel abstractive MDS model based on DPP attention; we integrate this mechanism into different neural architectures and show that it brings consistent improvements across all of them; and we report state-of-the-art results on the MultiNews dataset.[1]

2. Related Work

Multi-Document Summarization Most previous solutions to multi-document summarization adopt non-neural, extractive methods (Carbonell & Goldstein, 1998; Radev, Jing, Styś, & Tam, 2004; Erkan & Radev, 2004; Barzilay, McKeown, & Elhadad, 1999).

More recently, various encoder-decoder architectures (Liu & Lapata, 2019; Fabbri et al., 2019; Perez-Beltrachini et al., 2019; Liu, Saleh, Pot, Goodrich, Sepassi, Kaiser, & Shazeer, 2018; Zhang, Tan, & Wan, 2018; Lebanoff, Song, & Liu, 2018) have been ported to this task thanks to the development of large-scale datasets for model training. Among these, two approaches are closely related to our work on account of handling redundancy explicitly. Lebanoff et al. (2018) first pre-train an abstractive summarization model on single-document data and then fine-tune it on smaller multi-document benchmarks. They use a separately trained Maximal Marginal Relevance (MMR, Carbonell & Goldstein, 1998) module to select a relevant and non-redundant sentence from the input documents for the generation of the next summary sentence. Fabbri et al. (2019) incorporate this MMR mechanism as hierarchical attention into an end-to-end trained Pointer-Generator network (See et al., 2017).

Our proposal differs from both approaches in terms of granularity; they operate at the *sentence* level while our model operates at the *word* level, resembling more the phrase selection approach of Barzilay et al. (1999). Another important difference lies in the way previously selected content is modelled. Lebanoff et al. (2018) explicitly select distinct sentences from the input, while Fabbri et al. (2019) do not track previously selected content and compute diversity as self-attention among *all* source sentences at each time step. In contrast, our DPP guided attention computes diversity at each time step between input tokens and a *summary* of *previous* context decisions.

Coverage and Redundancy Coverage and redundancy have been previously handled by accumulating attention scores through decoding steps and using these to decrease the attention scores of subsequent steps (Nallapati et al., 2016; See et al., 2017). Paulus, Xiong, and Socher (2017) propose an additional intra-attention mechanism to instil information about previous decisions. We do not keep past attention scores but rather compute diversity at each decoding step with respect to a previous decoding context. Gehrmann et al. (2018) propose a summary coverage penalty as a hard constraint at decoding time. The intuition behind this is to penalize candidate hypotheses whenever the decoder directs a large propor-

1. Our code and data are available at `https://github.com/lauhaide/dppattn`

tion of the attention towards only a few tokens therein. Benmalek, Khabsa, Desu, Cardie, and Banko (2019) keep track of what has been generated so far by directly modifying the encoder's hidden states. Their solution assumes the underlying model is trained step-by-step and is therefore not suitable for non-recurrent approaches where training is carried out in parallel. In contrast, our attention mechanism can be added to Convolutional and Transformer-based architectures.

Determinantal Point Processes Kulesza and Taskar (2012) were the first to apply DPPs to extractive multi-document summarization, viewing it as a subset selection task. They provide a DPP decomposition defined in terms of relevance (i.e., a parametrized feature-based function) and diversity (i.e., cosine similarity over sentence level TF-IDF feature vectors). Yao, Fan, Zhao, Wan, Chang, and Xiao (2016) use DPPs for Twitter timeline generation which they again conceptualize as a subset selection problem. Cho, Lebanoff, Foroosh, and Liu (2019a) propose an improved similarity function based on capsule networks for extractive summarization, while Cho, Li, Yu, Foroosh, and Liu (2019b) fine-tune a BERT model to learn representations of sentence similarity and importance and then use these to extract sentences with DPP. More recently, Cho, Song, Li, Yu, Foroosh, and Liu (2020) applied DPP to highlight a subset of important and non-redundant text segments in a multi-document input. These approaches use DPP inference to extract a concrete subset of elements, i.e., sentences or tweets. In our formulation, the notion of subset selection is soft, DPPs influence which elements to select from the input representation created by the encoder but we still compute a distribution of scores over the entire input set.

DPPs have also been used recently to improve neural text generation. Song, Yan, Feng, Zhang, Zhao, and Zhang (2018b) define a DPP over decoder representations for diverse word prediction and hypothesis re-ranking and apply their model to response generation where the generated text is shorter and arguably less constrained than summarization. Li, Liu, Litvak, Vanetik, and Huang (2019) use DPPs to adjust the attention distribution within a convolutional model applied to single-document summarization. They compute diversity scores among the input elements (e.g., word representations) and include a loss term to optimize the DPP scores of sets of attention weights. Our approach is conceptually simpler and more general, we compute attention scores by applying DPPs at each time step.

Explainability Previous work has studied the role of attention distributions in explaining model decisions in text classification tasks (Jain & Wallace, 2019; Wiegreffe & Pinter, 2019). Although model predictions cannot be attributed to attention weights (Jain & Wallace, 2019), Wiegreffe and Pinter (2019) empirically show that these capture some notion of token importance. Indeed, in sequence-to-sequence modelling, attention distributions have been used to provide token saliency, e.g., copy and coverage mechanisms. Our DPP attention mechanism adds some degree of interpretability by the way it constructs attention weights. Along the lines of hard and sparse attention mechanisms (Jain & Wallace, 2019) where explicit decisions are made, our mechanism will explicitly downgrade input units similar in content to previous selected contents.

3. Problem Formulation

We formulate the abstractive MDS task as a sequence-to-sequence generation problem (Liu et al., 2018; Fabbri et al., 2019) where both the document cluster $\mathcal{D} = \{d_1 \cdots d_m\}$ and the corresponding multi-sentence summary are represented as a single sequence (documents are separated by a special token). Given input sequence $\mathcal{X} = (x_{11} \cdots x_{|d_1|1}$ [SEP] $\cdots$ [SEP] $x_{1m} \cdots x_{|d_m|m})$ where x_{ij} is the i-th token from the j-th input document, our goal is to generate a summary consisting of a sequence of tokens $\mathcal{Y} = (y_1, \cdots, y_{|\mathcal{Y}|})$ where $P(\mathcal{Y}|\mathcal{X}) = \prod_{t=1}^{|\mathcal{Y}|} P(y_t|y_{1:t-1}, \mathcal{X})$.

3.1 Determinantal Point Process

Let $\mathcal{S} = \{1, \cdots, |\mathcal{S}|\}$ denote a finite set of elements. A determinantal (discrete) point process is a probability distribution $\mathcal{P}$ on all subsets $2^{\mathcal{S}}$ of $\mathcal{S}$; it is defined in terms of a positive semi-definite kernel matrix L (i.e., all determinants are ≥ 0) indexed by the elements of $\mathcal{S}$ (Borodin, 2009), such that if $\mathbf{S}$ is a random set drawn according to $\mathcal{P}$, the probability of $S \subseteq \mathcal{S}$ being this set is given by:

$$\mathcal{P}(S = \mathbf{S}; L) = \frac{\det(L_S)}{\det(L + I)} \tag{1}$$

$$\det(L + I) = \sum_{S \subseteq \mathcal{S}} \det(L_S) \tag{2}$$

where $L \in \mathbb{R}^{|\mathcal{X}| \times |\mathcal{X}|}$, I is the identity matrix, $\det(\cdot)$ is the determinant of a matrix and L_S is the submatrix formed by taking from L the entries corresponding to elements in S. $\mathcal{P}(S = \mathbf{S}; L)$ is proportional to the determinant of L_S. The (proportional) probability of a set of two elements $S = \{i, j\}$ is:

$$\mathcal{P}(\{i, j\}; L) \propto \det(L_{\{i,j\}}) = L_{ii} L_{jj} - L_{ij}^2 \tag{3}$$

where intuitively, the probability of a set of items increases with their relevance but decreases according to their similarity.

Given a DPP defined by $\mathcal{P}(\cdot; L)$, we can select the subset S with highest probability (S^{MAP}). In MDS, the creation of a summary involves selecting the subset of content units from the input documents which is maximally relevant and diverse. While this search problem is NP-hard, it can be approximated by greedy selection. Kulesza and Taskar (2012) obtain extractive summaries by greedily incorporating input sentences up to a budget. In our encoder-decoder setting, the set of content units $\mathcal{S}$ is $\mathcal{X}$, namely the set of input word representations produced by the encoder. And the decoder incrementally attends to a subset $X \subseteq \mathcal{X}$ that is relevant and diverse for generating summary $\mathcal{Y}$. Note that in this setting the selection of subset X is naturally incremental as content items are considered at each decoding step.

There are different ways to define the kernel matrix L (Kulesza & Taskar, 2010; Song et al., 2018b; Mariet, Ovadia, & Snoek, 2019). We adopt the decomposition proposed by Kulesza and Taskar (2010) which assumes that L is a Gram matrix:

$$L_{ij} = q_i \phi_i^T \phi_j q_j \tag{4}$$

$$\mathcal{P}(\{i, j\}; L) \propto q_i^2 q_j^2 (1 - (\phi_i^T \phi_j)^2) \tag{5}$$

where $q_i \in \mathbb{R}^+$ measures the *relevance* of element i, ϕ_i is a feature mapping over element i and $\phi_i^T \phi_j$ measures the *similarity* between elements i and j, and $\|\phi_i\|_2 = 1$ so $\phi_i^T \phi_j \in [-1, 1]$. This decomposition permits to model relevance and dissimilarity independently. We then define the L matrix in terms of the representations created by the encoder and decoder networks.

3.2 DPP Attention

We first explain how attention can be redefined as DPP-based soft selection within a recurrent encoder-decoder architecture, e.g. LSTM (Hochreiter & Schmidhuber, 1997), and then illustrate how it can be incorporated into non-recurrent architectures like CNNs and Transformers.

