EMBEDDIA

Cross-Lingual Embeddings for Less-Represented Languages in European News Media

Research and Innovation Action
Call: H2020-ICT-2018-1
Call topic: ICT-29-2018 A multilingual Next generation Internet
Project start: 1 January 2019 Project duration: 36 months

D5.5: Final dynamic news generation technology (T5.2)

Executive summary
This deliverable describes the final parts of work conducted within Task T5.2 of Work Package WP5. In the previous Deliverable D5.3 from the same task we described two approaches for dynamically deciding the order in which information is presented to the reader in automatically generated news texts. In this deliverable, we describe continuation of the work on improving these methods, most notably the neural network based method briefly described in Deliverable D5.3; present a new method for taking explicitly into account the information the reader has obtained from a previously read text; and a word embedding-based method for determining the degree of similarity between two pieces of information.

Partner in charge: UH

Project co-funded by the European Commission within Horizon 2020 Dissemination Level		
PU	Public	PU
PP	Restricted to other programme participants (including the Commission Services)	–
RE	Restricted to a group specified by the Consortium (including the Commission Services)	–
CO	Confidential, only for members of the Consortium (including the Commission Services)	–

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 825153

Deliverable Information

Document administrative information	
Project acronym:	**EMBEDDIA**
Project number:	**825153**
Deliverable number:	**D5.5**
Deliverable full title:	**Final dynamic news generation technology**
Deliverable short title:	**Final dynamic news generation technology**
Document identifier:	**EMBEDDIA-D55-FinalDynamicNewsGenerationTechnology-T52-submitted**
Lead partner short name:	**UH**
Report version:	**submitted**
Report submission date:	**31/10/2021**
Dissemination level:	**PU**
Nature:	**R = Report**
Lead author(s):	**Leo Leppänen (UH)**
Co-author(s):	**Eliel Soisalon-Soininen (UH), Lidia Pivovarova (UH) Hannu Toivonen (UH)**
Status:	**__ draft, __ final, _x_ submitted**

The EMBEDDIA Consortium partner responsible for this deliverable has addressed all comments received. Changes to this document are detailed in the change log table below.

Change log

Date	Version number	Author/Editor	Summary of changes made
02/08/2021	v0.1	L. Leppänen (UH)	Document initialized.
30/09/2021	v0.2	E. Soisalon-Soininen (UH)	New section: learning news structure from corpora.
30/09/2021	v0.3	L. Pivovarova (UH)	New section: Taking background knowledge into account.
13/10/2021	v0.4	L. Leppänen (UH)	New section: Continued development of the heuristic-based method.
14/10/2021	v0.5	L. Leppänen (UH)	New section: Approximating information similarity using word embeddings.
15/10/2021	v0.6	L. Leppänen (UH)	Editing, removal of old text.
15/10/2021	v1.0	L. Leppänen (UH)	Submitted to internal review.
20/10/2021	v1.1	M. Purver (QMUL)	Internal review comments.
20/10/2021	v1.2	S. Pollak (JSI)	Internal review comments.
25/10/2021	v1.3	L. Leppänen (UH)	Addressed internal review comments.
25/10/2021	v1.4	E. Soisalon-Soininen (UH)	Addressed internal review comments.
25/10/2021	v2.0	L. Leppänen (UH)	Submitted for quality management.
25/10/2021	v2.1	Nada Lavrač (JSI)	Quality control.
26/10/2021	v2.2	L. Leppänen (UH)	Final modifications.
28/10/2021	final	L. Leppänen (UH)	Ready for submission.
29/10/2021	submitted	T. Anžič (JSI)	Report submitted.

Table of Contents

List of abbreviations

BERT	Bidirectional encoder representations from transformers
CNN	Convolutional neural network
GPU	Graphics processing unit
LSTM	Long short-term memory
NLG	Natural language generation
PMR	Perfect match ratio
TPU	Tensor processing unit

1 Introduction

The main objective of the EMBEDDIA project is to develop methods and tools for effective exploration, generation and exploitation of online content across languages thereby building the foundations for the multilingual next generation internet, for the benefit of European citizens and industry using less-represented European languages. One facet of this effort is Work Package 5 (WP5), which is concerned with *Natural Language Generation* (NLG). Natural language generation is a "subfield of artificial intelligence and computational linguistics that is concerned with the construction of computer systems that can produce understandable text in English or other human languages from some underlying non-linguistic representation of information" (Reiter & Dale, 1997; Gatt & Krahmer, 2018). More specifically, the focus of WP5 is on *automated journalism*, which concerns the automated generation of *news* texts (Dörr, 2015; Caswell & Dörr, 2018).

In order to support journalists and media companies in efficiently reaching as many demographics as possible, the objective of WP5 is to design and develop news automation systems that take data as input and produce a textual report in response; the methods designed in EMBEDDIA aim to be transferable across languages, transferable across domains, and transparent in their NLG process.

Following the requirements above, WP5 developed a flexible, accurate, and transparent NLG system architecture that can be transferred to new domains and languages with minimal human effort; develop tools for creation of dynamically evolving content, incorporating narrative structure and user knowledge; and develop tools for creation of figurative language and headlines. The work package consists of three tasks (where this deliverable reports on work done in Task T5.2):

- Task T5.1, Multilingual text generation from structured data, adapts NLG technology for the requirements of news generation. The task develops mechanisms for (i) determining what is interesting or important in the given data and deciding what to report, and for (ii) rendering that information in an accurate manner (iii) in multiple languages.
- Task T5.2, Multilingual storytelling and dynamic content generation, develops a novel method for automatically organising news articles based on the domain of the article.
- Task T5.3, Creative language use for multilingual news and headline generation, makes the generated texts more varied and colourful by generating creative expressions, especially in headlines. We find similar terms and metaphors by finding analogous terms in different contexts using context-dependent embeddings. A special focus will be on cross-cultural metaphors.

More specifically, the goal of Task T5.2 is to identify general methods for document *structuring* that can be similarly applied to many domains and situations. This means looking outside of both the 'story level templates' frequently used in industry, which are effectively large hand-coded decision trees for selecting what information to present in the text and in which order; and the massively data-driven approaches frequented in academia, which depend on very large corpora of aligned input data and output text. The first are hardly 'dynamic' and transfer poorly (mostly not at all) between domains, and while the latter are technically transferable, they tend to assume the existance of a large corpora of aligned input data and output text, which limits them from being applied to the myriad of potential news domains that have been traditionally too expensive to cover using human means, essentially creating a type of large-scale bootstrapping problem. For an extended overview of the relevant background to document planning, and the larger context of this work, we direct readers to the previously published deliverable D5.3.

In this deliverable, we report the final developments relating to task T5.2. In the previous Deliverable D5.3 from the same task we described a heuristic-based approach for dynamically deciding the order in which information is presented to the reader in automatically generated news texts, and introduced a work-in-progress approach base on neural processing. In this deliverable, we describe the continuation of our work on improving the first of these methods (Section 2); a word embedding-based variant of that method for determining the degree of similarity between two pieces of information (Section 3); a new method for taking explicitly into account the information the reader has obtained from a previously read text (Section 4); and the completed neural-network based method introduced in Deliverable D5.3

(Section 5). We then present our concluding thoughts on this work, including a brief touch on the forthcoming evaluation of the work (Section 6), and finally list the associated outputs (Section 7).

2 Initial evaluation of the heuristic-based baseline method

In the previous Deliverable D5.3 of this task, we described a heuristic-based method for structuring news content (see Section 4 of Deliverable D5.3). Inspired by an analysis of works on how human-written news texts tend to be structured, the heuristic bases its decisions on what information to include in the text, and in which order the information be presented in, on three factors: an estimate of the newsworthiness of a piece of information, thematic similarity and contextual similarity. Thematic similarity describes how similar the topics of two arbitrary messages (i.e. pieces of information) are. This aspect captures, for example, that – all other things equal – it is intuitively more reasonable to follow a fact about a nation's spending on health care with other messages related to health care spending, than with a message about some unrelated topic such as football. Contextual similarity describes how similar the contexts of two arbitrary facts are. It captures the intuitive notion that it is more reasonable to follow a fact about Finland's spending on preventive medicine in 2020 with another piece of information about Finland in 2020, rather than about Austria in 1990. The goal of combining these three factors is to produce a text that contains as newsworthy messages as possible, while also enforcing a level of coherence into the document.

Following minor modifications to the method by reformulating the thematic similarity measure in terms of un-shared suffix rather than shared prefix which slightly simplified the code in the context of our specific case study, we evaluated the method with domain experts, i.e. journalists. The evaluators were presented with articles generated using both the proposed heuristic-based method, as well as baseline texts created using a system that greedily builds the document plan from maximally newsworthy messages. During the evaluation, the evaluators would read a text and then indicate their agreement with the following statements:

Q1: The text matches the heading

Q2: The text is coherent

Q3: The text lacks some pertinent information

Q4: The text contains unnecessary information

Q5: The text has a suitable length

The results of the evaluation are presented in Table 1. The combined work, i.e. the method described in Deliverable D5.3 and since then slightly modified as described above, and the results obtained from the evaluation, were published by Leppänen and Toivonen (2021) in the Nordic Conference on Computational Linguistics. The published paper is presented as Appendix A of this deliverable.

3 Approximating information similarity using word embeddings

As noted above, a key part of the heuristic-based method described in D5.3 (see Leppänen and Toivonen (2021), provided as Appendix A in this deliverable) is a similarity measure between two data points. In the heuristic-based method, we estimate this similarity using two factors: first, whether the data points share the exact same values for the `location` and `timestamp` fields (i.e. whether they discuss the same location and the same point in time), and more importantly using a hierarchy of labels used to describe the interpretation of the datapoints' values.

Table 1: Results obtained during the evaluation. Parentheses indicate answer ranges and whether the higher (↑), lower (↓) or middle values are to be interpreted as the best. The p_{MWU} column contains the (uncorrected) p-value of a two-sided Mann-Whitney U test. An asterisk indicates the p-value is statistically significant also after applying a Bonferroni correction to account for multiple tests.

