

# Generating Summaries with Topic Templates and Structured Convolutional Decoders

Laura Perez-Beltrachini      Yang Liu      Mirella Lapata

Institute for Language, Cognition and Computation

School of Informatics, University of Edinburgh

10 Crichton Street, Edinburgh EH8 9AB

{lperez,mlap}@inf.ed.ac.uk      yang.liu2@ed.ac.uk

## Abstract

Existing neural generation approaches create multi-sentence text as a single sequence. In this paper we propose a structured convolutional decoder that is guided by the content structure of target summaries. We compare our model with existing sequential decoders on three data sets representing different domains. Automatic and human evaluation demonstrate that our summaries have better content coverage.

## 1 Introduction

Abstractive multi-document summarization aims at generating a coherent summary from a cluster of thematically related documents. Recently, Liu et al. (2018) proposed generating the lead section of a Wikipedia article as a variant of multi-document summarization and released WikiSum, a large-scale summarization dataset which enables the training of neural models.

Like most previous work on neural text generation (Gardent et al., 2017; See et al., 2017; Wiseman et al., 2017; Puduppully et al., 2019; Celikyilmaz et al., 2018; Liu et al., 2018; Perez-Beltrachini and Lapata, 2018; Marcheggiani and Perez-Beltrachini, 2018), Liu et al. (2018) represent the target summaries as a single long sequence, despite the fact that documents are organized into topically coherent text segments, exhibiting a specific structure in terms of the content they discuss (Barzilay and Lee, 2004). This is especially the case when generating text within a specific domain where certain topics might be discussed in a specific order (Wray, 2002). For instance, the summary in Table 1 is about a species of damselfly; the second sentence describes the *region* where the species is found and the fourth the *type of habitat* the species lives in. We would expect other Animal Wikipedia summaries to exhibit similar content organization.

In this work we propose a neural model which is guided by the topic structure of target summaries, i.e., the way content is organized into sentences and the type of content these sentences discuss. Our model consists of a structured decoder which is trained to predict a sequence of sentence topics that should be discussed in the summary and to generate sentences based on these. We extend the convolutional decoder of Gehring et al. (2017) so as to be aware of which topics to mention in each sentence as well as their position in the target summary. We argue that a decoder which *explicitly* takes content structure into account could lead to better summaries and alleviate well-known issues with neural generation models being too general, too brief, or simply incorrect.

Although content structure has been largely unexplored within neural text generation, it has been recognized as useful for summarization. Barzilay and Lee (2004) build a model of the content structure of source documents and target summaries and use it to extract salient facts from the source. Sauper and Barzilay (2009) cluster texts by target topic and use a global optimization algorithm to select the best combination of facts from each cluster. Although these models have shown good results in terms of content selection, they cannot generate target summaries. Our model is also related to the hierarchical decoding approaches of Li et al. (2015) and Tan et al. (2017). However, the former approach is auto-encoding the same inputs (our model carries out content selection for the summarization task), while the latter generates independent sentences. They also both rely on recurrent neural models, while we use convolutional neural networks. To our knowledge this is the first hierarchical decoder proposed for a non-recurrent architecture.

To evaluate our model, we introduce WIKICAT-SUM, a dataset<sup>1</sup> derived from Liu et al. (2018)

<sup>1</sup>Our dataset and code are available at <https://>

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.

agriocnemis zerafica **EOT** global distribution: the species is known from north-west uganda and sudan, through niger to mauritania and liberia: a larger sahelian range, i.e., in more arid zone than other african agriocnemis. record from angola unlikely. northeastern africa distribution: the species was listed by tsuda for sudan. [ · · · ]. **EOP** very small, about 20mm. orange tail. advised agriocnemis sp. id by kd dijkstra: [ · · · ] **EOP** same creature as previously posted as unknown, very small, about 20mm, over water, top view. advised probably agriocnemis, "whisp" damselfly. **EOP** [ · · · ] **EOP** justification: this is a widespread species with no known major widespread threats that is unlikely to be declining fast enough to qualify for listing in a threatened category. it is therefore assessed as least concern. **EOP** the species has been recorded from northwest uganda and sudan, through niger to mauritania and [ · · · ] **EOP** the main threats to the species are habitat loss due to agriculture, urban development and drainage, as well as water pollution.

Table 1: Summary (top) and input paragraphs (bottom) from the Animal development dataset (EOP/T is a special token indicating the end of paragraph/title).

which consists of Wikipedia abstracts and source documents and is representative of three domains, namely Companies, Films, and Animals. In addition to differences in vocabulary and range of topics, these domains differ in terms of the linguistic characteristics of the target summaries. We compare single sequence decoders and structured decoders using ROUGE and a suite of new metrics we propose in order to quantify the *content adequacy* of the generated summaries. We also show that structured decoding improves content coverage based on human judgments.