Recurrent Architectures Given an input document sequence $\mathcal{X}$, the encoder yields representations $(\mathbf{z}_1, \cdots, \mathbf{z}_{|\mathcal{X}|})$, while the decoder generates target summary $\mathcal{Y}$ token-by-token. The prediction of token y_t is based on the decoder hidden state $\mathbf{h}_t$ and the input context vector $\mathbf{c}_t$ at time step t:

$$\begin{aligned} P(y_t|y_{1:t-1}, \mathcal{X}) &= \text{softmax}(g(f(\mathbf{h}_t, \mathbf{c}_t))) \\ \mathbf{h}_t &= \text{LSTM}(\mathbf{h}_{t-1}, f(\mathbf{h}_{t-1}, \mathbf{c}_{t-1})) \end{aligned} \tag{6}$$

where $g(\cdot)$ is a neural network with one hidden layer parametrized by $W_o \in \mathbb{R}^{d \times |V|}$ ($|V|$ is the size of the output vocabulary and d the dimensionality of the hidden units), over a composition f of $\mathbf{h}_t$ and $\mathbf{c}_t$ (e.g., concatenation followed by a linear layer). Context vector $\mathbf{c}_t$ is a relevance-based weighted average of the input $\mathbf{c}_t = \sum_{j=1}^{|\mathcal{X}|} \alpha_{tj}\, \mathbf{z}_j$; relevance weights α_{tj} for input element x_j at time step t can be computed by a soft selection mechanism (Luong et al., 2015):

$$\alpha_{tj} = \frac{\exp(\text{score}(\mathbf{h}_t, \mathbf{z}_j))}{\sum_{j'} \exp(\text{score}(\mathbf{h}_t, \mathbf{z}_{j'}))} \tag{7}$$

In neural abstractive MDS, the decoder should ideally focus on a subset of relevant and diverse input representations to generate the summary. Existing architectures model this through the decoder states $\mathbf{h}_t$ (Equation (6)) which encode the summary subsequence up to step t and a relevance-based soft selection mechanism like the one shown in Equation (7). While $\mathbf{h}_t$ *implicitly* encodes which content should be attended to next, we incorporate diversity in the content selection process *explicitly* through the use of DPPs.

As shown in Figure 1, the weights ω_{tj} used to read the input (blue square vectors) depend on relevance (yellow square vectors) *and* diversity (green square vectors) values. The latter are computed between each input token x_j and the subset of tokens X_t (attended so far), by means of their representations $\mathbf{z}_j$ and context vector $\mathbf{Z}_t$. Note that this selection process is soft as $\mathbf{Z}_t$ does not contain the selected items themselves but their aggregate. At time t, the decoder attends to items from the input that are diverse with respect to $\mathbf{Z}_t$ and aggregates these into a new representation $\mathbf{Z}_{t+1}$.

Algorithm 1 illustrates greedy subset selection for DPP inference (Kulesza & Taskar, 2012; Yao et al., 2016). We compute DPP scores for all subsets of size two formed by the aggregate subset X_t and each input element x_j. These DPP scores are given by the

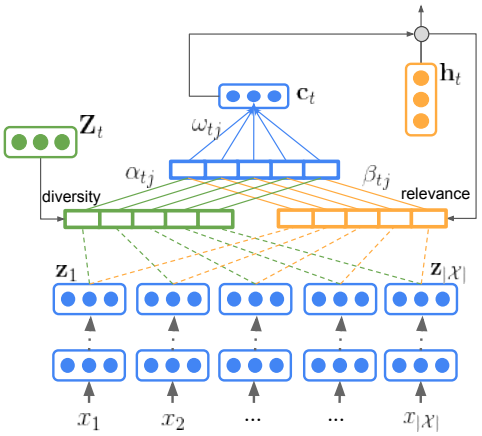


Figure 1: Encoder-Decoder DPP attention with diversity and relevance.

Algorithm 1 Approximate Computation of X^{MAP}

Input: Document content units $\mathcal{X} = \{1, \cdots, |\mathcal{X}|\}$
Input: Kernel matrix $L_{|\mathcal{X}|\times|\mathcal{X}|}$
Output: Selected content units X
1: $t \leftarrow 0$, $X_t \leftarrow \varnothing$
2: **repeat**
3: $\quad X_{t+1} \leftarrow X_t \cup \{\text{argmax}_{x_j \in \mathcal{X} \setminus X_t} \det(L_{(X_t \cup \{x_j\})})\}$
4: **until** *condition* (e.g., summary budget)
5: **return** X

determinant of a 2×2 matrix, $\det(L_{\{X_t, x_j\}})$. In practice, we do not need to explicitly create the entire kernel matrix L but just the necessary entries to compute pair-wise DPPs.

We construct L by defining the similarity component $\phi_t^T \phi_j$ as the cosine similarity between encoder representations $\phi_t = \mathbf{Z}_t$ for X_t and $\phi_j = \mathbf{z}_j$ for x_j and compute β_{tj}, the diversity score for the j-th input element, as:

$$\beta_{tj} = \det(L_{\{X_t, x_j\}}) = 1 - cos(\mathbf{Z}_t, \mathbf{z}_j)^2. \tag{8}$$

DPP attention weights are a combination of relevance (α_{tj}) and diversity (β_{tj}) scores. For each token x_j, we apply Equation (5), where $\phi_t^T \phi_j = \beta_{tj}$ $q_j^2 = \alpha_{tj}$, and $q_t = 1$ (the context element $\mathbf{Z}_t$ is always relevant):

$$\omega_{tj} = \alpha_{tj} \cdot \beta_{tj} \tag{9}$$

$$\mathbf{c}'_t = \sum_{j=1}^{|\mathcal{X}|} \omega_{tj}\, \mathbf{z}_j \tag{10}$$

where $\mathbf{c}'_t$ replaces $\mathbf{c}_t$ from Equation (6). Note that at time $t = 0$ DPP attention weights are based on relevance only.

Finally, there is some flexibility on how to compute $\mathbf{Z}_t$, the aggregate representation of the content attended so far. It can be simply the average sum of past input context vectors $\mathbf{Z}_t = \frac{1}{M}\sum_1^{t-1} \mathbf{c}'_m$; or alternatively it is possible to take the decoder's output vectors into account $\mathbf{Z}_t = \frac{1}{M}\sum_1^{t-1} f(\mathbf{h}_m, \mathbf{c}'_m)$ which also encode previous context.

Non-recurrent Architectures The integration of DPP attention into non-recurrent architectures is relatively straightforward. DPP scores are again computed based on representations of the input, the content attended so far, and the current decoder state. Analogously to LSTMs, ω_{tj} (Equation (9)) is the combination of relevance α_{tj} (Equation (7)) and diversity scores β_{tj} (Equation (8)). However, in non-recurrent architectures not all information from previous steps is available at time t since computation is parallelized. So, a key difference lies in the way we compute $\mathbf{Z}_t$, the representation of the content focused so far, which

we define in terms of previous layer representations. Specifically, for each layer l, we set $\mathbf{Z}_t^l = \frac{1}{M}\sum_1^{t-1} \mathbf{c}_m'^{(l-1)}$. In the first decoder layer, attention weights are computed based on relevance only.

As an example consider the CNN architecture defined in Gehring et al. (2017) where the decoder consists of stacked convolutional blocks (Equations 11–12) with output vectors $\mathbf{o}_t^l$ and input context vector $\mathbf{c}_t'$ for layer l. Each decoder layer uses DPP attention as defined in Equations (13–16); we omit the layer superscript for sake of clarity:

$$(\mathbf{o}_1^l, \cdots, \mathbf{o}_n^l) = \text{Conv}((\mathbf{o'}_1^{l-1}, \cdots, \mathbf{o'}_n^{l-1})) \tag{11}$$

$$\mathbf{o'}_t^l = \mathbf{o}_t^l + \mathbf{c}_t'^l \tag{12}$$

$$\mathbf{d}_t = W_d\mathbf{o}_t + \mathbf{g}_t \tag{13}$$

$$\alpha_{tj} = \frac{\exp(\mathbf{d}_t \cdot \mathbf{z}_j)}{\sum_{j'} \exp(\mathbf{d}_t \cdot \mathbf{z}_{j'})} \tag{14}$$

$$\beta_{tj} = 1 - cos(\mathbf{Z}_t, \mathbf{z}_j)^2 \tag{15}$$

$$\omega_{tj} = \alpha_{tj} \cdot \beta_{tj} \qquad \mathbf{c}_t' = \sum_{j=1}^{|\mathcal{X}|} \omega_{tj}(\mathbf{z}_j + \mathbf{e}_j). \tag{16}$$

Relevance scores α_{tj} are computed as in Gehring et al. (2017) where $\mathbf{g}_t$ is the previous target token embedding. As before, each $\mathbf{z}_j$ is the output of the last encoder layer and $\mathbf{e}_j$ is the embedding for input token x_j.