	Our method			Baseline			
Statement	Median	Mean	SD.	Median	Mean	SD.	p_{MWU}
Q1 (1–7, ↑)	5	4.40	1.64	2	1.80	0.41	< 0.001*
Q2 (1–7, ↑)	5	4.33	1.76	2	1.60	0.51	< 0.001*
Q3 (1–7, ↓)	4	4.47	1.81	6	5.80	1.42	0.049
Q4 (1–7, ↓)	5	5.13	1.55	6	6.33	0.62	0.024
Q5 (1–5, 3 best)	3	2.93	0.59	4	4.07	0.70	< 0.001*

As an example of the latter type of similarity, we would use the value `health:cost:hc2:mio_eur` to indicate that a number is to be interpreted as the amount of money, measured in millions of euros, spent by some nation on rehabilitative care in some time period. Denoting the previous example as F_1, we can consider two other examples F_2 and F_3, where F_2 is `health:cost:hc2:eur_hab`, the cost of rehabilitative care as euros per inhabitant and F_3 is `health:cost:hc41:mio_eur`, the cost of health care related imaging services in millions of euros. Intuitively, F_1 and F_2 are thematically closer than F_1 and F_3. We'd then approximate this similarity by observing either the amount of shared prefix or unshared suffix between two such hierarchical label.

The first of these factor has a significant downside in that it is completely binary, where it would be preferable to consider, e.g., Finland and Sweden more similar than Finland and Albania. The second factor, in turn, depends on the existence of the hierarchy of labels. While our specific context gives us such a hierarchy effectively for free, this requirement still limits the applicability of our approach to domains where such hierarchies are more difficult to establish.

Motivated by the above observations, we investigated an alternative approach based on word embedding-based text similarity. Namely, our revised approach assigns each message (i.e. an atomic unit of information that can be either included in or excluded from the text, see Deliverable D5.2 for additional details) an embedding in a word-embedding space using the following approach. First, we create from each message a temporary single-node document plan consisting of only that message. Next, we apply the template selection, lexicalization, entity name resolution and morphological realization stages of the NLG pipeline to the temporary document plan, thus obtaining a sentence-length natural language representation of the message. For example, the English language representation of one message is as follows: "*In January 2021, in Austria, the monthly growth rate of the harmonized consumer price index for the category 'food and non-alcoholic beverages' was -3.3 points*".

Having thus obtained natural language representations for all the potentially interesting messages, we then conduct a modified document planning process, where the similarity component of the document planning heuristic is based on the cosine similarity between the messages' representations in word embedding space, rather than the binary equivalence and the hierarchy similarities used in the previously described heuristic-based method. In other words, when comparing the similarity between two messages, we take their natural language expressions and query a language model (in our case, the FinEst BERT model developed within the EMBEDDIA project) for their embeddings. As the natural language expressions consist of multiple BERT tokens, we use the FinEst model's tokenizer to split the sentence into BERT tokens, obtain those tokens' word embeddings, and take their mean. This results in a BERT-embedding sized sentence embedding, which we treat as a *message embedding*. As an alternative to taking the mean of the individual word embeddings, it would be possible to use the `[CLS]` token embedding instead. The similarity between two messages is then calculated using the cosine distance between the messages' embeddings.

Initial experiments with the approach described above revealed that, while the process did seemingly

In January 2021, in Austria, the monthly growth rate of the harmonized consumer price index for the category 'food and non-alcoholic beverages' was -3.3 points. It was 4.3 percentage points less than the EU average. In February 2021, it was 2.9 percentage points more than the EU average. It was 3.2 points. The country had the highest monthly growth rate of the harmonized consumer price index for the category 'food and non-alcoholic beverages' across the observed countries. In January 2021, the country had the 23rd highest value for it across the observed countries.

In February 2021, the monthly growth rate of the harmonized consumer price index for the category 'unprocessed food' was 5.2 percentage points more than the EU average. It was 5.7 points. In January 2021, it was 5.9 percentage points less than the EU average. It was -4.3 points. In February 2021, the country had the highest monthly growth rate of the harmonized consumer price index for the category 'unprocessed food' across the observed countries. In April 2021, the country had the 25th highest value for it across the observed countries.

Figure 1: An excerpt from a text produced using the word embedding-based document planning approach.

improve upon the simple baseline approach that greedily selects the most newsworthy messages to the document plan, it is not clear that it performs better than the heuristic-based approach.

An additional complication of the approach is related to its resource usage. While the individual embeddings are not very costly to compute, and take relatively little memory, the amount of messages that have to be considered in the generation process is very large. Even after limiting the data to very recent data points, we often see messages number close to 200,000. Producing, and most importantly storing in memory, this many embeddings takes non-trivial resources, thus limiting the approach's potential usefulness in a real-time setting especially in contexts where the processing is done without hardware such as GPUs or TPUs that would significantly speed up the neural computations.

To counteract this problem, we incorporated limits to the number of messages that can be considered. At present, the system only generates the top-10,000 most newsworthy messages for countries other than the focus country of the text being generated, and during document planning only conducts the embedding-based similarity calculation for a short-list of the 1,000 most high-scoring candidate messages. An example of a text produced from Eurostat data describing consumer price indices is presented in Figure 1. The example was generated using the Eurostat case study system within which this approach was developed (see deliverable D5.2 for additional details on the system).

4 Taking background knowledge into account

According to the task description, document planning should take into account background information already acquired by the user. In an optimal situation, we would have access to a *global* user history which contained information about all the articles the user had previously read. This, however, would require, maintaining user accounts within the EMBEDDIA platform and providing confidentiality to an extent not feasible within the project. As such, we present a modified document planning process that takes into account a *local* history of the user that is available without setting up complex user accounts or logging of the texts read. Our approach is based on obtaining the messages present in the previously read text, and adjusting newsworthiness weights during the generation of the next text based on the previously used messages.

To obtain the messages included in the previously generated article, the system re-generates the document plan for the previously generated article. While a global history would, as noted above, require

some type of user activity storage, this type of local history is significantly easier to acquire. For example, if the system is embedded in a website that allows the user to request articles using dropdown menus to select that article's target location, it is trivial to include in a request what article the user was reading when they requested the next article.

The module takes as an input nucleus messages, satellite messages and messages, previously reported to a user. All messages are accompanied with newsworthiness scores. Assuming that the user just read the previous messages, the module aims at balancing of two goals: keeping cohesion and avoiding redundancy. To keep cohesion, the module increases newsworthiness for the nucleus messages that belong to the same types with the previous messages. To avoid redundancy, the module decreases newsworthiness of the satellite messages that are identical to the previous messages.

As an example, let us consider a news article about Finland generated based on the Eurostat database. If this is the first news piece queried by the user, a news article will contain the text, shown in Figure 2.

> In March 2020, in Finland, the monthly growth rate of the harmonized consumer price index for the category 'health' was 2.4 points. It was 2.2 percentage points more than in US. In Turkey, it was 2.6 percentage points more than in US. It was 2.8 points. Finland had the 2nd highest monthly growth rate of the harmonized consumer price index for the category 'health' across the observed countries. In Turkey, the yearly growth rate of the harmonized consumer price index for the category 'health' was 10.4 percentage points more than in US.
>
> In Estonia, it was 1.3 percentage points more than in US and January 2020, in Finland, the monthly growth rate of the harmonized consumer price index for the category 'education' was 1 percentage points less than in US. In the country, it was -0.8 points. It was 0.8 percentage points less than the EU average. In Lithuania, it was 1.3 percentage points more than in US and Iceland, it was 0.9 percentage points more than in US.

Figure 2: An example of a text produced when the text is the first generated for the user.

However, if the user has previously read an article about Lithuania, the same query will result in a different output, presented in Figure 3.

Among other differences, it can be seen that the first and the last sentences in the text have changed. The first, unconditioned, article starts with a message about the 'health' category in March 2020, while the article conditioned by the previous message reports 'education' in January 2020—this is to increase cohesion, since 'education' in January was reported in the news article about Lithuania. At the same time, the last sentence disappears because we assume that the user already knows this information.

The cohesion-preserving adjustment is presented in Algorithm 1. For each message in the nucleus, the algorithm tries to find a similar previous message, i.e. a message that has the same *value_type* and *timestamp*. Then the newsworthiness of the new message is adjusted proportionally to the newsworthiness of the previous message, with adjustable coefficient α.

The redundancy-avoiding adjustment is presented in Algorithm 2. If the message has been previously reported, its newsworthiness decreases with exponential scaling factor β. The idea is that if a message is highly relevant for the current query it could be in principle repeated but this might happen only with very few most worthy messages.

The approach described above has been developed in the context of the Eurostat case study system (see Deliverable D5.2), and will be evaluated in that larger context within the scope of Task T5.4.

In January 2020, in Finland, the monthly growth rate of the harmonized consumer price index for the category 'education' was 1 percentage points less than in US. It was -0.8 points and 0.8 percentage points less than the EU average. In February 2020, it was 0.2 percentage points less than in US. The country had the 8th highest monthly growth rate of the harmonized consumer price index for the category 'education' across the observed countries. The monthly growth rate of the harmonized consumer price index for the category 'education' was 0 points.

In March 2020, the monthly growth rate of the harmonized consumer price index for the category 'health' was 2.4 points. It was 2.2 percentage points more than in US. The country had the 2nd highest monthly growth rate of the harmonized consumer price index for the category 'health' across the observed countries. In February 2020, the country had the 12th highest value for it across the observed countries. The monthly growth rate of the harmonized consumer price index for the category 'health' was -0.3 points. It was 0.5 percentage points less than in US.

Figure 3: An example of a text produced when the text is generated after the user has previously read an article about Lithuania.