## 2 The Summarization Task

The Wikipedia lead section introduces the entity (e.g., *Country* or *Brazil*) the article is about, highlighting important facts associated with it. Liu et al. (2018) further assume that this lead section is a summary of multiple documents related to the entity. Based on this premise, they propose the multi-document summarization task of generating the lead section from the set of documents cited in Wikipedia articles or returned by Google (using article titles as queries). And create WikiSum, a large-scale multi-document summarization dataset with hundreds of thousands of instances.

Liu et al. (2018) focus on summarization from very long sequences. Their model first selects a subset of salient passages by ranking all paragraphs from the set of input documents (based on their TF-IDF similarity with the title of the article). The  $L$  best ranked paragraphs (up to 7.5k tokens) are concatenated into a flat sequence and a decoder-only architecture (Vaswani et al., 2017) is used to generate the summary.

We explicitly model the topic structure of summaries, under the assumption that documents cover different topics about a given entity, while the summary covers the most salient ones and organizes them into a coherent multi-sentence text. We further assume that different lead summaries are appropriate for different entities (e.g. Animals

vs. Films) and thus concentrate on specific domains. We associate Wikipedia articles with “domains” by querying the DBPedia knowledge-base. A training instance in our setting is a (domain-specific) paragraph cluster (*multi-document* input) and the Wikipedia lead section (target *summary*). We derive sentence topic templates from summaries for Animals, Films, and Companies and exploit these to guide the summariser. However, there is nothing inherent in our model that restricts its application to different domains.

## 3 Generation with Content Guidance

Our model takes as input a set of ranked paragraphs  $\mathcal{P} = \{p_1 \dots p_{|\mathcal{P}|}\}$  which we concatenate to form a flat input sequence  $\mathcal{X} = (x_1 \dots x_{|\mathcal{X}|})$  where  $x_i$  is the  $i$ -th token. The output of the model is a multi-sentence summary  $\mathcal{S} = (s_1, \dots, s_{|\mathcal{S}|})$  where  $s_t$  denotes the  $t$ -th sentence.

We adopt an encoder-decoder architecture which makes use of convolutional neural networks (CNNs; Gehring et al. 2017). CNNs permit parallel training (Gehring et al., 2017) and have shown good performance in abstractive summarization tasks (e.g., Narayan et al. 2018). Figure 1 illustrates the architecture of our model. We use the convolutional encoder of Gehring et al. (2017) to obtain a sequence of states  $(z_1, \dots, z_{|\mathcal{X}|})$  given an input sequence of tokens  $(x_1, \dots, x_{|\mathcal{X}|})$ . A *hierarchical* convolutional decoder generates the target sentences (based on the encoder outputs). Specifically, a *document-level* decoder first generates sentence vectors (LSTM Document Decoder in Figure 1), representing the content specification for each sentence that the model plans to decode. A *sentence-level* decoder (CNN Sentence Decoder in Figure 1) is then applied to generate an actual sentence token-by-token. In the following we describe the two decoders in more detail and how they are combined to generate summaries.

### 3.1 Document-level Decoder

The document-level decoder builds a sequence of sentence representations  $(s_1, \dots, s_{|\mathcal{S}|})$ . For exam-

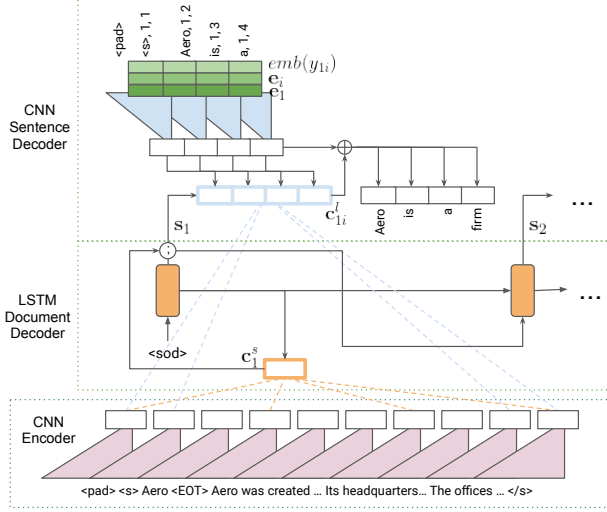


Figure 1: Sequence encoder and structured decoder.

ple,  $s_1$  in Figure 1 is the vector representation for the sentence *Aero is a firm*. This layer uses an LSTM with attention. At each time step  $t$ , the LSTM will construct an output state  $s_t$ , representing the content of the  $t$ -th sentence that the model plans to generate:

$$\mathbf{h}_t = \text{LSTM}(\mathbf{h}_{t-1}, \mathbf{s}_{t-1}) \quad (1)$$

$$\mathbf{s}_t = \tanh(\mathbf{W}_s[\mathbf{h}_t; \mathbf{c}_t^s]) \quad (2)$$

where  $\mathbf{h}_t$  is the LSTM hidden state of step  $t$  and  $\mathbf{c}_t^s$  is the context vector computed by attending to the input. The initial hidden state  $\mathbf{h}_0$  is initialized with the averaged sum of the encoder output states.