DPP attention can be integrated in a similar way with the Transformer architecture (Vaswani et al., 2017) where each decoder layer consists of multi-head attention (MHAtt), feed-forward (FFN), and layer normalization (LN) blocks:

$$\tilde{\mathbf{h}}_{self}^l = \text{LN}(\mathbf{h}^{l-1} + \text{MHAtt}(\mathbf{h}^{l-1}, \mathbf{h}^{l-1})) \tag{17}$$

$$\tilde{\mathbf{h}}_{cont}^l = \text{LN}(\mathbf{h}_{self}^l + \text{MHAtt}^{\text{DPP}}(\mathbf{h}_{self}^l, \mathbf{z})) \tag{18}$$

$$\mathbf{h}^l = \text{LN}(\tilde{\mathbf{h}}_{cont}^l + \text{FFN}(\tilde{\mathbf{h}}_{cont}^l)) \tag{19}$$

The multi-head attention mechanism MHAtt$(\cdot)$ can be conceptualized as a soft-lookup function that operates on an associative array. For a given set of queries Q, the attention uses a (scaled dot-product) similarity function to compare each query with a set of keys K. The resulting similarities are normalized and used as weights to compute a context vector which is a weighted sum over a set of values V associated with the keys:

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V \tag{20}$$

where d_k is the dimensionality of the key. The representation of Q, K and V is split into segments of equal dimension giving rise to multiple heads on which the operation in Equation (20) can be applied. The output representations of each head are finally combined with a concatenation operation.

To explain the integration of DPP attention (MHAtt$^{\text{DPP}}$) in this architecture we assume only one head, but it is trivial to generalize to multiple heads. Queries are linear projections

of decoder hidden states while keys and values are linear projections of representations produced by the encoder. We therefore combine relevance scores given by $\alpha_{tj} = \text{softmax}(\frac{\mathbf{q}_t \cdot \mathbf{k}_j}{\sqrt{d_k}})$ with diversity scores β_{tj} as:

$$\beta_{tj} = 1 - cos(\mathbf{Z}_t, \mathbf{k}_j) \qquad \omega_{tj} = \alpha_{tj} \cdot \beta_{tj} \tag{21}$$

$$\mathbf{c}'_t = \sum_{j=1}^{|\mathcal{X}|} \omega_{tj} \mathbf{v}_j \ , \ \mathbf{p}_t = \sum_{j=1}^{|\mathcal{X}|} \omega_{tj} \mathbf{k}_j. \tag{22}$$

where diversity β_{tj} is computed over the set of keys from the input, $\mathbf{c}'_t$ is the input context vector, and $\mathbf{p}_t$ is a weighted representation on keys used to define context representation $\mathbf{Z}^l_t = \frac{1}{M} \sum_1^{t-1} \mathbf{p}^{l-1}_m$ for each layer l.

The models defined by any of the architectures presented above are trained to optimize the negative log likelihood $\mathcal{L}_{\mathcal{NLL}} = -\sum_{t=1}^{|\mathcal{Y}|} \log P(y_t | y_{1:t-1}, \mathcal{X})$.

4. Experimental Setup

Data We first perform various controlled experiments on the WikiCatSum dataset (Perez-Beltrachini et al., 2019). WiKiCatSum is derived from the WikiSum (Liu et al., 2018) dataset and contains Wikipedia lead sections paired with a set of documents either from the article's references section or crawled from the Web. In order to handle the long and noisy document input, paragraphs within a cluster are typically ranked by TF-IDF (Liu et al., 2018) or with a learned ranker module (Liu & Lapata, 2019). For our controlled experiments, we limit the size of the input sequence to $L = 500$ containing the best document paragraphs according to an oracle which ranks paragraphs with ROUGE-2 against the article lead sections.

We then move to a more realistic setting and evaluate our approach on the MultiNews dataset (Fabbri et al., 2019) which contains manually written summaries for multiple news articles. Summaries in this dataset are rather long (263.66 tokens on average), however the input documents are considerably cleaner compared to WikiCatSum. We use the MultiNews data as preprocessed in Fabbri et al. (2019). Table 1 shows statistics for both datasets.

Comparison Systems Our experiments evaluate DPP attention across different sequence-to-sequence architectures and compare it against the coverage mechanism proposed in See et al. (2017). On WikiCatSum dataset, we compare a Pointer-Generator (PG) summarization model (Gehrmann et al., 2018) which is based on LSTMs, a Copy Transformer (CTF) model (Gehrmann et al., 2018), and a Convolutional Sequence-to-Sequence (CVS2S) architecture (Gehring et al., 2017). We compare base versions of these models on their own and with the addition of See et al. (2017)'s coverage mechanism (+CovLoss) and the proposed attention mechanism (+DPP). Note that the full coverage mechanism of See et al. (2017) with previous attention weights is only possible in the recurrent PG architecture for which we include an additional variant, namely +CovLoss+CovVec.

On the MultiNews dataset, we compare architectures proposed for it in the literature, namely PG and CTF (Fabbri et al., 2019). We also report results for the Hi-MAP MDS model (Fabbri et al., 2019) which consists of a pointer-generator network (See et al., 2017)

	Pairs	Nb.Words	Nb.Sents	LTR
Film	51,399/2,958/2,861	98.11	4.17	59.46
Company	54,043/2,902/2,981	124.18	5.09	56.29
Animal	47,371/2,705/2,652	92.65	4.71	53.23
MultiNews	44,972/5,622/5,622	263.66	9.97	51.64

Table 1: Number of instances in train/validation/test partitions (Pairs), average summary length (Nb.Words) and number of sentences per summary (Nb.Sents), and average lemma-token ratio (LTR) on clusters' content words.

and an additional attention module based on Maximal Marginal Relevance (MMR, Carbonell & Goldstein, 1998); the latter ranks (input) sentences based on their relevance and novelty with regard to the summary content generated so far. For an upper-bound comparison, we include results achieved by PEGASUS$_{BASE}$ and PEGASUS$_{LARGE}$ pre-trained models (Zhang, Zhao, Saleh, & Liu, 2020a). These are Transformer (Vaswani et al., 2017) based encoder-decoder models with a large number of parameters, pre-trained on large news (HugeNews) and Web (C4) corpora using a self-supervised summarisation objective, i.e. predicting the most salient sentence(s) from the input document, and fine-tuned on MultiNews.[2] Results for PEGASUS$_{LARGE}$ are based on inputs of length 1024 tokens.

Model Selection All CVS2S models use the same encoder and decoder convolutional blocks. The encoder block uses 4 layers, 256 hidden dimensions and stride 3; the decoder uses the same configuration but 3 layers. All embedding sizes are set to 256.[3] PG model variants have a single-layer BiLSTM encoder with 128 word-embeddings, and a single-layer LSTM decoder and 512 hidden sizes. All transformer-based models use encoder and decoder modules with 4 layers and 512 hidden dimensions.[4] CVS2S variants do not use copy from the input in contrast to PG and CTF ones.

All CVS2S and CTF +DPP models (both in for WikiCatSum and MultiNews datasets) compute the aggregate representation of the content attended so far $\mathbf{Z}_t$ based on input context vectors ($\frac{1}{M}\sum_1^{t-1}\mathbf{c}_m$). For the PG +DPP variants, $\mathbf{Z}_t$ is computed on the combination with the decoder output vectors ($\frac{1}{M}\sum_1^{t-1} f(\mathbf{h}_m, \mathbf{c}'_m)$). Decoder states from previous steps in recurrent models provide extra information about previous decoding decisions.

We decode with a beam of size 5. We normalize the log-likelihood of the candidate hypotheses y by their length, $|y|^\alpha$ with $\alpha = 0.9$ (Wu, Schuster, Chen, Le, Norouzi, Macherey, Krikun, Cao, Gao, Macherey, Klingner, Shah, Johnson, Liu, Kaiser, Gouws, Kato, Kudo, Kazawa, & Dean, 2016) for Animal and MultiNews but set $\alpha = 0$ on the Film and Company datasets. All CVS2S models (Fairseq) use no length normalization. We use trigram blocking (Paulus, Xiong, & Socher, 2018) with all models on the WikiCatSum dataset but no coverage penalty (Gehrmann et al., 2018) as experiments indicated it was hurting performance. Trigram blocking is a hard constraint similar to coverage penalty, it aims to reduce redundancy in decoded summary S, by skipping candidate sentence c if there exists

2. PEGASUS$_{BASE}$ (PEGASUS$_{LARGE}$) has hidden size 768 (1024), feed-forward 3072 (4096), number of self-attention heads 12 (16), and number of layers 12 (16).
3. Implementation is based on code from `https://github.com/pytorch/fairseq`
4. We used the OpenNMT base implementations of Copy Transformer and Pointer-Generator, `http://opennmt.net/OpenNMT-py/Summarization.html`.

a trigram overlapping between c and S. Conversely, we use coverage penalty ($\beta = 5$) on MultiNews but no trigram blocking. To make relevance scores sharper, we experimented with temperature values $\tau < 1$ in Equation (7) (Section 3.2) at inference time. In particular for +DPP variants this would further gear the attention towards a more incremental reading of the input content elements. Indeed, within the CTF architecture, sparser relevance brought no improvements on the base and +CovLoss variants but increased the performance of +DPP (with best $\tau = 0.6$).

For the analysis of repetitions (Section 5, Table 3) and attention behaviour (Section 6), we set the beam size to 1 and use none of the decoding control mechanisms mentioned in previous paragraph.

Training Details All convolutional models used dropout (Srivastava, Hinton, Krizhevsky, Sutskever, & Salakhutdinov, 2014) in the encoder and decoder with a rate of 0.2. For the normalization and initialization of the convolutional architectures, we follow Gehring et al. (2017). All CVS2S models were trained with Nesterov's accelerated gradient method, again following (Gehring et al., 2017). CVS2S were trained on a single GPU with batch size 5.

For transformer-based models, we applied dropout with probability of 0.2 and label smoothing (Szegedy, Vanhoucke, Ioffe, Shlens, & Wojna, 2016) with smoothing factor 0.1. The optimizer was Adam (Kingma & Ba, 2015) with learning rate of 2, $\beta_1 = 0.9$, and $\beta_2 = 0.998$; we also applied learning rate warm-up over the first 8,000 steps (6,000 on Animal and Company datasets), and decay as in (Vaswani et al., 2017). CTF variants were trained with 2 GPUs and batch size of 12,288 tokens for WiKiCatSum datasets; and 2 GPUs with batch size of 16,384 tokens for MultiNews.