Algorithm 1 Pseudocode describing how newsworthiness is adjusted for Nucleus

function INCREASECOHESION(*Nucleus*, *PreviousMessages*)
  for all $m \in Nucleus$ **do**
    $prev\_worthiness$ = FINDPREVIOUSWORTHINESS(m, *PreviousMessages*)
    $m.newsworthiness \leftarrow m.newsworthiness \times (\frac{prev\_worthiness}{\alpha} + 1)$
  end for
end function

function FINDPREVIOUSWORTHINESS(m, *PreviousMessages*)
  for all $p \in PreviousMessages$ **do**
    if $m.value\_type = p.values\_type$ & $m.timestamp = p.timestamp$ **then**
      return $p.newsworthiness$
    end if
  end for
  return 0
end function

Algorithm 2 Pseudocode describing how newsworthiness is adjusted for Satellites

function AVOIDREDUNDANCY(*Satellites*, *PreviousMessages*)
  $biggest\_prev\_worthiness \leftarrow \max(p.newsworthinessforp \in PreviousMessages)$
  for all $m \in Satellites$ **do**
    if $m \in PreviousMessages$ **then**
      $m.newsworthiness \leftarrow m.newsworthiness \times \frac{\beta^{m.worthiness}}{\beta^{biggest\_prev\_worthiness}}$
    end if
  end for
end function

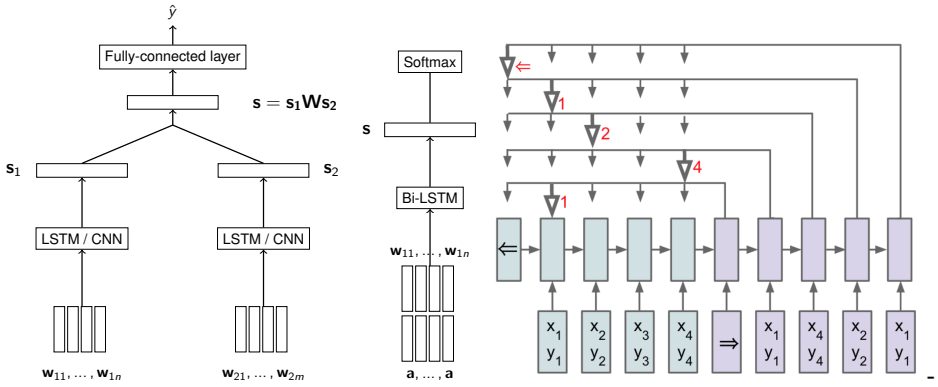

Figure 4: General architectures of the pairwise (left) and position (middle) classifiers, as well as the pointer network (right) as in Vinyals et al. (2015). The vectors **w** refer to token embeddings, and **a** to a paragraph embedding.

5 Learning news structure from corpora

A significant obstacle to the use of end-to-end neural approaches to document planning is the lack of suitable training data where professionally written news reports are aligned with the data records that led to the reports, i.e. the outputs and inputs of the journalistic process. The availability of such datasets is in general very limited, except for certain domains such as sports. For example, Puduppully, Dong, and Lapata (2019) use a dataset of human-written summaries of basketball games aligned with the corresponding score tables, and train an end-to-end neural model for document planning. However, since their approach is suitable to only a limited set of domains, we are highly interested in machine learning based methods that would not require such aligned datasets. As newsrooms have extensive archives of reports written by journalists, i.e. the process outputs, we want to investigate methods for document structuring that are trained on textual corpora alone.

We present three kinds of neural approaches for document structuring in such a data-driven manner, using a corpus of human-written news articles but without the underlying data records. Common for these three approaches is that they are all trained to predict the 'correct' position of an input sentence, i.e. the sentence's most appropriate position within a news article given its content. Since the pieces of information in a news article tend to be arranged according to their relative newsworthiness (Thomson, White, & Kitley, 2008; White, 1997), we make the assumption that, when training these models, a sentence s_i observed before another sentence s_j in a news article is also more newsworthy than s_j. Although we admit that newsworthiness certainly does not always decrease linearly with each sentence, we postulate that such neural models can learn at least some useful patterns between a sentence's newsworthiness and its content. A data-driven neural model might also learn such patterns that could be difficult for the human to notice or write into rules. General architectures of these three models are depicted in Figure 4.

Our first approach is a pairwise neural network, which takes a sentence pair (s_i, s_j) as input, and outputs a value $y \in [0, 1]$, its prediction of whether s_i should precede s_i in a paragraph. The sentences are fed into the network as sequences of token embeddings which reflect the context-dependent meaning of the tokens. This token embedding sequence is then transformed into one representation of the whole sentence, which we accomplish using either a convolution operation or a recurrent neural network. When the pair of sentences (s_i, s_j) has been transformed into a pair of sentence representation vectors $(\mathbf{s}_i, \mathbf{s}_j)$, this vector pair is then further merged into one vector $\mathbf{s}$ through a bilinear transformation such that $\mathbf{s} = \mathbf{s}_i \mathbf{W} \mathbf{s}_j$, where $\mathbf{W} \in \mathbb{R}^{n \times n}$ is a weight matrix and $\mathbf{s}_i, \mathbf{s}_j \in \mathbb{R}^n$. Finally, this merged vector $\mathbf{s}$ is fed to a fully-connected layer, whereafter the final output value y is computed using a sigmoid activation function. This pairwise approach is similar to Chen, Qiu, and Huang (2016) and Agrawal, Chandrasekaran, Batra, Parikh, and Bansal (2016).

The second approach is a neural network that works as a sentence position classifier, following the architecture of Bohn, Hu, Zhang, and Ling (2019). Instead of taking sentence pairs (s_i, s_j) as its input, it takes only one sentence at a time, and outputs a value $y \in \{1, 2, \dots, Q\}$, where Q is the number of *quantiles* into which a paragraph of news text is divided. That is, this approach does not predict the exact position of a sentence within a paragraph, but the most appropriate quantile into which it should belong. The number of quantiles has to be pre-determined, and is fixed for all paragraphs in both training and testing phases regardless of the actual number of sentences in paragraphs. In order to assign each sentence s_i a quantile q_i, we use the formula

$$q_i = \lfloor \frac{i-1}{\frac{n}{Q}} \rfloor \forall i = 1, \dots, Q, \tag{1}$$

where n is the number of sentences in a paragraph. That is, in case there are fewer sentences than quantiles, some quantiles will not be assigned to any sentence, while in the opposite case, several sentences might be assigned to the same quantile.

In this approach, the input sentence s_i is fed into the network as a sequence of token embeddings, as in the first approach, but this time each token embedding is concatenated with an embedding representing the whole article. In this way, the network can relate the given input sentence to its context. In the first approach, this is done by comparing one sentence to another, their relative order given as the training label. Subsequently, this sequence of embeddings is fed to a recurrent layer that outputs a single vector representing the whole sentence. As in the first approach, this vector is then further passed on to a fully-connected layer, and finally an output layer of dimension Q, indicating the predicted quantile of the sentence using the softmax activation function.

Our third approach is a sequence-to-sequence recurrent neural network called a *pointer network* (Vinyals, Fortunato, & Jaitly, 2015). The purpose of this model is to predict the correct sentence sequence (paragraph) $O = (o_1, \dots, o_n)$ given an unordered set of n sentences $S = \{s_1, \dots, s_n\}$, where o_i is the index of some sentence such that the correct i-th sentence is s_{o_i}. The model consists of an *encoder* and a *decoder*, which are both recurrent LSTM networks.

The encoder is fed the sentences S as sentence embeddings $\mathbf{S} = \{\mathbf{s}_1, \dots, \mathbf{s}_n\}$ and it produces a sequence of vector representations $\mathbf{E} = [\mathbf{e}_1, \dots, \mathbf{e}_n]$, i.e. hidden states, where the last vector $\mathbf{e}_n$ is a representation of the whole sequence. Given the encoder hidden states $\mathbf{E}$, the decoder outputs in n steps its prediction $\hat{O} = (\hat{o}_1, \dots, \hat{o}_n)$ for the correct sequence. At each step $i = 1 \dots n$, the decoder's input consists of the encoder hidden states $\mathbf{E} = [\mathbf{e}_1, \dots, \mathbf{e}_n]$, the decoder's hidden state $\mathbf{d}_{i-1}$ from the previous step, and the decoder's previous output (embedding of predicted sentence) $\mathbf{s}_{\hat{o}_{i-1}}$. At the first step $i = 1$, the encoder's last hidden state is used as the decoder's first hidden state ($\mathbf{d}_0 = \mathbf{e}_n$), and $\mathbf{s}_0$ is a random vector.

At each step, the decoder uses an attention layer to compute a vector $\mathbf{u}_i$, attention vector $\mathbf{a}_i$, and a new hidden state vector $\mathbf{d}_i$ such that

$$\mathbf{u}_i = \mathbf{v}^T \tanh(\mathbf{W}_1\mathbf{E} + \mathbf{W}_2\mathbf{d}_{i-1})$$

$$\mathbf{a}_i = \text{softmax}(\mathbf{u}_i)$$

$$\mathbf{d}_i = \sum_{j=1}^{n} a_{ij}\mathbf{e}_j \oplus \mathbf{s}_{i-1}$$

where $\mathbf{v}, \mathbf{W}_1, \mathbf{W}_2$ are the model's learnable parameters and $\oplus$ is a concatenation operation. The attention vector $\mathbf{a}_i$ can be regarded as a probability distribution over the input sentences, and the index of its maximum element gives the next sentence the network 'points' to. Duplicate predictions are avoided by setting the probabilities of previously predicted indices to zero. The new hidden state $\mathbf{d}_i$ is a concatenation of the attention-weigted sum vector of the encoder's hidden states and the decoder input $\mathbf{s}_{i-1}$,

In our experiments, we train these neural models with a news corpus of statistical reports from Statistics Finland.[1] This dataset consists of statistical news articles related to Finland, written in Finnish, Swedish,

[1] https://www.stat.fi/

and English. So far we have focused our experiments on the English subcorpus, which contains 4,383 news articles, of which we set aside 876 (20%) for testing. The only pre-processing we do is filtering out sentences with fewer than three words, as well as paragraphs with fewer than two sentences. This results in a total of 21,854 paragraphs (77,054 sentences), of which 17,287 paragraphs (61,026 sentences) are used for training. We focus our experiments on learning structure and sentence ordering within paragraphs instead of whole articles, since this is more appropriate for our task.

From this corpus, we sample one training dataset (inputs aligned with desired outputs) for each model. For the pairwise classifier, we split paragraphs in half, and form pairs of sentences (s_i, s_j) and label them with $y = 1$ if s_i is in the first half, otherwise with $y = 0$. We also experiment with sampling such sentence pairs from the first and last paragraphs of articles, as well as by taking all possible pairs from a paragraph, but splitting paragprahs into halves seems to result in better performance. For the sentence position classifier, we label each sentence of a paragraph with a quantile determined using equation (1). For the pointer network, we shuffle the sentences of each paragraph and assign the shuffled sequence with correct sentence indices.

In our implementation, we encode each sentence from string format into either one sentence embedding or a sequence of token embeddings. To do this, we opt to use the Sentence-BERT model (Reimers & Gurevych, 2019), a modified version of the BERT language model (Devlin, Chang, Lee, & Toutanova, 2019), specifically designed for obtaining both sentence and context-dependent token embeddings efficiently. We fine-tune this model on our Statistics Finland corpus.