We use a soft attention mechanism (Luong et al., 2015) to compute the context vector  $\mathbf{c}_t^s$ :

$$\alpha_{tj}^s = \frac{\exp(\mathbf{h}_t \cdot \mathbf{z}_j)}{\sum_{j'} \exp(\mathbf{h}_t \cdot \mathbf{z}_{j'})} \quad (3)$$

$$\mathbf{c}_t^s = \sum_{j=1}^{|\mathcal{X}|} \alpha_{tj}^s \mathbf{z}_j \quad (4)$$

where  $\alpha_{jt}^s$  is the attention weight for the document-level decoder attending to input token  $x_j$  at time step  $t$ .

### 3.2 Sentence-level Decoder

Each sentence  $s_t = (y_{t1}, \dots, y_{t|s_t|})$  in target summary  $\mathcal{S}$  is generated by a sentence-level decoder. The convolutional architecture proposed in Gehring et al. (2017) combines word embeddings with positional embeddings. That is, the word representation  $\mathbf{w}_{ti}$  of each target word  $y_{ti}$  is combined with vector  $\mathbf{e}_i$  indicating where this word is in the sentence,  $\mathbf{w}_{ti} = \text{emb}(y_{ti}) + \mathbf{e}_i$ . We extend this

representation by adding a *sentence positional embedding*. For each  $s_t$  the decoder incorporates the representation of its position  $t$ . This explicitly informs the decoder which sentence in the target document to decode for. Thus, we redefine word representations as  $\mathbf{w}_{ti} = \text{emb}(y_{ti}) + \mathbf{e}_i + \mathbf{e}_t$ .

### 3.3 Hierarchical Convolutional Decoder

In contrast to recurrent networks where initial conditioning information is used to initialize the hidden state, in the convolutional decoder this information is introduced via an attention mechanism. In this paper we extend the multi-step attention (Gehring et al., 2017) with sentence vectors  $s_t$  generated by the document-level decoder.

The output vectors for each layer  $l$  in the convolutional decoder, when generating tokens for the  $t$ -th sentence are<sup>2</sup>:

$$\{\mathbf{o}_{t1}^l, \dots, \mathbf{o}_{tn}^l\} = \text{conv}(\{\mathbf{o}_{t1}^{l-1}, \dots, \mathbf{o}_{tn}^{l-1}\}) \quad (5)$$

$$\mathbf{o}_{ti}^l = \mathbf{o}_{ti}^l + \mathbf{s}_t + \mathbf{c}_{ti}^l \quad (6)$$

where  $\mathbf{o}_{ti}^l$  is obtained by adding the corresponding sentence state  $\mathbf{s}_t$  produced by the document-level decoder (Equation (2)) and sentence-level context vector  $\mathbf{c}_{ti}^l$ .  $\mathbf{c}_{ti}^l$  is calculated by combining  $\mathbf{o}_{ti}^l$  and  $\mathbf{s}_t$  with the previous target embedding  $\mathbf{g}_{ti}$ :

$$\mathbf{d}_{ti}^l = W_d^l(\mathbf{o}_{ti}^l + \mathbf{s}_t) + \mathbf{g}_{ti} \quad (7)$$

$$a_{tij}^l = \frac{\exp(\mathbf{d}_{ti}^l \cdot \mathbf{z}_j)}{\sum_{j'} \exp(\mathbf{d}_{ti}^l \cdot \mathbf{z}_{j'})} \quad (8)$$

$$\mathbf{c}_{ti}^l = \sum_{j=1}^{|\mathcal{X}|} a_{tij}^l (\mathbf{z}_j + \mathbf{e}_j) \quad (9)$$

The prediction of word  $y_{ti}$  is conditioned on the output vectors of the top convolutional layer, as  $P(y_{ti}|y_{t\{1:i-1\}}) = \text{softmax}(W_y(\mathbf{o}_{ti}^L + \mathbf{c}_{ti}^L))$ . The model is trained to optimize negative log likelihood  $\mathcal{L}_{NLL}$ .

### 3.4 Topic Guidance

To further render the document-level decoder topic-aware, we annotate the sentences of ground-truth summaries with topic templates and force the model to predict these. To discover topic templates from summaries, we train a Latent Dirichlet Allocation model (LDA; Blei et al. (2003)), treating sentences as documents, to obtain sentence-level topic distributions. Since the number of topics discussed in the summary is larger than the

<sup>2</sup>Padding and masking are used to keep the autoregressive property in decoding.