Pointer-Generator models were trained with the Adagrad optimizer (Duchi, Hazan, & Singer, 2011) and learning rate of 0.15. PG models were trained for 50,000 epochs and best models were selected based on ROUGE scores on the validation set. PG variants were trained with 2 GPUs and batch size of 40 instances for WiKiCatSum datasets; and 4 GPUs with batch size of 40 instances for MultiNews.

5. Results

Automatic Evaluation As standard in abstractive multi-document summarization, we use the automatic ROUGE F_1 metric (Lin, 2004). We measure unigram and bigram overlap (ROUGE-1 and ROUGE-2) as well as the longest common sub-sequence (ROUGE-L) and skip-gram based ROUGE (ROUGE-SU4). In addition to token overlap metrics, we also report BERTScore F_1 (Zhang, Kishore, Wu, Weinberger, & Artzi, 2020b) and Sentence Mover's Similarity (SMS, Clark, Celikyilmaz, & Smith, 2019) to evaluate semantic similarity between the reference and generated summaries.[5] BERTScore F_1 is a token level metric that computes semantic overlap by considering cosine similarity among tokens in the reference and generated summaries. Cosine similarities are computed on BERT (Devlin, Chang, Lee, & Toutanova, 2019) contextualised token embeddings. An advantage of this metric is that it accounts for potential paraphrasing of summary content. SMS also evaluates semantic overlap but at the sentence level (sentence representations are computed as

5. We use the tool released in `https://github.com/eaclark07/sms` with Glove embeddings and bert-score version is roberta-large_L17_no-idf_version=0.3.9(hug_trans=4.5.1).

the average of their word embeddings weighted by the proportion of words they contains). It is suitable for evaluating multi-sentence texts, as is our case with MultiNews and WikiCatSum. Given two multi-sentence texts, it first computes the minimal cost of transforming one into the other (based on the Words Mover Distance (WMD, Kusner, Sun, Kolkin, & Weinberger, 2015) and then transforms this cost into a similarly score (i.e., the smaller the cost, the higher the similarity). Clark et al. (2019) show that this measure correlates well with human judgments and is robust to repetitions (i.e., negatively affected when the text contains repetitions).

We also make use of automatic fact checking systems to assess factual correctness. For the news domain, we employ FactCC (Kryscinski, Keskar, McCann, Xiong, & Socher, 2019), a BERT-based classifier trained on news texts to identify conflicts between a source document and a generated summary. Given a document-sentence pair as input, it assigns a positive label if factual information mentioned in the sentence is consistent with the document, otherwise it assigns a negative label. We approximate factual correctness for multi-document summarisation by classifying all document-sentence pairs formed by pairing all documents in the multi-document input with all sentences in the summary. We then aggregate sentence scores by considering a sentence to be consistent if it is fact checked in at least one of the input documents.[6] We report percentage of positive labels aggregated per summary as a factual correctness score.

For the WikiCatSum dataset, we apply the factual accuracy metric defined in Goodrich, Rao, Liu, and Saleh (2019) that compares *subject-relation-object* triples extracted from the reference (T set) against those extracted from the generated summary (G set). The metric is based on those triples whose *subject-relation* part exists in both T and G and counts the proportion thereof whose *object* part coincides.[7] We use the relation extraction model provided by Sorokin and Gurevych (2017). This model extracts triples and maps relation mentions (i.e., different wordings of the same relation) to knowledge-base relation identifiers. The model was trained on Wikipedia and Wikidata Knowledge-Base (Vrandečić & Krötzsch, 2014).

Table 2 summarizes our results on WikiCatSum; we report average scores across categories (Animal, Film, and Company), under standard decoding with beam search and decoding constraints as described in Section 4. Convolutional architectures are shown in the first block, Pointer Generator models in the second block, and Transformers in the third block. As can be seen, +DPP variants consistently outperform base models (CVS2S, PG, and CTF) and models trained with coverage loss (CVS2S+CovLoss, PG+CovLoss, PG+CovLoss+CovVec, and CTF+CovLoss) on all metrics.

In Table 3, we analyse the degree to which the models generate repetitions under greedy decoding without enforcing any hard constraints such as trigram blocking and coverage or length penalties. We report unigram (rep1) and bigram (rep2) repetitions which we compute as the fraction of next token predictions that have already appeared in the previous prefix (Welleck, Kulikov, Roller, Dinan, Cho, & Weston, 2020). We see that CVS2S+DPP and CTF+DPP reduce repetitions with respect to base and +CovLoss variants. In particular, CTF+DPP produces outputs which are on average ˜30 tokens shorter compared to CTF+CovLoss. On the contrary, all PG variants (middle block) produce broadly similar

6. We use the FactCC model released in `https://github.com/salesforce/factCC`.
7. We relax this constraint and consider sub-strings also as valid.

Model	R1	R2	RL	SU4	BS	SMS	Fact_{acc}
CVS2S	32.66	17.39	28.37	17.78	0.826	73.17	22.9
CVS2S+CovLoss	32.93	17.55	28.57	17.98	0.828	74.41	23.4
CVS2S+DPP	**33.10**	**17.65**	**28.70**	**18.11**	0.827	**74.28**	**23.7**
PG	27.21	15.44	24.30	9.73	0.849	77.73	27.3
PG+CovLoss	26.84	15.37	24.02	9.66	0.845	74.00	26.9
PG+CovLoss+CovVec	26.75	15.20	23.90	9.56	0.848	**79.48**	27.2
PG+DPP	**27.52**	**15.62**	**24.54**	**9.92**	0.849	78.84	**27.4**
CTF	29.87	16.13	25.79	11.58	0.850	81.49	27.4
CTF+CovLoss	30.39	16.43	26.10	**11.92**	0.850	81.28	27.3
CTF+DPP	**30.45**	**16.70**	**26.52**	11.84	0.850	**81.66**	**27.9**

Table 2: ROUGE ROUGE, BERTScore (BS), Sentence Mover's Similarity (SMS), factual accuracy (Fact_{acc}) scores on WikiCatSum test set. We follow (Clark et al., 2019) and report SMS scaled by a factor of 1000.

Model	rep1	rep2	avg.len
CVS2S	.43	.24	63.44
CVS2S+CovLoss	.42	.23	62.05
CVS2S+DPP	**.39**	**.19**	54.95
PG	.34	.13	49.87
PG+CovLoss	.34	.12	48.82
PG+CovLoss+CovVec	.34	.12	47.20
PG+DPP	.34	.13	48.61
CTF	.45	.24	114.15
CTF+CovLoss	.49	.29	138.42
CTF+DPP	**.43**	**.23**	109.38

Table 3: Unigram (rep1) and bigram (rep2) repetition for greedy decoding with no penalties on the WikiCatSum validation set. Average summary length (avg.len) measured by tokens.

output in terms of length and repetition. This suggests that the improved performance for the PG+DPP variant in Table 2 is not simply due to eliminating repetitions. Rather, the model is able to select better content (see Fact_{acc} in Table 2) leading to more precise summaries. Table 4 shows system output generated by base models and their +DPP variants on WikiCatSum. Across architectures, DPP models strike a good balance between coverage, redundancy, and factual errors. More examples are given in Appendix A.

Table 5 shows our results on MultiNews. Recall that we compare the performance of previously used architectures, Pointer-Generator and Copy Transformer. We also include results for the Hierarchical MMR-Attention Pointer-generator (Hi-MAP) model of Fabbri et al. (2019) and PEGASUS (base and large) models (Zhang et al., 2020a). As far as Transformer variants are concerned (third block), we observe that CTF+DPP outperforms the base model, while the model trained with the coverage loss (CTF+CovLoss) does not bring gains in any of the metrics. We suspect the coverage loss with the dot product attention is less effective when generating relatively long summaries.

Gold
Kyun! Ho Gaya Na..., released in English as Look What's Happened Now, is a 2004 Indian Hindi romance film directed by Samir Karnik starring Vivek Oberoi and Aishwarya Rai in lead roles. This was the first film of leading South Indian film actress Kajal Aggarwal, who played a small role as a friend of Aishwarya Rai.
CVS2S
Kyun! Ho Gaya Na..., is a 2011 Indian romantic comedy film directed by Hansal Mehta. The film stars Arjun Rampal, Aishwarya Rai, Vivek Oberoi, Sonu Sood, Sonu Walia, Sonu Sood and ~~Vivek Oberoi~~ in pivotal roles.
CVS2S+DPP
Kyun! Ho Gaya Na..., is a 2016 Indian romantic comedy film written and directed by Karnik UNK. The film stars Arjun Sarja, Aishwarya Rai, Arjun Sarja and Vivek Oberoi in lead roles. The film had musical score by A. T. Ummer.
PG
Kyun! Ho Gaya Na is a 2013 Bollywood romantic comedy film directed by Karnik UNK and produced by UNK UNK under the banner of UNK films. The film features Arjun Khanna and Vivek Rai in the lead roles. Music of the film has been composed by Ashok Bhadra.
PG+DPP
Kyun! Ho Gaya Na... is a 2005 American romantic comedy film directed and written by Karnik Blends. The film stars Aishwarya Rai, Vivek Oberoi, ~~Aishwarya Oberoi, Vivek Rai,~~ Amitabh Bachchan and Vivek Malhotra. The film had its world premiere at the Sundance film festival on january 20, 2016.
CTF
Kyun! Ho Gaya Na. is a 2013 Indian romantic comedy film directed by Karnik. it stars Amitabh Bachchan and Aishwarya Rai in the lead roles, with Vivek Oberoi, ~~Vivek Oberoi and Rai~~ in supporting roles. It was released on 18 january 2013.
CTF+DPP
Kyun! Ho Gaya Na. Na... it is a romantic comedy film starring Arjun Oberoi, Vivek Oberoi ~~and Vivek Oberoi~~ in the lead roles.