We experiment with two variants of the pairwise classifier with different mechanisms of computing sentence representations: one is a bi-directional LSTM recurrent network, while the other is a convolutional network. In both cases we use one recurrent and one convolutional layer respectively. The hidden state vector of the LSTM has dimension 80, and the convolutional network has 10 kernels, kernel size 1024 $\times$ 6, pool size 1 $\times$ 2, and the length of the input sentence is limited to 30 tokens. The number of hidden units in the fully-connected layer is 64 and the dropout parameter is 0.5. The position classifier also uses a bi-directional LSTM to compute a sentence representation, has a hidden state vector of dimension 100 and the number of quantiles, i.e. output vector dimension is $Q = 10$. In the pointer network, number of attention units is 100, hidden vector size is 100 in the encoder and 100 + 1024 in the decoder, after concatenation with the input embedding. The dimension of the input embeddings is 1024 in all cases, which in the case of the position classifier is concatenated with the embedding of the whole paragraph, and the dimension doubled.

Training the models, we use binary cross-entropy loss function for the pairwise classifiers, and general cross entropy for both the position classifier and the pointer network. For model validation, we use cross validation by splitting the training set into five folds. In order to determine an appropriate number of training epochs, we use early stopping should the validation loss increase for two consecutive epochs. We train the final models using 10 epochs (20 for the pointer network) and batch size 32, Adadelta optimiser with learning rate $\alpha = 1.0$ and parameters $\rho = 0.9$ and $\epsilon = 1e - 6$.

We take a two-fold approach to evaluating the models' capability of learning news structure. First, we compare their performance using quantitative metrics on the task of ordering whole paragraphs. Then, having determined the model that performs best according to these quantitative metrics, we apply the chosen model to news generation and use human evaluation to assess this model's performance.

We evaluate the models on the task of ordering a shuffled paragraph. However, only the pointer network outputs directly an ordering. In order to predict an ordering with a pairwise model, we use beam search so that candidate next sentences are scored by comparing them with all other sentences, and taking the sum of the log-probabilities of all comparisons, similar to Chen et al. (2016). With the position classifier, we score each sentence by computing the weighted average of the predicted quantile, as in Bohn et al. (2019). The evaluation metrics for the ordering task are Kendall's τ (Lapata, 2006), perfect match ratio (PMR), and positional accuracy (PAcc), which are computed as follows:

$$\tau = \frac{1}{|D|} \sum_{p \in D} \frac{2S(O_p, \hat{O}_p)}{\frac{|p|(|p|-1)}{2}} \in [-1, 1]$$

$$PMR = \frac{1}{|D|} \sum_{p \in D} \mathbb{I}(O_p = \hat{O}_p)$$

$$PAcc = \frac{1}{|D|} \sum_{p \in D} \sum_{i \in |p|} \mathbb{I}(o_{s_i} = \hat{o}_{s_i}),$$

where D is the dataset of paragraphs, p is a paragraph i.e. a set of sentences, and $S(O_p, \hat{O}_p)$ is the minimum number of adjacent transpositions required to change the predicted ordering $\hat{O}_p$ to the correct ordering O_p.

The results of our quantitative evaluation are shown in Table 2. We see that the pointer network outperforms the other models with the pairwise CNN right below it, while the other two models have quite clearly lower performance, though still clearly better than random ordering. Thus, we opt to experiment with both the pointer network and pairwise CNN in news generation.

Table 2: Results on the sentence ordering task.

Model	**Kendall's** τ	**PMR** (%)	**PAcc** (%)
Random	0.006	23.4	30.45
Pair-CNN	0.414	42.0	44.6
Pair-BiLSTM	0.189	32.4	36.7
Position-BiLSTM	0.128	28.9	34.6
Pointer network	**0.438**	**44.0**	**45.9**

In order to assess whether these models learn useful patterns of news structure, we apply them to the document planning part of our news generation system. More specifically, we use them for the scoring of candidate messages to be included in a generated news article. To do this, we use the models to give a score to each message, and update its original newsworthiness score by multiplying it with a coefficient. We noticed that the pairwise classifier's architecture is more suitable for this scoring task than the pointer network, and thus we opted to use the pairwise CNN despite its slightly lower performance at the ordering task.

First, we change the messages temporarily into a string format that is close to the final realisation, in order for the models to be able to process them. At this point, the messages have already been assigned a newsworthiness score earlier in the pipeline, by which they have been sorted. With the pairwise CNN, we compute another score for each message by comparing it with k adjacent messages, and computing the averege log-probability of the predictions for all comparisons. Since the model was trained to classify whether one sentence s_i should precede another sentence s_j, a high score for a message would mean that according to the model, that message should be placed before the other k messages. We then change the average log probability back to a probability value v, and compute a coefficient c such that $c = v * (b - a) + a$, where $[a, b]$ is a pre-determined range for the coefficient. The larger this range, the more weight is placed on the neural model's scoring. We found a range of [0.02, 50] to be one that weights the neural score quite significantly, resulting in outcomes noticeably different from the baseline planner (see Figure 5).

At the time of writing, the human evaluation questionnaire is waiting responses. However, Figure 5 above gives an impression of the difference between a simple baseline document planner, our document planner (see Deliverable D5.3, and Appendix A of this deliverable), and a simple planner using the pairwise CNN for updating message newsworthiness scores. The simple planner restricts each paragraph to one category (e.g., 'health') but otherwise the sentences are selected according to the un-updated newsworthiness scores. For example, we observe in the example output that the neural output focuses only in the country given in the headline, while the simplest baseline mentions many different countries, and our document planner also mentions Turkey, despite the headline being about Finland.

Consumer prices in Finland

In March 2020, in Finland, the monthly growth rate of the harmonized consumer price index for the category 'health' was 2.4 points. In Turkey, the harmonized consumer price index for the category 'health' was 70.26 points more than in US. It was 181.7 points. In February 2020, it was 65.53 points more than in US. In March 2020, the monthly growth rate of the harmonized consumer price index for the category 'health' was 2.8 points. In February 2020, the harmonized consumer price index for the category 'health' was 176.79 points.

In January 2020, in Finland, the monthly growth rate of the harmonized consumer price index for the category 'education' was 1 percentage points less than in US. In February 2020, in Turkey, it was 0.9 points. It was 0.7 percentage points more than in US. In Sweden, it was 0.8 points. It was 0.6 percentage points more than in US. In January 2020, in Estonia, it was 1.3 percentage points more than in US.

In March 2020, in Finland, the monthly growth rate of the harmonized consumer price index for the category 'health' was 2.4 points. It was 2.2 percentage points more than in US. In Turkey, the harmonized consumer price index for the category 'health' was 70.26 points more than in US. It was 181.7 points. The monthly growth rate of the harmonized consumer price index for the category 'health' was 2.8 points. Finland had the 2nd highest monthly growth rate of the harmonized consumer price index for the category 'health' across the observed countries.

In January 2020, the monthly growth rate of the harmonized consumer price index for the category 'education' was 1 percentage points less than in US. It was -0.8 points. It was 0.8 percentage points less than the EU average. In Estonia, it was 1.3 percentage points more than in US. It was 1.5 points. It was 1.5 percentage points more than the EU average.

In March 2020, in Finland, the monthly growth rate of the harmonized consumer price index for the category 'health' was 2.4 points. It was 2.2 percentage points more than in US. The country had the 2nd highest monthly growth rate of the harmonized consumer price index for the category 'health' across the observed countries. In February 2020, the country had the 12th highest monthly growth rate of the harmonized consumer price index for the category 'health' across the observed countries. It was -0.3 points. It was 0.5 percentage points less than in US.

In January 2020, the monthly growth rate of the harmonized consumer price index for the category 'education' was 1 percentage points less than in US. It was -0.8 points. It was 0.8 percentage points less than the EU average. In February 2020, it was 0 points. It was 0.2 percentage points less than in US. The country had the 8th highest monthly growth rate of the harmonized consumer price index for the category 'education' across the observed countries.

Figure 5: Example outputs: simple baseline (left), our heuristic-based document planner as described in Appendix A of this deliverable (middle) and the simple baseline extended with neural scoring (right).

6 Conclusions and further work

Task T5.2 has continued the development and evaluation of the two document planning methods first described in Deliverable D5.3 of this same task. An expert evaluation of the first (see Section 2), a method based on an ensemble of heuristics, showed that it statistically significantly outperformed a simpler baseline method. The second, neural network -based approach (see Section 5), was finalized from the early developments described in Deliverable D5.3. Initial evaluations of this approach indicate it is promising, but a proper human evaluation by domain experts is needed to establish how its performance relates to the first method. This will be done within the scope of Task T5.4.

In addition, we described two augmentations to the baseline method described in Deliverable D5.3 and in Appendix A of this deliverable. The first of these is a method for using word embeddings to determine concept similarities, as described in Section 3. The second is a method for accounting for repetition, i.e. the information a user would have obtained from a text they previously read, is described in Section 4.

A more extensive evaluation of the above-described work, conducted as part of a larger evaluation of the Eurostat system to which both T5.1 and T5.2 contribute, will be conducted within the scope of Task 5.4.

7 Associated outputs

Parts of this work are described in detail in the following publication:

Citation	Status	Appendix
Leppänen, L., & Toivonen, H. (2021) A Baseline Document Planning Method for Automated Journalism. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*. Linköping University Electronic Press.	Published	Appendix A

The contributions described above have been implemented in the context of a EMBEDDIA news generation system producing news texts from Eurostat datasets. This system is described in Deliverables D5.2 and D5.4. We are currently preparing scientific publications based on the works described both here and in Deliverables D5.2 and D5.4, and as such the source code repository listed below is not yet public. The source code will be made public later with a suitable open source license. As the morphological models used in the systems are licensed under the GPL license, we expect that the systems themselves will also be released as under the GPL license.

Description	URL	Availability
Eurostat news generation system (Sections 2-4, source code)	`https://github.com/EMBEDDIA/eurostat-nlg`	To become public
Neural news structuring code (Section 5, source code)	`https://github.com/EMBEDDIA/eunlg-with-neural-ordering`	To become public

References

Agrawal, H., Chandrasekaran, A., Batra, D., Parikh, D., & Bansal, M. (2016). Sort story: Sorting jumbled images and captions into stories. *arXiv preprint arXiv:1606.07493*.