Company
#12: operation, start, begin, facility, company, expand
#29: service, provide, airline, member, operate, flight
#31: product, brand, sell, launch, company, include
#38: base, company, office, locate, development, headquarters
<i>Epos Now's UK headquarters are located in Norwich, England and their US headquarters are in Orlando, Florida.</i> [#38]
Film
#10: base, film, name, novel, story, screenplay
#14: win, film, music, award, nominate, compose
#18: film, receive, review, office, box, critic
#19: star, film, role, play, lead, support
<i>The film is based on the novel Intruder in the dust by William Faulkner.</i> [#10]
Animal
#0: length, cm, reach, grow, centimetre, size, species
#1: forewing, hindwing, spot, line, grey, costa
#17: population, species, threaten, list, number, loss, endanger
#24: forest, habitat, consist, area, lowland, moist, montane
<i>It might be in population decline due to habitat loss.</i> [#17]

Table 2: Topics discovered for different domains and examples of sentence annotations.

Category	InstNb	R1	R2	RL	TopicNb
Company	62,545	.551	.217	.438	40
Film	59,973	.559	.243	.456	20
Animal	60,816	.541	.208	.455	30

Table 3: Number of instances (InstNb), ROUGE 1-2 recall (R1 and R2) of source texts against target summaries and number of topics (TopicNb).

number of topics discussed in a single sentence, we use a symmetric Dirichlet prior (i.e., we have no a-priori knowledge of the topics) with the concentration parameter set to favour sparsity in order to encourage the assignment of few topics to sentences. We use the learnt topic model consisting of  $\mathcal{K} = \{k_1, \dots, k_{|\mathcal{K}|}\}$  topics to annotate summary sentences with a topic vector. For each sentence, we assign a topic label from  $\mathcal{K}$  corresponding to its most likely topic. Table 2 shows topics discovered by LDA and the annotated target sentences for the three domains we consider.

We train the document-level decoder to predict the topic  $k_t$  of sentence  $s_t$  as an auxiliary task,  $P(k_t|s_{1:t-1}) = \text{softmax}(W_k(\mathbf{s}_t))$ , and optimize the summation of the  $\mathcal{L}_{NLL}$  loss and the negative log likelihood of  $P(k_t|s_{1:t-1})$ .

## 4 Experimental setup

**Data** Our WIKICATSUM data set includes the first 800 tokens from the input sequence of paragraphs (Liu et al., 2018) and the Wikipedia lead sections. We included pairs with more than 5 source documents and with more than 23 tokens in the lead section (see Appendix A for details). Each dataset was split into train (90%), validation (5%) and test set (5%). Table 3 shows dataset statistics.

We compute recall ROUGE scores of the input documents against the summaries to assess the amount of overlap and as a reference for the interpretation of the scores achieved by the models. Across domains content overlap (R1) is ~50 points. However, R2 is much lower indicating that there is abstraction, paraphrasing, and content selection in the summaries with respect to the input. We rank input paragraphs with a *weighted* TF-IDF similarity metric which takes paragraph length into account (Singhal et al., 2017).

The column TopicNb in Table 3 shows the number of topics in the topic models selected for each domain and Table 2 shows some of the topics (see Appendix A for training and selection details). The optimal number of topics differs for each domain. In addition to general topics which are discussed across domain instances (e.g., topic #0 in Animal), there are also more specialized ones, e.g., relating to a type of company (see topic #29 in Company) or species (see topic #1 in Animal).

**Model Comparison** We compared against two baselines: the Transformer sequence-to-sequence model (TF-S2S) of Liu et al. (2018) and the Convolutional sequence-to-sequence model (CV-S2S) of Gehring et al. (2017). CV-S2D is our variant with a single sequence encoder and a structured decoder; and +T is the variant with topic label prediction. TF-S2S has 6 layers, the hidden size is set to 256 and the feed-forward hidden size was 1,024 for all layers. All convolutional 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. CV-S2D models are trained by first computing all sentence hidden states  $s_t$  and then decoding all sentences of the summary in parallel. See Appendix A for models training details.

At test time, we use beam size of 5 for all models. The structured decoder explores at each sentence step 5 different hypotheses. Generation stops when the sentence decoder emits the **End-Of-Document** (EOD) token. The model trained to predict topic labels, will predict the **End-Of-Topic** label. This prediction is used as a hard constraint by the document-level decoder, setting the probability of the EOD token to 1. We also use trigram blocking (Paulus et al., 2018) to control for sentence repetition and discard consecutive sentence steps when these overlap on more than 80% of the tokens.