Table 4: Gold and model summaries for a WikiCatSum validation example. Text in red highlights inaccurate and hallucinated facts and repetitions are striken out.

Model	R1	R2	RL	SU4	BS	SMS	Fact_{acc}	Length
PEGASUS_{BASE} (C4)	42.24	13.27	21.44	—	—	—	—	—
PEGASUS_{LARGE} (C4)	46.74	17.95	24.26	—	—	—	—	—
PEGASUS_{LARGE} (HugeNews)	47.52	18.72	24.91	—	—	—	—	—
Hi-MAP	43.47	14.89	—	17.41	—	—	—	—
PG	44.89	15.18	20.56	18.48	0.855	134.06	76.9	223
PG+CovLoss	44.92	15.98	21.62	18.63	0.854	136.65	76.5	234
PG+CovLoss+CovVec	45.07	16.03	**21.74**	**18.77**	0.855	136.80	**77.3**	226
PG+DPP	44.80	15.93	21.59	18.54	0.854	135.71	76.7	228
PG+CovLoss+DPP	**45.20**	**16.07**	21.62	**18.79**	0.855	**137.00**	76.6	233
CTF	44.71	15.15	20.80	18.57	0.853	131.16	79.0	236
CTF+CovLoss	44.04	14.87	20.51	18.09	0.853	128.27	79.4	234
CTF+DPP	**45.84**	**15.94**	**21.02**	**19.19**	0.852	**136.28**	**81.9**	235

Table 5: ROUGE, BERTScore (BS), Sentence Mover's Similarity (SMS), factual accuracy (Fact_{acc}) and average summary length in tokens on the MultiNews test set. We follow Clark et al. (2019) and report SMS scaled by a factor of 1000. Scores for Hi-MAP and PEGASUS models are taken from Fabbri et al. (2019) and (Zhang et al., 2020a), respectively.

With regard to recurrent models (second block), we observe that PG+DPP outperforms the base Pointer-Generator, and a variant trained with coverage loss (PG+CovLoss) but not when attention weights from previous steps are taken into account (PG+CovLoss+CovVec). We conjecture that coverage is more effective under recurrent architectures which rely on the

attention mechanism to memorize previous attention vectors at each step. Nevertheless, addition of our DPP mechanism to a model pre-trained with coverage loss (PG+CovLoss+DPP) improves over PG+CovLoss+CovVec across all token overlap metrics, save RL. We speculate that pre-training with a coverage loss yields useful representations which our DPP attention mechanism can take advantage of. On semantics based metrics, although the (+CovLoss+DPP) variant scores lower on Fact_{acc} (w.r.t. source documents) it achieves better SMS and on par BS scores in metrics computed against the reference summaries. Finally, we should point out that all +DPP variants improve over Hi-Map which integrates the MMR mechanism into hierarchical attention. All models perform better than PEGASUS_{BASE} in terms of ROUGE-1 and -2 and are comparable in terms of ROUGE-L. Improvements for this larger pre-trained model are modest, possibly due to the fact that the C4 Web corpus represents a different domain and the input is limited to 512 sub-word units. On the other hand, PEGASUS_{LARGE} achieves larger improvements in terms of ROUGE; this model sees larger chunks of input documents (1,024 tokens) and is pre-trained on news corpora. Example outputs of PG and CTF variants are given in Appendix A.

Both on WikiCatSum and MultiNews, our DPP variants improve the generated summaries in terms of token overlap (ROUGE) and semantic similarity (SMS). BS does not highlight any differences among summaries; we attribute this to the fact that our summaries are long and BS works better as a sentence level metric. Factual accuracy metrics reveal that DPP summaries are more faithful in terms of content. This agrees with the human evaluation results in the next section, judges prefer outputs by DPP variants in terms of factual accuracy.

Human Evaluation In addition to automatic evaluation, we also assessed system output by eliciting human judgments. Human evaluation was carried out on 50 instances which were randomly selected from the MultiNews test set.

The study was conducted on the Amazon Mechanical Turk platform using *Best-Worst Scaling* (BWS, Louviere, Flynn, & Marley, 2015). As the inputs and summaries themselves are rather long, we collected pairwise preference judgments. Participants were presented with a human-written summary and two system summaries and asked to decide which summary was *better* and *worse* in relation to the gold. AMT participants rated summaries along four dimensions: *Informativeness* (Does the summary cover information about the news event present in the human summary?), *Factual Accuracy* (Does the summary avoid producing factual errors?), *Non-Redundancy* (Does the summary avoid repeating information?) and *Grammaticality* (Is the summary fluent and grammatical?). The score of a system was computed as the percentage of times it was chosen as best minus the percentage of times it was selected as worst; scores range from -1 (worst) to 1 (best). We elicited 3 judgments per instance.

Our first evaluation aimed at assessing the impact of DPP attention within each architecture. To this end, we compared the summaries generated by the CTF and CTF+DPP models as well as PG and PG+DPP. The top and middle blocks in Table 6 show our results. For the CTF architecture, DPP attention is perceived better across all four dimensions, and all pairwise differences are statistically significant (using a paired student t-test; $p < 0.05$). For the PG architecture, DPP significantly improves on factual accuracy, informativeness and fluency and performs on par with the base model on redundancy.

	FactAcc	Inform	Gramm	No-Redun
CTF	-0.200	-0.187	-0.173	-0.107
CTF+DPP	**0.200**	**0.187**	**0.173**	**0.107**
PG	-0.173	-0.173	-0.107	-0.053
PG+DPP	**0.173**	**0.173**	**0.107**	0.053
HiMAP	-0.233	-0.327	-0.300	-0.280
CTF+DPP	**0.133**	0.120	0.113	0.073
PG+CovLoss+DPP	0.100	**0.207**	**0.187**	**0.207**

Table 6: System ranking according to human judgments on factual accuracy (FactAcc), informativeness (Inform), grammaticality (Gramm), and non-redundancy (No-Redun).

The second study compared our best DPP variants, PG+CovLoss+DPP and CTF+DPP, against HiMap, the state-of-the-art system of Fabbri et al. (2019). As shown in Table 6 (bottom block), PG+CovLoss+DPP outperforms HiMap across all dimensions and improves informativeness, fluency, and non-redundancy over CTF+DPP. Pairwise differences between HiMap and +DPP models are all statistically significant (using a one-way ANOVA with posthoc Tukey HSD tests $p<0.01$) while differences between DPP models aren't.

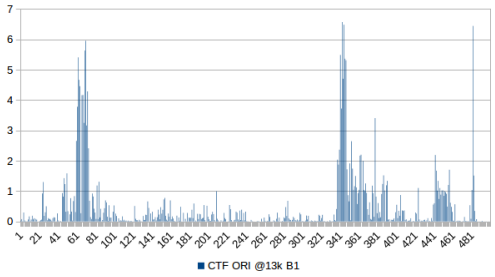
6. Attention Analysis

In this section we inspect the behaviour of attention distributions through generation time steps at inference time. To this end, we link attention weights to input elements, i.e., tokens or token spans, as a way of visualising the workings of different attention modules. This analysis is not intended as an attribution analysis (i.e., finding model features responsible of model predictions) (Jain & Wallace, 2019; Wiegreffe & Pinter, 2019). We use greedy decoding without any hard constraints (e.g., trigram blocking) to illustrate the bare bone behaviour of attention distributions and their relation to the generated summaries.

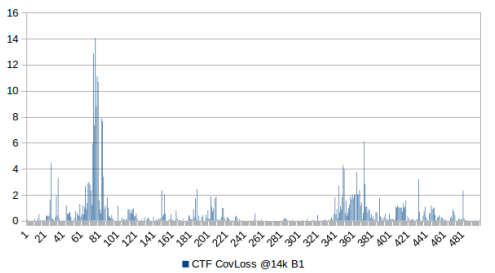
Figure 2 shows accumulated step-wise attention scores on input words on the three Transformer variants CTF, CTF+CovLoss and CTF+DPP. The attention distribution corresponds to the head used for copy. Ideally, we want the attention scores to be placed on relevant and distinct elements from the input which contribute the core content of the summary. As we see in graph (2a), CTF distributes most of its attention among three spans of the source sequence. These are around positions ˜61–81 (*fourth consecutive year*, *sold 80 million copies*), ˜341-361 *sold 81m copies*, and ˜481 (*sell copies*). Accordingly, we see this content repeated in the generated output. So, positions are different but the content is similar. The CTF+CovLoss model in graph (2b) places substantially more attention around positions 73–76 (*The Da Vinci Code*) corresponding with repetitions in the generated text. The CTF+DPP model (2c) keeps the attention scores above 1 at very specific points and places attention at the end of the input sequence (positions ˜484–500) where the last document talks about the note placed asking donors hand in their vinyl. The corresponding summary is richer in content.