Bohn, T., Hu, Y., Zhang, J., & Ling, C. (2019, September). Learning sentence embeddings for coherence modelling and beyond. In *Proceedings of the international conference on recent advances in natural language processing (ranlp 2019)* (pp. 151–160). Varna, Bulgaria: INCOMA Ltd. Retrieved from `https://www.aclweb.org/anthology/R19-1018` doi: 10.26615/978-954-452-056-4-018

Caswell, D., & Dörr, K. (2018). Automated journalism 2.0: Event-driven narratives: From simple descriptions to real stories. *Journalism practice*, *12*(4), 477–496.

Chen, X., Qiu, X., & Huang, X. (2016). Neural sentence ordering. *arXiv preprint arXiv:1607.06952*.

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).

Dörr, K. N. (2015). Mapping the field of algorithmic journalism. *Digital journalism*.

Gatt, A., & Krahmer, E. (2018). Survey of the state of the art in natural language generation: Core tasks, applications and evaluation. *Journal of Artificial Intelligence Research*, *61*, 65–170.

Lapata, M. (2006). Automatic evaluation of information ordering: Kendall's tau. *Computational linguistics.*, *32*(4).

Leppänen, L., & Toivonen, H. (2021). A baseline document planning method for automated journalism. In *Proceedings of the 23rd nordic conference on computational linguistics (nodalida).*

Puduppully, R., Dong, L., & Lapata, M. (2019). Data-to-text generation with content selection and planning. In *Proc. 33rd AAAI conference on artificial intelligence.*

Reimers, N., & Gurevych, I. (2019, November). Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (emnlp-ijcnlp)* (pp. 3982–3992). Hong Kong, China: Association for Computational Linguistics. Retrieved from `https://www.aclweb.org/anthology/D19-1410` doi: 10.18653/v1/D19-1410

Reiter, E., & Dale, R. (1997). Building applied natural language generation systems. *Natural Language Engineering*, *3*(1), 57–87.

Thomson, E. A., White, P. R., & Kitley, P. (2008). "Objectivity" and "hard news" reporting across cultures: Comparing the news report in English, French, Japanese and Indonesian journalism. *Journalism studies*, *9*(2), 212–228.

Vinyals, O., Fortunato, M., & Jaitly, N. (2015). Pointer networks. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, & R. Garnett (Eds.), *Advances in neural information processing systems* (Vol. 28, pp. 2692–2700). Curran Associates, Inc. Retrieved from `https://proceedings.neurips.cc/paper/2015/file/29921001f2f04bd3baee84a12e98098f-Paper.pdf`

White, P. (1997). Death, disruption and the moral order: the narrative impulse in mass-media 'hard news' reporting. *Genres and institutions: Social processes in the workplace and school*, *101*, 133.

Appendix A: A Baseline Document Planning Method for Automated Journalism

A Baseline Document Planning Method for Automated Journalism

Leo Leppänen
University of Helsinki
Department of Computer Science
leo.leppanen@helsinki.fi

Hannu Toivonen
University of Helsinki
Department of Computer Science
hannu.toivonen@helsinki.fi

Abstract

In this work, we present a method for content selection and document planning for automated news and report generation from structured statistical data such as that offered by the European Union's statistical agency, Eurostat. The method is driven by the data and is highly topic-independent within the statistical dataset domain. As our approach is not based on machine learning, it is suitable for introducing news automation to the wide variety of domains where no training data is available. As such, it is suitable as a low-cost (in terms of implementation effort) baseline for document structuring prior to introduction of domain-specific knowledge.

1 Introduction

Automated generation of news texts from structured data – often referred to as 'automated journalism' (Graefe, 2016; Dörr, 2015; Caswell and Dörr, 2018) or 'news automation' (Linden, 2017; Sirén-Heikel et al., 2019; Dierickx, 2019) – is of great interest to various news producers. It is seen as a way of 'providing efficiency, increasing output and aiding in reallocating resources to pursue quality journalism' (Sirén-Heikel et al., 2019, p. 47). While data-to-text NLG systems are still far from common especially among the smaller, regional news industry players, at least among the larger newsrooms the use of NLG approaches has clearly been established (Fanta, 2017).

While secrecy in the industry makes it difficult to establish the commercial reality as an outsider, the limited available evidence indicates that commercial automated journalism is mostly done using rule-based methods despite a surge of academic interest in increasingly complex neural methods for NLG (e.g. Puduppully et al., 2019; Ferreira et al., 2019): Interviews of news automation users indicate that the employed methods are mostly based on templates (Sirén-Heikel et al., 2019), as are the few open source code repositories of real-world news automation systems (Yleisradio, 2018). Indeed, some NLG industry experts believe that especially end-to-end neural models do not match customer needs at this time (Reiter, 2019).

Contributing factors include a lack of control (Reiter, 2019); issues with hallucination of non-grounded output (Nie et al., 2019; Dušek et al., 2019; Reiter, 2018); the difficulty in surgically correcting any issues identified in trained neural models beyond additional training; as well as the difficulty of establishing what the 'worst case' performance of a neural model is.

In addition, we believe that that while neural NLG methods are theoretically highly transferable, the *practical* transferability of neural NLG solutions to many news domains is limited by a lack of training data. While newsrooms have extensive archives of news text, these are rarely associated with the matching data that is the 'input' for each piece of news text (E.g., MacKová and Sido, 2020, pp. 43–44, Kanerva et al., 2019, p. 247). At the same time, the non-trainable methods for NLG, too, suffer from difficulties in transferability and reusability (Linden, 2017).

In this work, we investigate document planning (selecting what content and in what order should appear in the document) for structured, statistical data-to-text NLG in the context of automated journalism targeting human journalists. We are not in search of a perfect method, but rather something that is relatively easy to implement as a subdomain-independent baseline and which can then be enhanced with domain-specific processing later-on. Such a method would make it easier to introduce automated journalism solutions to completely new subdomains within the larger statistical data domain.

2 Structuring Hard News

When queried for insight into news structure, journalists and academics often recite the concept of the "(inverted) news pyramid", where the news article is structured so that the order in which information appears in the text reflects the journalist's belief about the importance of the piece of information (Thomson et al., 2008). While the precise origin of the structure is not clear (Pöttker, 2003), it has become so prototypical that it is held self-evident in the journalistic trade literature: "*Every journalist knows how to write a traditional news text: start with the most important thing and continue until you have either said everything relevant or the space reserved for the story runs out*" (Sulopuisto, 2018, translated from Finnish).

A more rigorous analysis of the structures employed in 'hard' news is presented by White (1997), who argues that hard news articles have an 'orbital' structure consisting of a *nucleus* which represents the main point of the article and *satellites* that give context and additional information about the nucleus. White (1997) assigns the role of the nucleus to the combination of the headline and the lead paragraph of the article, and describes the subsequent paragraphs as the satellites. White (1997) identifies five possible relations between a satellite and the nucleus: elaboration, cause-and-effect, justification, contextualization and apprisal. Thomson et al. (2008), in turn, identify that the satellites can elaborate, reiterate, describe causes or consequences, contextualize or provide additional assessment. An important observation is that – as indicated by 'orbital' – these satellites are relatively freely reorderable without affecting readability or meaning. Together, these two observations indicate that a good document plan for hard news (1) prioritizes more newsworthy items and (2) contains some overarching theme (exemplified by the nucleus) so that the text as a whole is coherent, i.e. the satellites are in some way related to the nucleus.

The relations identified by White (1997) and Thomson et al. (2008) are highly similar to those identified in the more general Rhetorical Structure Theory (RST) (Mann and Thompson, 1988), which uses similar nucleus-satellite terminology. However, whereas White (1997) and Thomson et al. (2008) analyze news text on the level of paragraphs, RST can be applied on a more fine-grained level to much shorter text spans. As RST shows that similar relations can be applied on a sub-paragraph level, we hypothesize that a reasonably approximation of a news article might be constructed by applying White's (1997) orbital theory also *within* paragraphs, by considering the first *sentence* of the paragraph a nucleus, and the others as satellites.

Importantly, we interpret the orbital theory of news structuring to suggest that – as the satellites are freely orderable – the actual *type* of relation is not as important for document planning as knowing that *some* relation exists between the satellite and the nucleus. We hypothesize that while identifying whether a specific (RST) relation exists between two arbitrary pieces of information requires domain knowledge, an approximation of whether two arbitrary pieces of information are related in *some* way could be obtained by inspecting their similarity in a domain-independent fashion.

That is, we expect that a piece of information regarding the US health care funding in 2020 is more likely to be related in *some way* to a piece of information discussing the US health care funding in 2020 than to another piece of information discussing the health care funding in Sweden in 1978. If a heuristic or similarity measure identifying such relations could be identified, it could be used together with some estimate of newsworthiness to construct paragraph and document plans that seek to maximize both the key aspects identified above: newsworthiness and the relatedness of the content.

As noted in the introduction, there is a distinction between the theoretical and the practical transferability of neural processing methods. We believe that a good baseline document planning and content selection approach should avoid the need for training data present in the many of recently proposed document planning and content selection approaches. This rules out as unsuitable most recent work that are based on learning from an aligned corpus of data and human-written texts, such as Angeli et al. (2010), Konstas and Lapata (2013), Wiseman et al. (2017), Zhang et al. (2017), Li and Wan (2018), Dou et al. (2018) and Puduppully et al. (2019).

Outside of these trainable approaches, to our knowledge, most other document planning approaches are based on '*hand-engineered*' (Konstas and Lapata, 2013), domain-specific methods. A highly relevant survey of various document planning methods is presented by Gkatzia (2016). While these previous works are – to at least some degree – domain-specific, they establish concepts

and ideas that are highly relevant for our goal. Both Hallett et al. (2006) and Gatt et al. (2009) describe a core set of information, called 'summary spine' or 'key events', that they hold as more important than the rest of the available information. They, as well as Banaee et al. (2013), also employ a numeric estimate of importance. Demir et al. (2010) identify that content already selected for inclusion in the document plan affects how well suited so-far unselected content is for inclusion. Sripada et al. (2003) identify Gricean maxims (Grice, 1975) as providing requirements for document planning and content selection.

3 Context

Our work on document planning is done in the context of a series of data-to-text NLG applications producing short highlights of structured statistical data. Importantly, the applications are intended to be deployed in contexts where they must be able to produce texts highlighting between 10 and 30 data points from datasets measured in 100.000s of data points. The resulting texts are intended to both alert journalists to potential news and to provide them with a starting place from which to write the final news text.