Model	Company			Film			Animal		
	R1	R2	RL	R1	R2	RL	R1	R2	RL
TF-S2S	.260	.095	.204	.365	.188	.310	<b>.440</b>	<b>.288</b>	<b>.400</b>
CV-S2S	.245	.094	.199	.346	.198	.307	.422	.284	.385
CV-S2D	<b>.276</b>	.105	.213	.377	.208	.320	.423	.273	.371
CV-S2D+T	.275	<b>.106</b>	<b>.214</b>	<b>.380</b>	<b>.212</b>	<b>.323</b>	.427	.279	.379
	A		C	A		C	A		C
CV-S2S	.046		.307	.097		.430	<b>.229</b>		<b>.515</b>
CV-S2D	.051		.314	.098		.429	.219		.499
CV-S2D+T	<b>.051</b>		<b>.316</b>	<b>.101</b>		<b>.433</b>	.223		.506

Table 4: ROUGE F-scores (upper part) and additional content metrics (bottom part).

## 5 Results

**Automatic Evaluation** Our first evaluation is based on the standard ROUGE metric (Lin, 2004). We also make use of two additional automatic metrics. They are based on unigram counts of content words and aim at quantifying how much the generated text and the reference overlap with respect to the input (Xu et al., 2016). We expect multi-document summaries to cover details (e.g., names and dates) from the input but also abstract and rephrase its content. *Abstract* (A) computes unigram f-measure between the reference and generated text excluding tokens from the input. Higher values indicate the model’s abstraction capabilities. *Copy* (C) computes unigram f-measure between the reference and generated text only on their intersection with the input. Higher values indicate better coverage of input details.

Table 4 summarizes our results on the test set. In all datasets the structured decoder brings a large improvement in ROUGE-1 (R1), with the variant using topic labels (+T) bringing gains of +2 points on average. With respect to ROUGE-2 and -L (R2 and RL), the CV-S2D+T variant obtains highest scores on Company and Film, while on Animal it is close below to the baselines. Table 4 also presents results with our additional metrics which show that CV-S2D models have a higher overlap with the gold summaries on content words which do not appear in the input (A). All models have similar scores with respect to content words in the input and reference (C).

**Human Evaluation** We complemented the automatic evaluation with two human-based studies carried out on Amazon Mechanical Turk (AMT) over 45 randomly selected examples from the test set (15 from each domain). We compared the TS-S2S, CV-S2S and CV-S2D+T models.

The first study focused on assessing the extent to which generated summaries retain salient information from the input set of paragraphs. We fol-

Model	Company		Film		Animal	
	QA	Rank	QA	Rank	QA	Rank
TF-S2S	5	1.87	6	2.27	9	1.87
CV-S2S	5	2.27	6.67	1.76	8.33	2.04
CV-S2D+T	7	1.87	7	1.98	9.33	2.09

Table 5: QA-based evaluation and system ranking.

lowed a question-answering (QA) scheme as proposed in Clarke and Lapata (2010). Under this scheme, a set of questions are created based on the gold summary; participants are then asked to answer these questions by reading system summaries alone without access to the input. The more questions a system can answer, the better it is at summarizing the input paragraphs as a whole (see Appendix A for example questions). Correct answers are given a score of 1, partially correct answers score 0.5, and zero otherwise. The final score is the average of all question scores. We created between two and four factoid questions for each summary; a total of 40 questions for each domain. We collected 3 judgements per system-question pair. Table 5 shows the QA scores. Summaries by the CV-S2D+T model are able to answer more questions, even for the Animals domain where the TS-S2S model obtained higher ROUGE scores.

The second study assessed the overall content and linguistic quality of the summaries. We asked judges to rank (lower rank is better) system outputs according to *Content* (does the summary appropriately capture the content of the reference?), *Fluency* (is the summary fluent and grammatical?), *Succinctness* (does the summary avoid repetition?). We collected 3 judgments for each of the 45 examples. Participants were presented with the gold summary and the output of the three systems in random order. Over all domains, the ranking of the CV-S2D+T model is better than the two single-sequence models TS-S2S and CONVS2S.

## 6 Conclusions

We introduced a novel structured decoder module for multi-document summarization. Our decoder is aware of which topics to mention in a sentence as well as of its position in the summary. Comparison of our model against competitive single-sequence decoders shows that structured decoding yields summaries with better content coverage.