We also analyse how attention is placed on the source multi-document sequence across the validation set. Again, we want to asses how much attention is placed on different spans from the input sequence, the proportion of spans which are highly attended to, and

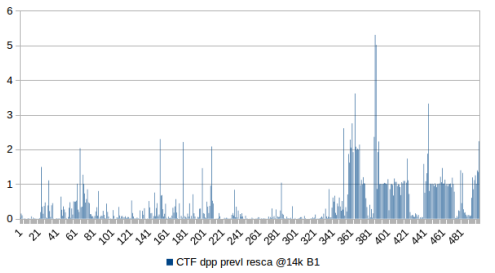
–The Da Vinci Code has sold so many copies –that would be at least 80 million– that it's bound to turn up in book donation piles. But at one charity shop in the UK, it's been donated so heavily that the shop has posted a sign propped up on a tower of Da Vinci Code copies that reads: "you could give us another Da Vinci Code... but we would rather have your vinyl!" the manager of the Oxfam shop in Swansea tells the telegraph that people are laughing and taking pictures of the sizable display: "I would say that we get one copy of the book every day." he says people buy them "occasionally," but with vinyl sales up 25% in the past year, they'd rather take records. Dan Brown's book isn't the only one that shops like Oxfam struggle to re-sell. Last year, Oxfam was hit with a large and steady supply of Fifty Shades of Grey, and it similarly begged donors: "please –no more." but Brown has a particular kind of staying power. The Da Vinci Code was published in 2003, and within six years Brown had booted John Grisham from the no. 1 slot on the list of writers whose books were most often donated to Oxfam's 700 shops, reported The Guardian at the time. The Independent in 2012 reported Brown's best-seller was the most-donated book for the fourth year running. (see why dan brown took heat from the philippines.)



(a) –Dan Brown's Lost Symbol is published next month, and it's not just a relaunch of the Dan Brown novel. The Oxfam shop in Swansea, Canada, has sold more than 80 million copies of The Da Vinci Code, The Guardian reports. Brown, who has sold more than 80 million copies of the book, has sold more than 80 million copies of the book, including his fourth consecutive year. Brown, who has sold more than 80 million copies of the book, has sold more than 80 million copies of the book, including his fourth consecutive year. Brown, who has sold more than 80 million copies of the book, has sold more than 81m copies of the book, including his fourth consecutive year. Oxfam's Oxfam shop, which has sold more than 81m copies of the book, has been receiving an average of one copyof the book, The Guardian reports. Brown has sold more than 81m copies of the book, including his fourth consecutive year, and his fourth consecutive year. Oxfam's Oxfam shop, which has sold more than 81m copies of the book, has been revealed as the most donated author to Oxfam's 700 high street shops.



(b) –Dan Brown has been receiving an average of 80 million copies of The Da Vinci Code worldwide, and he's now sold more than 80 million copies of the book, the guardian reports. The book, the most donated author to Oxfam's 700 high street shops, has been revealed as the most donated author to Oxfam's 700 high street shops, the second-most likely writer to be ditched in a charity shop by readers keen to make some room for books. The book, which had been sold more than 80 million copies of The Da Vinci Code, had all four books to his name –although his fifth consecutive year was published in The Da Vinci Code worldwide, The Guardian reports. The book charts The Da Vinci Code, which had all four books on The Da Vinci Code, and had all four books on The Da Vinci Code, The Guardian reports. The book charts The Da Vinci Code, which had all four books on The Da Vinci Code, and had all four books on The New York Times list. The book charts The Da Vinci Code, which had all four books on The Da Vinci Code, and had all four books on The Da Vinci Code. The book charts The Da Vinci Code, which had all four books on The Da Vinci Code, and had all four books on The Da Vinci Code.[rep.last.sent]



(c) –A charity shop in Swansea, the Lost Symbol, and booksellers say they'll rather donors hand in their vinyl instead of their vinyl instead of The Da Vinci Code. The Oxfam shop in Swansea, the fourth consecutive year, has been receiving an average of one copy of the Dan Brown novel a week for months, leaving them with little room for any other books, reports The Guardian. The book, which is published next month, is the most donated author to Oxfam's 700 high street shops, reports The Guardian. With just four books to his name, Brown did well to see off competition from John Grisham, author of more than 20 and the second-most likely writer to be ditched in a charity shop by readers. The book, which is published next month, is published next month, and Brown has sold more than 80 million copies of the book, reports The Guardian. The book, which is published next month, is published next month, and Brown is reportedly the most donated author to Oxfam's 700 high street shops.

Figure 2: (a) CTF, (b) CTF+CovLoss, and (c) CTF+DPP.

the diversity thereof. The three measures in Table 7 illustrate this. We divide the source sequence of length 500 tokens in spans of length 5 and consider the proportion of spans (out

	CTF	+CovLoss	+DPP
Proportion of spans with attn scores ≥ 0.6	48.51	46.74	44.30
Diversity of spans with attn scores ≥ 0.6	54.97	55.31	56.33
Average step-wise attention entropy	1,76	1,65	1,54

Table 7: Attention behaviour for CTF , CTF+CovLoss and CTF+DPP variants on validation set with beam size 1 and no decoding constraints. Values are averages across instances.

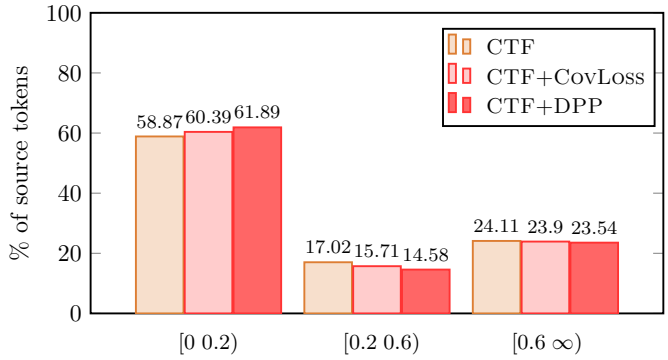


Figure 3: Average percentage of source tokens within different attention ranges.

of 100) that have elements with accumulated attention greater or equal than 0.6. We then measure the diversity of these highly influential spans (second row of the table). We do this by computing the type token ratio $|uniq.unigrams|/|unigrams|$, the higher this value the more diverse the focused spans are. Finally, we look at the average step-wise attention distribution entropy (in nats). The lower this value means at each step the decoder strongly focuses at different source elements rather than evenly attending to them. From Table 7 we can see that the DPP attention focuses on fewer input spans but with higher diversity. This agrees with the fact that the DPP attention will incrementally encourage the decoder to focus on spans uncovered so far. In addition, the lower entropy of the DPP attention highlights that the model makes stronger decisions on which elements to focus, probably reducing attention of those elements already covered.

Figure 3 shows another view of how attention is placed over the input representations. It considers the percentage of tokens from the input sequence that accumulate little attention ($[0, 0.2)$), medium attention ($[0.2, 0.6)$) and high attention ($[0.6, \infty)$) scores across the summary decoding steps. CTF+DPP induces a 3% increase in tokens accumulating close to zero attention scores, i.e., CTF+DPP promotes attention sparsity in an informed way. The smaller percentage of tokens with middle and high attention weights potentially reduces redundant input readings as the diversity analysis shows in Table 7.

Finally, Table 8 shows results for all our evaluation metrics for the three CTF variants with greedy decoding. As we can see, the CFT+DPP outperforms the other variants in all metrics achieving greater similarity with the reference and a large improvement in factual content. DPP attention combined with the copy mechanism potentially regularises output token distributions (Meister, Cotterell, & Vieira, 2020) by decreasing scores of highly probable tokens when they are redundant.

Model	R1	R2	RL	SU4	BS	SMS	Fact_{acc}	Length
CTF	35.29	10.31	19.30	12.88	0.833	98.19	59.0	253
CTF+CovLoss	35.23	10.25	19.10	12.85	0.832	104.77	63.5	253
CTF+DPP	**35.81**	**10.47**	**19.50**	**13.20**	0.833	**109.02**	**65.7**	244

Table 8: ROUGE, BERTScore (BS), Sentence Mover's Similarity (SMS), factual accuracy (Fact_{acc}) and average summary length in tokens on the MultiNews validation set with greedy decoding. We follow Clark et al. (2019) and report SMS scaled by a factor of 1000.

7. Conclusions

We introduced a novel DPP attention mechanism for abstractive MDS where input content is selected both in terms of relevance and diversity. Controlled experiments show that it can be effectively integrated into various sequence-to-sequence architectures achieving improvements in both noisy scenarios (WikiCatSum) and the generation of very long summaries (MultiNews). Human evaluation studies further confirm that the proposed approach improves summary quality in terms of informativeness, factual accuracy, and non-redundancy. It is worth noting that there is nothing inherent in the DPP attention that restricts its application to other language generation tasks.

While in this work we chose a soft representation for the subset of elements selected at a given time step (past memory) as future work it would be interesting to combine the DPP attention mechanism with a *hard* construction of the past memory, i.e., using the past selected tokens (Narayan, Maynez, Adamek, Pighin, Bratanič, & McDonald, 2020).

Acknowledgments

We thank the JAIR reviewers for their constructive feedback. We gratefully acknowledge the financial support of the European Research Council (award number 681760).

Appendix A. Example Output

Examples of system output for the Multi-News dataset in Table 9 and WikiCatSum dataset in Table 10.

Table 9: Example of gold summary and models' outputs on MultiNews validation set.