Our system, adapted from Leppänen et al. (2017a), is based on a pipeline of components with dedicated responsibilities similar to those described by Reiter and Dale (2000) and Reiter (2007). For this work, the relevant part of the architecture is the Document Planner component. This component receives as input two sets of *message* data structures, an example of which is shown in Table 1.[1] The messages are extracted automatically from tables of statistical data obtained from Eurostat.

The *core set* contains messages that are known to be highly relevant to the generation task. Unlike the "summary spine' of Hallett et al. (2006), the set is unlinked and unordered, and not all members of the set are guaranteed to be included in the document plan. The *expanded set*, contains messages that *can* be, but are not guaranteed to be, relevant for the document. Expressed using the terminology from Section 2, we assume that only messages in the core set can be nuclei, while messages from either set can be satellites.

These core and expanded sets are determined automatically from user input. When requesting a new text, the user of the system must define a dataset the text is to be generated from, for example the consumer price data available from Eurostat. This dataset is then divided into the core set and the expanded set by the user when they select what country the generated text should focus on. For example, if the user were to select that the text should discuss French consumer prices, the core set would contain all data from the consumer price dataset that pertains directly to France, while the rest of the consumer price dataset (including data pertaining to the UK, Finland, Croatia, etc.) would be set as the expanded set.

We estimate each message's 'newsworthiness' using the Interquartile Range based method described by Leppänen et al. (2017b) with the values scaled to have mean 0 and standard deviation 1 for the purposes of this computation. The resulting value is conceptually similar to 'importance' of Gatt et al. (2009) and 'risk' of Banaee et al. (2013). The IQR based method compares each data point in turn to a larger distribution, giving it higher scores the further it is from the area between the first and the third quartile of the larger distribution. Values between the quartiles are given a minimal, uniform, score that is dependent on the shape of the distribution. In other words, higher IQR values indicate that the value is more of an outlier compared to the rest of related data in the dataset. As such, it captures a degree of 'unexpectedness', which is an important aspect of newsworthiness (Galtung and Ruge, 1965).

We do not use the domain-specific parts of the method described by Leppänen et al. (2017b). That is, we make no value judgement of whether messages pertaining to French consumer prices are more newsworthy than messages pertaining to Croatian consumer prices, nor do we make judgements of whether changes in the price of education are more or less newsworthy than changes in the price of alcohol and tobacco. However, we do weight the scores so that messages with the `timestamp` field being closer to present receive higher weights, as recency is an important aspect of newsworthiness. While we have described our method for computing the `newsworthiness` value in some detail, we emphasize that for the rest of this article we only assume that the `newsworthiness` values are non-negative and that higher values indicate higher newsworthiness.

More crucially for the method described be-

[1] The concrete implementation details are somewhat more complex. We omit details irrelevant for this work.

low, we specify that the `value_type` fields (which describe how the messages' values are to be interpreted) contain members of a hierarchical taxonomy of data types represented as colon-separated hierarchies of labels. For example, the `value_type` field value `health:cost:hc2:mio_eur` would indicate that the number in the `value` field is the amount of money (`cost`), measured in millions of euros (`mio_eur`), spent by some nation (as defined by the `location` and `location_type` fields) on rehabilitative care (`hc2`) in some time period (as defined by the `timestamp` and `timestamp_type` fields) and that this is part of the larger health care topic (`health`). In our case, these labels are automatically established from the headers of the input data tables.

The goal of document structuring is to produce a three-level tree-structure with ordered children. The root node corresponds to the document as a whole and the mid-level structures correspond to paragraphs. The leaves are the messages selected for inclusion in the document. While the messages have not yet, at this stage, been associated with any linguistic structures, they can be conceptualized as being phrases or very short sentences. We are thus concurrently determining both the content and the structure the document.

We emphasize that our applications are employed in domains where they must be able to select some 10-30 messages from a pool of potential messages numbering in 100,000s. Given infinite computational resources, it would be preferential to construct all possible document plans and then score them in some fashion. This, however, is infeasible given the size of the search space. Previously, other authors have employed, for example, stochastic searches with significantly smaller search spaces (Mellish et al., 1998). Indeed, some kind of a beam search approach could be very useful in smartly searching a subset of the search space. However, we have thus far been unable to identify a document-level metric that adequately balances the 'total amount of newsworthiness' in a text with the length of the text, a requirement for beam search.

4 Research Objective

Based on the above considerations, our main goal is to identify a widely applicable method for content selection and document planning that matches the following requirements:

REQ1: The method needs to be highly performant

REQ2: The method should not be dependent on domain knowledge

REQ3: The document should have a theme

REQ4: The document should have multiple paragraphs but not be excessively long

REQ5: The paragraphs should have distinct themes related to the document theme

REQ6: The paragraph themes should be newsworthy in their own right

REQ7: The paragraphs should not be excessively long or short

REQ8: All messages should relate to the paragraph theme

REQ9: All messages should be newsworthy

REQ10: Within each paragraph, the messages should be presented in an order that produces a coherent narrative

Again, we emphasize that our goal is not to identify a method that is optimal for any specific scenario, but rather to determine a baseline method that is *adequate* for a broad spectrum of applications and sub-domains.

5 A Baseline Approach to Document Planning

Optimally, we would wish to produce some sort of a *globally optimal* document plan. However, as discussed above, this would entail significant computational costs and require a scoring function applicable to the document as a whole. As such, we propose a method for producing document plans in a greedy, linear, and iterative fashion. At every stage, decisions are made considering only a limited local context, thus avoiding the need for a method of determining the global quality of the document plan, thus fulfilling REQ1 ('The method needs to be highly performant').

The document's overall theme, in our use case, is selected by the user who initiates the generation task. In initiating the task, the users selects both a dataset and a focus location. The generation process then derives the *core messages* and *expanded messages* sets (the inputs to the Document Planner, see Section 3) so that both sets discuss the dataset

Field	Description	Example value
`where`	What location the fact relates to	Finland
`where_type`	What the type of the location is	country
`timestamp`	The time (or time range) the fact relates to	2020M05
`timestamp_type`	The type of the timestamp	month
`value`	A (usually) numeric value	0.01
`value_type`	Interpretation of `value`	cphi:hicp2015:cp-hi02:rt01
`newsworthiness`	An estimate of how newsworthy the message is	1

Table 1: An example of a message. The hypothetical message states that in the fifth month of 2020, in Finland, the consumer price index, using the year 2015 as the start of the index, of alcoholic beverages and tobacco changed by 0.01 points with respect to the value of the index during the previous month.

indicated by the user (i.e. messages from other datasets are not generated) and that the core set contains messages pertaining to the user's indicated focus location, while messages pertaining to all other locations are in the expanded set. This fulfills REQ3 ('The document should have a theme'). This step is also independent of the specific subdomain, thus fulfilling REQ2 ('The method should not be dependent on domain knowledge'). This step thus fulfills all the relevant requirements. Next, we'll describe how both the first and subsequent paragraphs can be planned in a way consistent with the requirements defined above.

5.1 Planning the First Paragraph

At the start of the document planning process, we select the most newsworthy message from the *core messages* set to act as the nucleus (n_1) of the first paragraph (p_1). This nucleus establishes the theme of the first paragraph as follows: We inspect the `value_type` field of this first nucleus n_1, and retrieve a prefix `Prefix`(n_1). The prefix is the least amount of colon-separated labels wherein the total amount of prefixes in the core set is greater than the minimal amount of paragraphs a document can have, in our case two. In our case, as a consequence of our label hierarchy, this is always the first three colon-separated units. For the message shown in Table 1, the prefix would thus be `cphi:hicp2015:cp-hi02`, meaning that the first paragraph's theme would be the prices of alcoholic beverages and tobacco. This fulfills REQ5, 'the paragraphs should have distinct themes related to the document theme' for the first paragraph.

Next, the first paragraph is completed with satellites from the union of the *core messages* and the *expanded messages* sets. These satellites are initially filtered so that only messages that have the same prefix as the nucleus n_i are considered in paragraph p_i to fulfill REQ8 ('All messages should relate to the paragraph theme'). The satellites are then selected in a linear, greedy, and iterative manner to fulfill REQ1.

For selecting the k'th satellite to a partially constructed paragraph already containing $k-1$ satellites and one nucleus, we consider both the newsworthiness of the available messages (REQ9), as well as how well they would fit the already constructed segment (REQ8). Observing only the newsworthiness would produce a highly incoherent narrative, whereas focusing only on the narrative risks leaving out highly important information.

Following the reasoning in Section 2, we assume that two subsequent messages are more likely to form a good narrative if they are similar. As such, we need a method for weighing the message's newsworthiness by the similarity of the message to the last message of the under-construction paragraph, thus balancing the requirements of REQ8 and REQ9. In terms of the message objects described in Table 1, it seems to us that the intuitive aspects of similarity are related to the degree of similarity within the 'meta' fields such as `timestamp`, `location` and `value_type`.

For the `timestamp` and `location` fields, we can state that two messages that have identical values in the fields are more similar that two messages that are otherwise the same but have distinct values for said fields. We call this the *contextual* similarity of the messages, and the fields the *contextual fields* (F_c), as these fields provide us access to the larger context in which the `value` and `value_type` fields can be interpreted. Contextual similarity captures the notion that it is likely better to follow a fact about French healthcare spending in 2020 with another piece of information about France in 2020,

rather than about Austria in 1990.

In more precise terms, we propose the following weighing scheme for contextual similarity: The similarity $sim_c(A, B)$ of two messaged A and B is the product of weights $w_f > 1$ for each field f among the contextual fields F_c, where both A and B have the same value for the field:

$$sim_c(A, B) = \prod_{\{f \in F_c | A.f = B.f\}} w_f \quad (1)$$

This value strictly increases as more fields are shared between A and B. We explicitly define the similarity to be zero if there are no fields f where A and B share a value. If w_f is a uniform value for all fields f, this scheme is completely domain-agnostic. Setting different weights w_f for each field $f \in F_c$ allows for encoding some domain knowledge about which fields are the most important for the text, thus providing a method for producing more tailored texts at the cost of slightly violating REQ2. In our case study, we set $w_{timestamp} = 1.1$ and $w_{location} = 1.5$.