## Acknowledgments

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

## References

- Regina Barzilay and Lillian Lee. 2004. Catching the drift: Probabilistic content models, with applications to generation and summarization. *arXiv preprint cs/0405039*.
- David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. **Latent dirichlet allocation**. *Journal of Machine Learning Research*, 3:993–1022.
- Asli Celikyilmaz, Antoine Bosselut, Xiaodong He, and Yejin Choi. 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)*, pages 1662–1675. Association for Computational Linguistics.
- James Clarke and Mirella Lapata. 2010. Discourse constraints for document compression. *Computational Linguistics*, 36(3):411–441.
- Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017. The webnlg challenge: Generating text from rdf data. In *Proceedings of the 10th International Conference on Natural Language Generation*, pages 124–133, Santiago de Compostela, Spain. (INLG 2017).
- Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. 2017. Convolutional Sequence to Sequence Learning. In *Proceedings of the 34th International Conference on Machine Learning*, pages 1243–1252, Sydney, Australia.
- Diederik P. Kingma and Jimmy Ba. 2015. **Adam: A method for stochastic optimization**. In *Proceedings of the International Conference on Learning Representations, ICLR*.
- Jiwei Li, Thang Luong, and Dan Jurafsky. 2015. A hierarchical neural autoencoder for paragraphs and documents. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1106–1115. Association for Computational Linguistics.
- Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, pages 74–81, Barcelona, Spain.
- Peter Liu, Mohammad Saleh, Etienne Pot, Ben Goodrich, Ryan Sepassi, Lukasz Kaiser, and Noam Shazeer. 2018. Generating Wikipedia by summarizing long sequences. In *Proceedings of the 6th International Conference on Learning Representations*, Vancouver, Canada.
- Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421, Lisbon, Portugal.
- Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, Baltimore, Maryland.
- Diego Marcheggiani and Laura Perez-Beltrachini. 2018. **Deep Graph Convolutional Encoders for Structured Data to Text Generation**. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 1–9, Tilburg University, The Netherlands. Association for Computational Linguistics.
- Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. **Don’t give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization**. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807, Brussels, Belgium.
- Romain Paulus, Caiming Xiong, and Richard Socher. 2018. A deep reinforced model for abstractive summarization. In *Proceedings of the 6th International Conference on Learning Representations*, Vancouver, Canada.
- Laura Perez-Beltrachini and Mirella Lapata. 2018. **Bootstrapping Generators from Noisy Data**. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics*, pages 1516–1527, New Orleans, Louisiana.
- Ratish Puduppully, Li Dong, and Mirella Lapata. 2019. **Data-to-Text Generation with Content Selection and Planning**. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence*, Honolulu, Hawaii.
- Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta. ELRA. <http://is.muni.cz/publication/884893/en>.
- Michael Röder, Andreas Both, and Alexander Hinneburg. 2015. **Exploring the space of topic coherence measures**. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining, WSDM ’15*, pages 399–408, New York, NY, USA. ACM.
- Christina Sauper and Regina Barzilay. 2009. Automatically generating Wikipedia articles: A structure-aware approach. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the*

4th International Joint Conference on Natural Language Processing of the AFNLP, pages 208–216, Suntec, Singapore.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada.

Amit Singhal, Chris Buckley, and Manclar Mitra. 2017. Pivoted document length normalization. *ACM SIGIR Forum*, 51(2):176–184.

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

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. 2016. Re-thinking 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*, pages 2818–2826.

Jiwei Tan, Xiaojun Wan, and Jianguo Xiao. 2017. 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)*, pages 1171–1181, Vancouver, Canada.

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*, pages 5998–6008.

Sam Wiseman, Stuart Shieber, and Alexander Rush. 2017. Challenges in data-to-document generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2253–2263, Copenhagen, Denmark.

Alison Wray. 2002. *Formulaic Language and the Lexicon*. Cambridge University Press, Cambridge.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google’s neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, abs/1609.08144.

Category	SentNb	SentLen
Company	5.09±3.73	24.40±13.47
Film	4.17±2.71	23.54±11.91
Animal	4.71±3.53	19.68±18.69

Table 6: Average number of sentences in target summaries (SentNb) and sentence length (SentLen) in terms of word counts.

Wei Xu, Courtney Napoles, Ellie Pavlick, Quanze Chen, and Chris Callison-Burch. 2016. Optimizing statistical machine translation for text simplification. *Transactions of the Association for Computational Linguistics*, 4:401–415.

## A Appendix

### A.1 Data

WikiSum consist of Wikipedia articles each of which are associated with a set of reference documents.<sup>3</sup> We associate Wikipedia articles (i.e., entities) with a set of categories by querying the DBpedia knowledge-base.<sup>4</sup> The WikiSum dataset originally provides a set of URLs corresponding to the source reference documents; we crawled online for these references using the tools provided in Liu et al. (2018).<sup>5</sup>

We used the Stanford CoreNLP (Manning et al., 2014) to tokenize the lead section into sentences. We observed that the Animal data set contains overall shorter sentences but also sentences consisting of long enumerations which is reflected in the higher variance in sentence length (see SentLen in Table 6). An example (lead) summary and related paragraphs in shown in Table 7. The upper part shows the target summary and the bottom the input set of paragraphs. **EOP** tokens separate the different paragraphs, **EOT** indicates the title of the Wikipedia article.