Gold	– A 71-year-old lawyer is suing United Airlines for more than $1 million after an employee knocked him down during an argument two years ago, the Southeast Texas Record reports. According to the Washington Post, the July 21, 2015, incident started when the TSA at Houston's Bush Intercontinental Airport wouldn't let Ronald Tigner through due to an illegible boarding pass. According to the lawsuit Tigner filed june 7, he returned to the ticketing area. He says two United employees –Alejandro Anastasia and Ianthe Phillips-Allred– refused to help him there, instead laughing and cursing at him. Tigner's attorney, William Hoke, tells KPRC that when Tigner told Anastasia to "wipe that smile off his face," Anastasia threatened violence. Hoke says surveillance video of what happened next is "one of the most inhumane things I've ever seen." video shows Anastasia shove Tigner to the ground, where he lies motionless and splayed out for minutes. Hoke says Tigner hit his head on the ground and was knocked unconscious and that United employees stood around looking at tigner until a passerby finally stopped to help him. "They literally left him there like a piece of garbage," Hoke says. Tigner is suing United, Anastasia, and Phillips-Allred for negligence. The lawsuit claims he was left with "physical disfigurement" and "mental anguish." United says Anastasia's behavior was "unacceptable" and "does not reflect our values." Anastasia was charged with felony injury to an elderly individual and punished with a fine and anger-management classes. He no longer works for United. (United's new flight is a doozy.)
PG	– A 71-year-old man is suing United Airlines for $1 million after he asked for a new boarding pass, the AP reports. According to the lawsuit, the man was left lying there motionless when Dao refused to give up his seat onboard an overbooked flight, the AP reports. "This is just one of the most inhumane things I've ever seen in my life, "says attorney William Hoke." TSA then refused to allow him to pass through the security checkpoint "because of the illegible pass." Tigner was attempting to catch a flight out of Bush Intercontinental Airport, and the video of the Texas pushing a 71-year-old passenger to the floor during an argument over a ticket. The airline is still facing fallout over an incident with David Dao, a passenger who was violently dragged off a flight in Chicago last week, the AP reports. The passenger, Ronald Tigner, is suing United and two of its employees for more than $1 million in damages, the AP reports. The Hman has filed a lawsuit against Houston passenger to ground.
PG+CovLoss+CovVec	– A 71-year-old man is suing United Airlines for $1 million after he refused to give up his seat onboard during an argument over a ticket, the Houston Chronicle reports. Dao and Ianthe Phillips-Allred are also named as defendants, the AP reports. The airline is suing United and two of its employees for more than $1 million in damages, Ronald Tigner filed suit against us has obtained video of another physical confrontation between United Airlines employees and a 71-year-old Houston man. "This is just one of the most inhumane things I've ever seen in my life," attorney William Hoke said in a statement, per the Houston Chronicle. "He allegedly refused to allow him to pass through the security checkpoint "because of the illegible pass." the lawsuit filed last week in Harris county, Tex, where a frequent flyer, Ronald Tigner, a passenger who was violently dragged off a flight in Chicago in april, was released of a Houston-based employee pushing a 71-year-old passenger to the ground by an employee in 2015. The employee was charged with a felony of injuring an elderly individual.

Continued on next page

Table 9 – *Continued from previous page*

PG+CovLoss+DPP	– A 71-year-old man is suing United Airlines for $1 million after he was pushed to the ground by an employee in Chicago, reports the Houston Chronicle. The airline is still facing fallout over an incident with David Dao, a passenger who was violently dragged off a flight in Chicago in April. The video of the video was released of a Houston-based employee pushing a 71-year-old passenger to the floor during an argument over a ticket. The man was left lying there motionless. The confrontation took place about two years ago but was detailed in a lawsuit filed last week in Harris county. Alejandro Anastasia and Ianthe Phillips-Allred are also named as defendants. "This is just one of the most inhumane things I've ever seen in my life, "attorney William Hoke said in the video." He allegedly received a boarding pass that was "illegible." TSA then refused to allow him to pass through the security checkpoint "because of the illegible pass," a frequent flyer, father and grandfather tells the Houston Chronicle. Tigner was attempting to give up his seat onboard an overbooked flight, security officers began dragging him down the aisle and off the plane in front of horrified passengers.
CTF	– Dr. David Dao is suing United Airlines for $1 million in damages after he was pushed to the ground by an employee in Chicago, the Houston Chronicle reports. According to a lawsuit filed last week, Tigner was attempting to catch a flight out of Bush Intercontinental Airport when he was dragged off a United flight in April 2015. He was charged with felony injuring an elderly individual. The lawsuit states United Airlines refused to allow him to pass through the security checkpoint during an argument over a ticket. The video shows the man pushing a 71-year-old passenger to the floor, then physically injured an area attorney. The man, Ronald Tigner, filed a lawsuit against united on June 7 in Harris county, KPRC reports. According to the suit, Tigner was left lying motionless after the incident. The airline is seeking more than $1 million after the video was released of a Houston-based employee in the video. According to KPRC, Tigner's lawyer claims United Airlines employees "laughed at, cursed at and cursed at plaintiff," per The Washington Post. According to court papers, the airline is still facing fallout over an incident with David Dao, who was taken to a hospital in handcuffs after a scuffle between the two men and women in the flight. United Airlines says it will "vigorously defend the man and two of its employees" for decades.
CTF+DPP	– A 71-year-old Houston man is suing United and two of its employees for $1 million after he was pushed to the ground by an employee in an argument over a ticket. Alejandro and Anastasia Ianthe Phillips-Allred, a frequent flyer, were charged with a felony of injuring an elderly individual, the Houston Chronicle reports. The airline says Dao refused to give up his seat onboard pass through the security checkpoint "because of the illegible pass ," the attorney William Hoke says. The lawsuit claims Dao received a boarding pass that was obtained from the Harris county, Tex, who was attempting to catch a flight out of Bush Intercontinental Airport in April 2015, but was detailed in a lawsuit filed against united on june 7, 2015. Ianthe says he was left lying there at plaintiff, cursed and even physically injured an area attorney when he was dragged off a flight in front of horrified passengers. "I've ever seen in my life," Tigner tells KPRC. "This is just one of the most inhumane things I've ever seen." Tigner, who has been named as defendants, is a United Airlines lawyer, among other things, refused to offer assistance to plaintiff in the video, which was posted Tuesday evening by a woman pushing a 71-year-old passenger to the floor during a surveillance camera at the airport. He faces $1 million in damages from Ronald Reagan Airport in Chicago.

	Company
Gold	Thadomal Shahani Engineering College (TSEC) is an engineering and research institute in Mumbai, India. Founded in 1983, the year in which the government of Maharashtra granted permission to start private engineering colleges in the state, it is the first and the oldest private engineering institute affiliated to the University of Mumbai, one of the oldest Universities of Asia. TSEC is recognized by the all India council for technical education with the highest grade A.
PG	The Thadomal Shahani Engineering College (UNK) is the primary engineering college in Mumbai. The college is located in Mumbai, Mumbai, India. It is one of the most important colleges of Mumbai.
PG+CovLoss	Thadomal Shahani Engineering College is a engineering college located in Mumbai, Mumbai. It was founded in 1983 by UNK Shahani, who is the current chairman of the Shahani group.
PG+CovLoss+CovVec	Thadomal Shahani engineering college is a Mumbai of Mumbai. It is a private college in Mumbai. It is one of the most respected engineering colleges in Mumbai.
PG+DPP	The Thadomal Shahani Engineering College (also known as UNK) is a engineering college in Mumbai. It is one of the oldest best engineering colleges in Mumbai.
	Animal
Gold	Agriocnemis Zerafica is a species of damselfly in the family Coenagrionidae. It is native to Africa, where it is widespread across the central and western nations of the continent. It is known by the common name Sahel wisp. This species occurs in swamps and pools in dry regions. There are no major threats but it may be affected by pollution and habitat loss to agriculture and development.
CTF	Agriocnemis Zerafica, the UNK damselfly, is a species of damselfly in the family Platycnemididae. It is endemic to Uganda. Its natural habitat is subtropical or tropical moist lowland forests.
CTF+CovLoss	Agriocnemis Zerafica, the orange skimmer, is a species of damselfly in the family Coenagrionidae. It is found in Angola, Benin, Botswana, Burkina Faso, Burundi, Cameroon, Central African Republic, Chad, Democratic Republic of The Congo, Ivory Coast, Ghana, Liberia, Nigeria, Senegal, Sierra Leone, South Africa, Sudan, Tanzania, Uganda, Zambia, and Zimbabwe.
CTF+DPP	Agriocnemis Zerafica, the orange glider, is a species of dragonfly in the family Coenagrionidae. It is found in Senegal, Guinea-Bissau, Guinea, Sierra Leone, Liberia, Ivory Coast, Ghana, Togo, Benin, Nigeria, Cameroon, Gabon, The Rof the Congo, the Central African Republic, Angola, the Democratic Republic of Congo, Uganda, Kenya, Tanzania, Zambia, and Zimbabwe. Its natural habitats are subtropical or tropical moist montane forests and rivers. It is threatened by habitat loss.
	Film
Gold	Invincible is a 2006 American sports drama film directed by Ericson Core. It is based on the true story of Vince Papale (Mark Wahlberg), who played for the Philadelphia Eagles from 1976 to 1978 with the help of his coach, Dick Vermeil (Greg Kinnear). The film was released in the United States on August 25, 2006.
CVS2S	Invincible is a 2006 American sports drama film written and directed by UNK UNK. It stars Meryl Streep and Anne Hathaway. The film was released in the United States on June 24, 2006.
CVS2S+CovLoss	Invincible is a 2006 American sports drama film directed by Brett Ratner and Starring Meryl Streep and Anne Hathaway. The film was released in the United States on June 24, 2006.
CVS2S+DPP	Invincible is a 2006 American sports drama film written and directed by UNK UNK. It stars Meryl Streep and Anne Hathaway. It is based on the true story of UNK's childhood.

Table 10: Example of gold summary and models' outputs on WikiCatSum validation sets.

References

Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Barzilay, R., McKeown, K. R., & Elhadad, M. (1999). Information fusion in the context of multi-document summarization. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics*, pp. 550–557.

Benmalek, R., Khabsa, M., Desu, S., Cardie, C., & Banko, M. (2019). Keeping notes: Conditional natural language generation with a scratchpad mechanism. *arXiv preprint arXiv:1906.05275*.

Borodin, A. (2009). Determinantal point processes. *arXiv preprint arXiv:0911.1153*.

Carbonell, J. G., & Goldstein, J. (1998). The use of mmr, diversity-based reranking for reordering documents and producing summaries.. In *SIGIR*, Vol. 98, pp. 335–336.