The above consideration of similarity still ignores valuable information available from the `value_type` field, which describes how the value in the `value` field is to be interpreted. Denoting `health:cost:hc2:mio_eur` (the cost of rehabilitative care in millions of euros) by T_1, consider its similarity to T_2 = `health:cost:hc2:eur_hab`, the cost of rehabilitative care as euros per inhabitant, and T_3 = `health:cost:hc41:mio_eur`, the cost of health care related imaging services in millions of euros. Intuitively, T_1 and T_2 are thematically closer than T_1 and T_3. We model this similarity between two facts A and B simply as

$$sim_t(A, B) = \frac{1}{s(A, B)} \quad (2)$$

where $s(A, B)$ is the length – in colon-separated units – of the unshared suffix between A and B's `value_type` fields. That is, $s(T_1, T_2) = 1$ whereas $s(T_1, T_3) = 2$. We specify that $sim_t(\cdot, \cdot)$ is zero for all pairs without any shared prefix.

Our formulation of $sim_t(\cdot, \cdot)$ was influenced by the observation that in our context the messages' `value_type` values have a constant number of colon-separated segments. In cases where the lengths of the `value_type` values differ, an alternative formulation of

$$sim'_t(A, B) = \frac{2p(A, B)}{\ell(A) + \ell(B)} \quad (3)$$

where $\ell(\cdot)$ provides the length of the `value_type` value, and $p(\cdot, \cdot)$ is the length of shared *prefix* between A and B, both measured as colon-separated units, might be preferable if also more complex.

When considering whether the k'th satellite s_i^k of paragraph p_i should be a specific candidate $c \in C$, where C is all so far unused messages, we can combine the similarity metrics with the newsworthiness of c into a general fitness value as follows:

$$\begin{aligned} fit(c, x) = c.newsworthiness \\ \times sim_c(c, x) \\ \times sim_t(c, x) \\ \times set\_penalty(c) \end{aligned}$$

The $set\_penalty(c)$ factor depends on whether the message originates from the *core messages* set, or the *extended messages* set. For messages originating from the core message set, the penalty is 1. For messages originating from the extended messages set, the penalty is $\frac{1}{dist+1}$, where $dist$ is the distance from the previous core message.

The final score describing how good of an addition c would be as the kth satellite of the ith paragraph s_i^k is then obtained by taking the average of fitnesses of c in relation to both the nucleus n_i and the previous satellite s_i^{k-1} by computing:

$$score(c, n_i, s_i^{k-1}) = \frac{fit(c, n_i) + fit(c, s_i^{k-1})}{2}$$

This maximizes the newsworthiness of the paragraph's contents (fulfilling REQ9, 'all messages should be newsworthy'), while also enforcing relatedness to the theme of the paragraph (fulfilling REQ8, 'all messages should relate to the paragraph theme') by measuring against the nucleus and with the inclusion of the *set_penalty*. By continuously measuring against the previously selected satellite, the procedure also allows for interludes to e.g. discuss highly newsworthy information related to but not strictly about the paragraph's main topic, or 'thematic drift'. It thus fulfills REQ10 ('Within each paragraph, the messages should be presented in an order that produces a coherent narrative') while also paying attention to the pyramid model of news (See Section 2).

Using $score$, the highest scoring candidate $c_{top} = \arg\max_{c \in C} score(c, n_i, s_i^{k-1})$ is then compared to both an absolute threshold t_{abs} and the newsworthiness of the nucleus n_i multiplied by relative threshold value t_{rel}. Provided that the

maximal paragraph length has not been reached, the top candidate message c_{top} is appended to the paragraph p_i as the k'th satellite s_i^k in the document plan provided that either $score(c_{top}, n_i, s_i^{k-1}) \geq t_{abs}$ or $score(c_{top}, n_i, s_i^{k-1}) \geq t_{rel} \times n_i.newsworthiness$.

These thresholds ensure that the paragraph does not stray into minutiae, whether considered in absolute terms or in relation to the nucleus of the paragraph. In cases where the minimum paragraph length has not been reached, the thresholds are ignored and the top candidate is always appended. This accounts for REQ7 ('The paragraphs should not be excessively long or short').

The above considerations take into account several free parameters, namely the maximal and minimal paragraph lengths as well as the threshold values t_{rel} and t_{abs}. In our case study, we selected the minimal and maximal paragraph lengths as 2 and 5 messages empirically by trialing out various values and observing the resulting texts. These should, naturally, be based on the genre of text and the target audience. For the threshold values we selected 0.2 and 0.5, respectively, using the same method as with the paragraph lengths above. Both the thresholds and the minimal and maximal paragraph lengths should be viewed as (manually) tuneable hyperparameters.

5.2 Planning Subsequent Paragraphs

We then proceed to generate further paragraphs in a manner highly similar to that used when planning the first paragraph. The only distinction is that, when selecting the nucleus n_i for a subsequent paragraph p_i, we obtain the message from the *core messages* set with a highest newsworthiness value that has a prefix (theme) not yet discussed among the previously planned paragraphs p_1 - p_{i-1}:

$$n_i = \arg\max_{c \in C} c.newsworthiness \tag{4}$$

where

$$C = \Big\{ c \in CoreMessages | \texttt{Prefix}(c) \notin \{\texttt{Prefix}(n_k) | k \in [1..i-1]\} \Big\} \tag{5}$$

This ensures that the different paragraphs are highly newsworthy, thus fulfilling REQ6, while also fulfilling REQ5 for having distinct themes for the different paragraphs.

As when constructing the subsequent paragraphs, the total length of the document also needs to be considered. To fulfill REQ4 ('The document should have multiple paragraphs but not be excessively long'), we employ a variation of the method described in the previous section for ending individual paragraphs. A maximal length (in our case, 3 paragraphs) ensures that the document is not allowed to grow beyond reason, whereas a minimal length (for us, 2 paragraphs) ensures that the document is not unreasonably short. After the minimal length has been reached (but not yet the maximal length), a new paragraph is only started if the nucleus of the potential paragraph has a newsworthiness value that is at least 30 % of the newsworthiness value of the first nucleus of the document. This, as with the satellites, ensures that the the document does not stray into minutiae, balancing REQs 4 and 6. the maximal and minimal lengths, as well as the 30 % threshold, were determined by manual fine-tuning and should be viewed as tuneable hyperparameters.

6 Evaluation

The method described above was implemented in a larger NLG application producing news alerts for journalists from datasets provided by Eurostat. A variation of the same application was also developed with a simplified document planner. In this simplified planner, the planner always selects the maximally newsworthy available message as the message without any early stopping threshold. Nuclei are selected from the core messages set, while satellites can be from either set. Contrasting our proposed method with this simplified method enables us to evaluate the importance of narrative coherence in the generated texts. The larger application is multilingual, but the evaluation was conducted using English language texts.

Three experts were recruited from the Finnish News Agency STT, a national European news agency, to evaluate documents on the consumer price indices in five different European nations. For all nations, the judges were shown variants produced by both our proposed method and the simplified method. One of the selected countries is the country the news agency is based in, with the assumption that the judges would have high amounts of world knowledge they would be able to use in evaluating these texts. Another variant pair describes a country that is both relatively small and geographically remote (but still within EU), with the assumption that the journalists are unlikely to

Consumer Prices in Estonia

In June 2020, in Estonia, the monthly growth rate of the harmonized consumer price index for the category 'education' was 30.8 points. It was 30.7 percentage points more than the EU average. In July 2020, it was 0.4 percentage points less than the EU average. It was -0.4 points. In May 2020, the yearly growth rate of the harmonized consumer price index for the category 'education' was -20.5 points. It was 21.9 percentage points less than the EU average.

In August 2020, the monthly growth rate of the harmonized consumer price index for the category 'housing, water, electricity, gas and other fuels' was 2.5 points. It was 2.3 percentage points more than the EU average. In North Macedonia, it was 3 percentage points more than the EU average. It was 3.2 points. Estonia had the 3rd highest monthly growth rate of the harmonized consumer price index for the category 'housing, water, electricity, gas and other fuels' across the observed countries. In Sweden, the monthly growth rate of the harmonized consumer price index for the category 'housing, water, electricity, gas and other fuels' was 3.1 points.

Figure 1: Example output regarding Eurostat statistics on consumer prices. The text contains 12 messages, selected from among 207,210 messages available during generation.

have much world knowledge about this country's consumer prices. The three other countries were selected from among those bordering the first country, with the assumption that the journalists would have some, but not much, world knowledge relating to these countries. The final output texts were not inspected prior to selecting the countries.

All of the texts used in the evaluation were generated from a copy of the same underlying Eurostat dataset, entitled 'Harmonised index of consumer prices - monthly data [ei_cphi_m]'[2] downloaded in September 2020. It contains country-level data regarding the harmonized consumer prices indices, and their change over time, for various EU nations starting from January 1996. We preprocess the data by adding monthly rankings (i.e. determine what country had the greatest, the second greatest, etc. value for a specific index category during any specific month) and comparisons to the EU average values.

As the evaluation was focused on document planning and content selection, the larger system was simplified in some respects, e.g., to not conduct complex aggregation. This was done to minimize the effect of later stages of the generation process on the evaluation. As a result, the language in the evaluated documents was relatively stilted, as exemplified by Figure 1. The only manual alteration was the addition of headings to indicate the texts' intended themes.

The judges did not receive any direct compensation but their employer, the news agency, is a member of the EU-wide EMBEDDIA research project within which parts of this work was conducted. The evaluations were conducted online. The judges were first provided with some basic information on the type of documents they were to read (i.e. that the texts are intended to be news alerts for journalists, rather than publication ready news texts), the length of the task, etc. All instructions were in the judges' native language, in this case Finnish. The judges were not told which texts were produced by which variants nor how many variants were being tested. Following this, the judges were shown the documents one by one. For each document, the judges were asked to indicate their agreement with the following statements (translated from Finnish):

Q1: The text matches the heading

Q2: The text is coherent

Q3: The text lacks some pertinent information

Q4: The text contains unnecessary information

Q5: The text has a suitable length

For Q1–Q4, the judges indicated their agreement on a 7-point Likert scale ranging from 1 ('completely disagree') to 7 ('completely agree'). For Q5, the answers were provided on 5-point scale ranging from 1 ('clearly too short') to 3 ('length is suitable') to 5 ('clearly too long'). In addition, the judges were able to provide textual feedback for each individual text, as well as for the evaluation task as a whole. The judges' answers to Q1 – Q5, are aggregated in Table 2.