To discover sentence topic templates in summaries, we used the Gensim framework (Řehůřek and Sojka, 2010) and learned LDA models on summaries of the train splits. We performed grid search on the number of topics [10, . . . , 90] every ten steps, and used the context-vector-based topic coherence metric (cf. (Röder et al., 2015)) as guidance to manually inspect the output topic sets and

<sup>3</sup>We take the processed Wikipedia articles from [https://github.com/tensorflow/tensor2tensor/tree/master/tensor2tensor/data\\_generators/wikisum](https://github.com/tensorflow/tensor2tensor/tree/master/tensor2tensor/data_generators/wikisum) released on April 25th 2018.

<sup>4</sup>Entities of Wikipedia articles are associated with categories using the latest DBpedia release <http://wiki.dbpedia.org/downloads-2016-10> to obtain the instance types (<http://mappings.dbpedia.org/server/ontology/classes/>).

<sup>5</sup>The crawl took place in July 2018 and was supported by Google Cloud.

select the most appropriate ones. For competing topic sets, we trained the models and selected the topic set which led to higher ROUGE scores on the development set.

We used the following hyperparameters to train topic models with Gensim (Řehůřek and Sojka, 2010). We set the  $\alpha = 0.001$  and  $\eta = \text{'auto'}$ ; and used the following training configuration: `random_state=100, eval_every=5, chunksize=10000, iterations=500, passes=50`. We train on the preprocessed version of the summaries with lemmas of content words (stop words were removed).

## A.2 Model Training Details

In all convolutional models we used dropout (Srivastava et al., 2014) in both encoder and sentence-level decoder with a rate of 0.2. For the normalisation and initialisation of the convolutional architectures, we follow (Gehring et al., 2017). Similarly, to train the convolutional models we follow the optimisation setup in (Gehring et al., 2017).

For the transformer-based baseline we applied dropout (with probability of 0.1) before all linear layers and label smoothing (Szegedy et al., 2016) with smoothing factor 0.1. The optimizer was Adam (Kingma and 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, and decay as in (Vaswani et al., 2017).

We select the best models based on ROUGE scores on the development set.

As for the data, we discarded examples where the lead contained sentences longer than 200 tokens (often been long enumerations of items). For the training of all models we only retained those data examples fitting the maximum target length of the structured decoder, 15 sentences with maximum length of 40 tokens (sentences longer than this where split). We used a source and target vocabulary of 50K words for all datasets.

On decoding we normalise log-likelihood of the candidate hypotheses  $y$  by their length,  $|y|^\alpha$  with  $\alpha = 1$  (Wu et al., 2016), except for the structured decoder on the Animals dataset where we use  $\alpha = 0.9$ . For the transformer model we use  $\alpha = 0.6$ .

## A.3 Evaluation and System Outputs

In the automatic evaluation we used `pyrouge`<sup>6</sup> and ROUGE-1.5.5.pl with stemming (`parameters=-c 95 -r 1000 -n 2 -m`).

<sup>6</sup>[pypi.python.org/pypi/pyrouge](https://pypi.python.org/pypi/pyrouge)

Table 8 shows an example of gold summary and corresponding question set from the question-answering study in Section 5. Table 9 shows examples of system output on the development set. Specifically, we show summaries generated by CONVS2S and CONVS2D+Topic, and also include the reference Gold standard.



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.

agriocnemis zerafica **EOP** specimen count 1 record last modified 21 apr 2016 nmnh -entomology dept. taxonomy animalia arthropoda insecta odonata coenagrionidae collector eldon h. newcomb preparation envelope prep count 1 sex male stage adult see more items in specimen inventory entomology place area 5.12km. ne. dakar, near kamerene; 1:30-4:30 p.m., senegal collection date 21 may 1944 barcode 00342577 usnm number usnment342577 published name agriocnemis zerafica le roi **EOP** global distribution: the species is known from north-west uganda and sudan, through niger to mauritania and liberia: a larger sahelian range, i.e., in more arid zone than other african agriocnemis. record from angola unlikely. northeastern africa distribution: the species was listed by tsuda for sudan. this record needs confirmation. may also occur in kenya as well. **EOP** very small, about 20mm. orange tail. advised agriocnemis sp. id by kd dijkstra: hard to see details, but i believe this is not a. exilis **EOP** same creature as previously posted as unknown, very small, about 20mm, over water, top view. advised probably agriocnemis, "whisp" damselfly. **EOP** thank you for taking the time to provide feedback on the iucn red list of threatened species website, we are grateful for your input. **EOP** justification: this is a widespread species with no known major widespread threats that is unlikely to be declining fast enough to qualify for listing in a threatened category. it is therefore assessed as least concern. **EOP** the species has been recorded from northwest uganda and sudan, through niger to mauritania and liberia: a larger sahelian range, i.e., in more arid zone than other african **EOP** the main threats to the species are habitat loss due to agriculture, urban development and drainage, as well as water pollution. **EOP** no conservation measures known but information on taxonomy, population ecology, habitat status and population trends would be valuable.