Celikyilmaz, A., Bosselut, A., He, X., & Choi, Y. (2018). Deep communicating agents for abstractive summarization. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 1662–1675, New Orleans, Louisiana. Association for Computational Linguistics.

Cho, S., Lebanoff, L., Foroosh, H., & Liu, F. (2019a). Improving the similarity measure of determinantal point processes for extractive multi-document summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 1027–1038, Florence, Italy. Association for Computational Linguistics.

Cho, S., Li, C., Yu, D., Foroosh, H., & Liu, F. (2019b). Multi-document summarization with determinantal point processes and contextualized representations. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pp. 98–103, Hong Kong, China. Association for Computational Linguistics.

Cho, S., Song, K., Li, C., Yu, D., Foroosh, H., & Liu, F. (2020). Better highlighting: Creating sub-sentence summary highlights. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6282–6300, Online. Association for Computational Linguistics.

Chun, M. M., & Turk-Browne, N. B. (2007). Interactions between attention and memory. *Current opinion in neurobiology*, *17*(2), 177–184.

Clark, E., Celikyilmaz, A., & Smith, N. A. (2019). Sentence mover's similarity: Automatic evaluation for multi-sentence texts. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 2748–2760, Florence, Italy. Association for Computational Linguistics.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (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)*, pp. 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Duchi, J., Hazan, E., & Singer, Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, *12*(Jul), 2121–2159.

Erkan, G., & Radev, D. R. (2004). Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of artificial intelligence research*, *22*, 457–479.

Fabbri, A. R., Li, I., She, T., Li, S., & Radev, D. R. (2019). Multi-news: a large-scale multi-document summarization dataset and abstractive hierarchical model. *arXiv preprint arXiv:1906.01749*.

Gehring, J., Auli, M., Grangier, D., Yarats, D., & Dauphin, Y. N. (2017). Convolutional Sequence to Sequence Learning. In *Proceedings of the 34th International Conference on Machine Learning*, pp. 1243–1252, Sydney, Australia.

Gehrmann, S., Deng, Y., & Rush, A. (2018). Bottom-up abstractive summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 4098–4109. Association for Computational Linguistics.

Goodrich, B., Rao, V., Liu, P. J., & Saleh, M. (2019). Assessing the factual accuracy of generated text. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '19, p. 166–175, New York, NY, USA. Association for Computing Machinery.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*, 1735–80.

Jain, S., & Wallace, B. C. (2019). Attention is not Explanation. 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)*, pp. 3543–3556, Minneapolis, Minnesota. Association for Computational Linguistics.

Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations, ICLR*.

Kryscinski, W., Keskar, N. S., McCann, B., Xiong, C., & Socher, R. (2019). Neural text summarization: A critical evaluation. 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)*, pp. 540–551, Hong Kong, China. Association for Computational Linguistics.

Kulesza, A., & Taskar, B. (2010). Structured determinantal point processes. In *Advances in neural information processing systems*, pp. 1171–1179.

Kulesza, A., & Taskar, B. (2012). Learning determinantal point processes. *arXiv preprint arXiv:1202.3738*.

Kusner, M. J., Sun, Y., Kolkin, N. I., & Weinberger, K. Q. (2015). From word embeddings to document distances. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, p. 957–966. JMLR.org.

Lebanoff, L., Song, K., & Liu, F. (2018). Adapting the neural encoder-decoder framework from single to multi-document summarization. *arXiv preprint arXiv:1808.06218*.

Li, L., Liu, W., Litvak, M., Vanetik, N., & Huang, Z. (2019). In conclusion not repetition: Comprehensive abstractive summarization with diversified attention based on determinantal point processes. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pp. 822–832, Hong Kong, China. Association for Computational Linguistics.

Li, W., Xiao, X., Lyu, Y., & Wang, Y. (2018). Improving neural abstractive document summarization with explicit information selection modeling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 1787–1796, Brussels, Belgium. Association for Computational Linguistics.

Lin, C.-Y. (2004). ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out: Proceedings of the ACL-04 Wo rkshop*, pp. 74–81, Barcelona, Spain.

Liu, P., Saleh, M., Pot, E., Goodrich, B., Sepassi, R., Kaiser, L., & Shazeer, N. (2018). Generating Wikipedia by summarizing long sequences. In *Proceedings of the 6th International Conference on Learning Representations*, Vancouver, Canada.

Liu, Y., & Lapata, M. (2019). Hierarchical transformers for multi-document summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 5070–5081, Florence, Italy. Association for Computational Linguistics.

Louviere, J. J., Flynn, T. N., & Marley, A. A. J. (2015). *Best-Worst Scaling: Theory, Methods and Applications*. Cambridge University Press.

Luong, T., Pham, H., & Manning, C. D. (2015). Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 1412–1421, Lisbon, Portugal.

Macchi, O. (1975). The coincidence approach to stochastic point processes. *Advances in Applied Probability*, *7*(1), 83–122.

Mariet, Z. E., Ovadia, Y., & Snoek, J. (2019). Dppnet: Approximating determinantal point processes with deep networks. In *Advances in Neural Information Processing Systems*, pp. 3218–3229.

Meister, C., Cotterell, R., & Vieira, T. (2020). If beam search is the answer, what was the question?. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 2173–2185, Online. Association for Computational Linguistics.

Nallapati, R., Zhou, B., Gulcehre, C., Xiang, B., et al. (2016). Abstractive text summarization using sequence-to-sequence rnns and beyond. *arXiv preprint arXiv:1602.06023*.

Narayan, S., Cohen, S. B., & Lapata, M. (2018). Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. *arXiv preprint arXiv:1808.08745*.

Narayan, S., Maynez, J., Adamek, J., Pighin, D., Bratanič, B., & McDonald, R. (2020). Stepwise extractive summarization and planning with structured transformers. In *Proceedings of the EMNLP 2020*.

Paulus, R., Xiong, C., & Socher, R. (2017). A deep reinforced model for abstractive summarization. *arXiv preprint arXiv:1705.04304*.

Paulus, R., Xiong, C., & Socher, R. (2018). A deep reinforced model for abstractive summarization. In *Proceedings of the 6th International Conference on Learning Representations*, Vancouver, Canada.

Perez-Beltrachini, L., Liu, Y., & Lapata, M. (2019). Generating summaries with topic templates and structured convolutional decoders. *CoRR*, *abs/1810.09995*.

Radev, D. R., Jing, H., Styś, M., & Tam, D. (2004). Centroid-based summarization of multiple documents. *Information Processing & Management*, *40*(6), 919–938.

See, A., Liu, P. J., & Manning, C. D. (2017). Get to the point: Summarization with pointer-generator networks. *arXiv preprint arXiv:1704.04368*.

Song, K., Zhao, L., & Liu, F. (2018a). Structure-infused copy mechanisms for abstractive summarization. In *Proceedings of the 27th International Conference on Computational Linguistics*, pp. 1717–1729, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Song, Y., Yan, R., Feng, Y., Zhang, Y., Zhao, D., & Zhang, M. (2018b). Towards a neural conversation model with diversity net using determinantal point processes. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Sorokin, D., & Gurevych, I. (2017). Context-Aware Representations for Knowledge Base Relation Extraction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1784–1789. Association for Computational Linguistics.

Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, *15*(1), 1929–1958.

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. D., & Weinberger, K. Q. (Eds.), *Advances in Neural Information Processing Systems 27*, pp. 3104–3112. Curran Associates, Inc.

Suzuki, J., & Nagata, M. (2017). Cutting-off redundant repeating generations for neural abstractive summarization. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pp. 291–297, Valencia, Spain. Association for Computational Linguistics.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pp. 2818–2826.

Tan, J., Wan, X., & Xiao, J. (2017a). Abstractive document summarization with a graph-based attentional neural model. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1171–1181, Vancouver, Canada. Association for Computational Linguistics.

Tan, J., Wan, X., & Xiao, J. (2017b). Abstractive document summarization with a graph-based attentional neural model. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1171–1181. Association for Computational Linguistics.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. In *Advances in Neural Information Processing Systems*, pp. 5998–6008.

Vrandečić, D., & Krötzsch, M. (2014). Wikidata: a free collaborative knowledgebase. *Commun. ACM*, *57*(10), 78–85.

Welleck, S., Kulikov, I., Roller, S., Dinan, E., Cho, K., & Weston, J. (2020). Neural text generation with unlikelihood training. *8th International Conference on Learning Representations (ICLR) 2020*.

Wiegreffe, S., & Pinter, Y. (2019). Attention is not not explanation. 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)*, pp. 11–20, Hong Kong, China. Association for Computational Linguistics.

Wu, Y., Schuster, M., Chen, Z., Le, Q., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., Klingner, J., Shah, A., Johnson, M., Liu, X., Kaiser, u., Gouws, S., Kato, Y., Kudo, T., Kazawa, H., & Dean, J. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation..

Yao, J.-g., Fan, F., Zhao, W. X., Wan, X., Chang, E., & Xiao, J. (2016). Tweet timeline generation with determinantal point processes. In *Thirtieth AAAI Conference on Artificial Intelligence*.

Zhang, J., Tan, J., & Wan, X. (2018). Adapting neural single-document summarization model for abstractive multi-document summarization: A pilot study. In *Proceedings of the 11th International Conference on Natural Language Generation*, pp. 381–390, Tilburg University, The Netherlands. Association for Computational Linguistics.

Zhang, J., Zhao, Y., Saleh, M., & Liu, P. (2020a). PEGASUS: Pre-training with extracted gap-sentences for abstractive summarization. In III, H. D., & Singh, A. (Eds.), *Proceedings of the 37th International Conference on Machine Learning*, Vol. 119 of *Proceedings of Machine Learning Research*, pp. 11328–11339. PMLR.

Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., & Artzi, Y. (2020b). Bertscore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.