The results indicate that the proposed method statistically significantly increases the document's coherence (Q2, mean 4.33 vs. 1.60, median 5 vs 2), the matching of the document's content to the document's theme (Q1, mean 4.40 vs. 1.80, median 5 vs 2), and produces documents of more suitable length (Q5, mean 2.93 vs. 4.07, median 3 vs 4, with 3 being best). The proposed method also seems

[2] Available for download and browsing from `http://appsso.eurostat.ec.europa.eu/nui/show.do?dataset=ei_cphi_m`

	Our method			Baseline			
Statement	Median	Mean	SD.	Median	Mean	SD.	p_{MWU}
Q1 (1–7, ↑)	5	4.40	1.64	2	1.80	0.41	< 0.001*
Q2 (1–7, ↑)	5	4.33	1.76	2	1.60	0.51	< 0.001*
Q3 (1–7, ↓)	4	4.47	1.81	6	5.80	1.42	0.049
Q4 (1–7, ↓)	5	5.13	1.55	6	6.33	0.62	0.024
Q5 (1–5, 3 best)	3	2.93	0.59	4	4.07	0.70	< 0.001*

Table 2: Results obtained during the evaluation. Parentheses indicate answer ranges and whether the higher (↑), lower (↓) or middle values are to be interpreted as the best. The p_{MWU} column contains the (uncorrected) p-value of a two-sided Mann-Whitney U test. An asterisk indicates the p-value is statistically significant also after applying a Bonferroni correction to account for multiple tests.

to result in less unnecessary information being included in the document (Q4, mean 5.13 vs 6.33, median 5 vs 6), and in the text missing less necessary information (Q3, mean 4.47 vs 5.80, median 4 vs 6), but these effects are not statistically significant after correcting for multiple comparisons with the Bonferroni correction. We hypothesize this difference would become significant in a larger-scale evaluation.

The free-form textual feedback provided by the judges, as expected, indicates that the texts could be further improved. For example, in the case of the text shown in Figure 1, the judges called for a sentence explicitly noting that North Macedonia had the highest monthly growth rate. In addition, they noted it might be better to produce distinct, even shorter, texts as 'news alerts' while reserving the evaluated texts for use as a starting point when the journalist starts writing.

7 Conclusions

In this work, we have identified a need for, and proposed, a widely applicable baseline document planning method for generating journalistic texts from statistical datasets. Our method is based on observations on the similarities between the orbital theory of news structure (White, 1997) and Rhetorical Structure Theory (Mann and Thompson, 1988). While our proposed method is likely to fall short of the performance of subdomain-specific planning methods, results indicate that it achieves adequate performance while fulfilling a set of requirements identified based on the larger application domain of news generation.

Acknowledgements

This work is supported by the European Union's Horizon 2020 research and innovation program under grant agreement No 825153, project EMBEDDIA (Cross-Lingual Embeddings for Less-Represented Languages in European News Media), and grant agreement No 770299, project NewsEye (A Digital Investigator for Historical Newspapers).

References

Gabor Angeli, Percy Liang, and Dan Klein. 2010. A simple domain-independent probabilistic approach to generation. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 502–512.

Hadi Banaee, Mobyen Uddin Ahmed, and Amy Loutfi. 2013. Towards NLG for physiological data monitoring with body area networks. In *14th European Workshop on Natural Language Generation, Sofia, Bulgaria, August 8-9, 2013*, pages 193–197.

David Caswell and Konstantin Dörr. 2018. Automated journalism 2.0: Event-driven narratives: From simple descriptions to real stories. *Journalism practice*, 12(4):477–496.

Seniz Demir, Sandra Carberry, and Kathleen F. McCoy. 2010. A discourse-aware graph-based content-selection framework. In *Proceedings of the 6th International Natural Language Generation Conference*.

Laurence Dierickx. 2019. Why news automation fails. In *Computation+ Journalism Symposium, Miami, FL*.

Konstantin Nicholas Dörr. 2015. Mapping the field of algorithmic journalism. *Digital journalism*.

Longxu Dou, Guanghui Qin, Jinpeng Wang, Jin-Ge Yao, and Chin-Yew Lin. 2018. Data2text studio: Automated text generation from structured data. In

Proc. 2018 Conference on Empirical Methods in Natural Language Processing.

Ondřej Dušek, David M Howcroft, and Verena Rieser. 2019. Semantic noise matters for neural natural language generation. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 421–426.

Alexander Fanta. 2017. Putting Europe's robots on the map: automated journalism in news agencies. *Reuters Institute Fellowship Paper*, pages 2017–09.

Thiago Castro Ferreira, Chris van der Lee, Emiel van Miltenburg, and Emiel Krahmer. 2019. Neural data-to-text generation: A comparison between pipeline and end-to-end architectures. *arXiv preprint arXiv:1908.09022*.

Johan Galtung and Mari Holmboe Ruge. 1965. The structure of foreign news: The presentation of the Congo, Cuba and Cyprus crises in four Norwegian newspapers. *Journal of peace research*, 2(1):64–90.

Albert Gatt, Francois Portet, Ehud Reiter, Jim Hunter, Saad Mahamood, Wendy Moncur, and Somayajulu Sripada. 2009. From data to text in the neonatal intensive care unit: Using NLG technology for decision support and information management. *Ai Communications*, 22(3):153–186.

Dimitra Gkatzia. 2016. Content selection in data-to-text systems: A survey. *arXiv preprint*. Available at `https://arxiv.org/abs/1610.08375`.

Andreas Graefe. 2016. Guide to automated journalism.

Herbert P Grice. 1975. Logic and conversation. In *Speech acts*, pages 41–58. Brill.

Catalina Hallett, Richard Power, and Donia Scott. 2006. Summarisation and visualisation of e-health data repositories. In *UK E-Science All-Hands Meeting*.

Jenna Kanerva, Samuel Rönnqvist, Riina Kekki, Tapio Salakoski, and Filip Ginter. 2019. Template-free data-to-text generation of Finnish sports news. In *Proceedings of the 22nd Nordic Conference on Computational Linguistics*, pages 242–252, Turku, Finland. Linköping University Electronic Press.

Ioannis Konstas and Mirella Lapata. 2013. Inducing document plans for concept-to-text generation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1503–1514.

Leo Leppänen, Myriam Munezero, Mark Granroth-Wilding, and Hannu Toivonen. 2017a. Data-driven news generation for automated journalism. In *Proceedings of the 10th International Conference on Natural Language Generation*, pages 188–197.

Leo Leppänen, Myriam Munezero, Stefanie Sirén-Heikel, Mark Granroth-Wilding, and Hannu Toivonen. 2017b. Finding and expressing news from structured data. In *Proceedings of the 21st International Academic Mindtrek Conference*, pages 174–183. ACM.

Liunian Li and Xiaojun Wan. 2018. Point precisely: Towards ensuring the precision of data in generated texts using delayed copy mechanism. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1044–1055, Santa Fe, New Mexico, USA. ACL.

Carl-Gustav Linden. 2017. Decades of Automation in the Newsroom: Why are there still so many jobs in journalism? *Digital Journalism*, 5(2):123–140.

Veronika MacKová and Jakub Sido. 2020. The robotic reporter in the Czech News Agency: Automated journalism and augmentation in the newsroom. *Communication Today*, 11(1):36–53.

William C Mann and Sandra A Thompson. 1988. Rhetorical structure theory: Toward a functional theory of text organization. *Text-Interdisciplinary Journal for the Study of Discourse*, 8(3):243–281.

Chris Mellish, Alistair Knott, Jon Oberlander, and Mick O'Donnell. 1998. Experiments using stochastic search for text planning. In *Natural Language Generation*.

Feng Nie, Jin-Ge Yao, Jinpeng Wang, Rong Pan, and Chin-Yew Lin. 2019. A simple recipe towards reducing hallucination in neural surface realisation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2673–2679.

Ratish Puduppully, Li Dong, and Mirella Lapata. 2019. Data-to-text generation with content selection and planning. In *Proc. 33rd AAAI Conference on Artificial Intelligence*.

Horst Pöttker. 2003. News and its communicative quality: The inverted pyramid—when and why did it appear? *Journalism Studies*, 4(4):501–511.

Ehud Reiter. 2007. An architecture for data-to-text systems. In *Proceedings of the Eleventh European Workshop on Natural Language Generation*, pages 97–104. Association for Computational Linguistics.

Ehud Reiter. 2018. Hallucination in neural NLG. `https://ehudreiter.com/2018/11/12/hallucination-in-neural-nlg/`. Accessed: 2020-03-02.

Ehud Reiter. 2019. ML is used more if it does not limit control. `https://ehudreiter.com/2019/08/15/ml-limits-control/`. Accessed: 2020-07-25.

Ehud Reiter and Robert Dale. 2000. *Building natural language generation systems*. Studies in Natural Language Processing. Cambridge University Press.

Stefanie Sirén-Heikel, Leo Leppänen, Carl-Gustav Lindén, and Asta Bäck. 2019. Unboxing news automation: Exploring imagined affordances of automation in news journalism. *Nordic Journal of Media Studies*, 1(1):47–66.

Somayajulu G Sripada, Ehud Reiter, Jim Hunter, and Jin Yu. 2003. Generating English summaries of time series data using the Gricean maxims. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 187–196.

Olli Sulopuisto. 2018. Uutisia kortti kerrallaan. *Suomen Lehdistö*. `https://suomenlehdisto.fi/uutisia-kortti-kerrallaan/`.

Elizabeth A Thomson, Peter RR White, and Philip Kitley. 2008. "Objectivity" and "hard news" reporting across cultures: Comparing the news report in English, French, Japanese and Indonesian journalism. *Journalism studies*, 9(2):212–228.

Peter White. 1997. Death, disruption and the moral order: the narrative impulse in mass-media 'hard news' reporting. *Genres and institutions: Social processes in the workplace and school*, 101:133.

Sam Wiseman, Stuart Shieber, and Alexander Rush. 2017. Challenges in data-to-document generation. In *Proc. 2017 Conference on Empirical Methods in Natural Language Processing*.

Yleisradio. 2018. Avoin voitto. `https://github.com/Yleisradio/avoin-voitto`.

Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D. Manning. 2017. Position-aware attention and supervised data improve slot filling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 35–45, Copenhagen, Denmark. ACL.