Table 7: Summary (top) and input paragraphs (bottom) from the Animal development dataset.

Film	
Gold	Mary Queen of Scots is a 2013 Swiss period drama directed by Thomas Imbach. It is his first film in English and French language starring the bilingual french actress Camille Rutherford. The film portrays the inner life of Mary, the Queen of Scotland. The film is based on austrian novelist Stefan Zweig's 1935 biography, Mary Stuart, a long-term bestseller in Germany and France but out of print in the UK and the us for decades until 2010. The film was first screened at the 2013 International Film Festival Locarno and was later shown at the 2013 Toronto International Film Festival.
QA	What does the film portrays? [the inner life of Mary , the Queen of Scotland] At which festival was the film first screened? [2013 International Film Festival Locarno] Who is the author of the novel the film is based on? [Stefan Zweig]
TF-S2S	Mary Queen of Scots is a 2013 British biographical film based on the life of Mary Queen Mary Queen of Scots. It was directed by Ian Hart and stars Vanessa Redgrave as the title role. It was released in the United Kingdom on 18 april 2013.
CV-S2S	Mary Queen of Scots is a 2013 German drama film directed by Thomas UNK. It was screened in the contemporary world cinema section at the 2013 Toronto International Film Festival.
CV-S2D+T	Mary Queen of Scots ( german : das UNK der UNK ) is a 2013 German drama film directed by Thomas UNK. The film is based on the life of Mary Ellen of Scots. The film was released in the united states on January 17 , 2013.

Table 8: Example of Gold summary, question set and system outputs for the QA evaluation study.

Company	
Gold	Seagull Book, formerly called Seagull Book & Tape, is an American retail chain bookstore focusing on products for members of the Church of Jesus Christ of latter-day Saints (lds church), with over two dozen stores in Utah, Idaho, Arizona, and Nevada. It was the second largest lds bookstore until being acquired in 2006 by market-leader deseret book, and since then Seagull has continued to operate as a discount chain, distinct from deseret book branded retail stores.
CV-S2S	Seagull Book & Tape, Inc. is a book publishing company based in American Fork, Utah, United States. It was founded in 1987 by Jonathan UNK.
CV-S2D+T	Seagull Book & Tape, Inc. is an American book retailer with 26 stores throughout Utah, Idaho and California. The company is based in Boise, Idaho. The company is based in Boise, Idaho, with its sister company Seagull Book & Tape.
Film	
Gold	To Write Love on Her Arms (also known as Day One; formerly Renee) is a 2012 American biographical drama film written and directed by Nathan Frankowski, starring Kat Dennings, Chad Michael Murray, Rupert Friend, Juliana Harkavy, Corbin Bleu and Mark Saul. The film is based on the life of troubled teenager Renee Yohe and the founding of To Write Love on Her Arms by Jamie Tworkowski, after he and others helped Yohe to overcome her challenges enough to be able to enter rehab. The film premiered on March 11, 2012 at the Omaha Film Festival, and was eventually released direct-to-dvd on March 3, 2015.
CV-S2S	To UNK Love on Her Arms is a 2015 American biographical drama film directed by Renee UNK and written by Renee UNK. The film is based on the true story of a girl whose journey is threatened by her arms.
CV-S2D+T	To Write Love on Her Arms is a 2015 American biographical drama film directed by Renee UNK. The film is based on the true story of Renee UNK. The film was released in the United States on March 3, 2015. The film is based on the book of the same name by Renee UNK.
Animal	
Gold	Compacta Capitalis is a moth in the Crambidae family. It was described by Grote in 1881. It is found in North America, where it has been recorded from Maryland to Florida, West to Texas and possibly Colorado, North to Illinois. The wingspan is about 35 mm. The forewings are white with a reddish-brown shading at the base and along the inner margin and two black discal spots, as well as an irregular subterminal line. There is a dark apical blotch on both wings. Adults are on wing from May to August.
CV-S2S	Compacta UNK is a moth in the Crambidae family. It was described by Barnes and McDunnough in 1918. It is found in North America, where it has been recorded from Alabama, Florida, Georgia, Illinois, Indiana, Kentucky, Maine, Maryland, Massachusetts, Minnesota, New Brunswick, New Hampshire, New Jersey, New York, North Carolina, Ohio, Oklahoma, Ontario, Pennsylvania, Quebec, South Carolina, Tennessee, Texas and Virginia.
CV-S2D+T	Compacta UNK is a moth in the Crambidae family. It was described by Grote in 1878. It is found in North America, where it has been recorded from Florida. It is also found in Mexico. The wingspan is about 20 mm. Adults have been recorded on wing from April to September.

Table 9: Examples of system output on the development set.