HTSS: A novel hybrid text summarisation and simplification architecture

Farooq Zaman[a], Matthew Shardlow[b], Saeed-Ul Hassan[*,a], Naif Radi Aljohani[c], Raheel Nawaz[b]

[a] *Information Technology University, 346-B, Ferozepur Road, Lahore, Pakistan*
[b] *Manchester Metropolitan University, Manchester, United Kingdom*
[c] *Faculty of King Abdulaziz University, Jeddah, Kingdom of Saudi Arabia*

A R T I C L E I N F O

Keywords:
Summarisation
Simplification
Pointer generator
Deep learning

A B S T R A C T

Text simplification and text summarisation are related, but different sub-tasks in Natural Language Generation. Whereas summarisation attempts to reduce the length of a document, whilst keeping the original meaning, simplification attempts to reduce the complexity of a document. In this work, we combine both tasks of summarisation and simplification using a novel hybrid architecture of abstractive and extractive summarisation called HTSS. We extend the well-known pointer generator model for the combined task of summarisation and simplification. We have collected our parallel corpus from the simplified summaries written by domain experts published on the science news website EurekaAlert (www.eurekalert.org). Our results show that our proposed HTSS model outperforms neural text simplification (NTS) on SARI score and abstractive text summarisation (ATS) on the ROUGE score. We further introduce a new metric (CSS_1) which combines SARI and Rouge and demonstrates that our proposed HTSS model outperforms NTS and ATS on the joint task of simplification and summarisation by 38.94% and 53.40%, respectively. We provide all code, models and corpora to the scientific community for future research at the following URL: https://github.com/slab-itu/HTSS/.

1. Introduction

Text simplification and summarisation are well established tasks within the field of natural language processing (NLP). Text summarisation is the task of condensing a given document into a required size while preserving the information contained in the original source document (Collins, Augenstein, & Riedel, 2017; Nikolov, Pfeiffer, & Hahnloser, 2018). On the other hand, text simplification is the task of simplifying a text so that it is easy to read, while keeping the original meaning preserved (Alva-Manchego, Martin, Scarton, & Specia, 2019; Dong, Li, Rezagholizadeh, & Cheung, 2019b; Kriz et al., 2019; Nisioi, Štajner, Ponzetto, & Dinu, 2017a; Surya, Mishra, Laha, Jain, & Sankaranarayanan, 2019). At the intersection of these two tasks lies an understudied, yet very concrete problem: the generation of simplified summaries. In this task, a lengthy technical document is taken and converted into a concise and easy to read format. At present, this is a costly manual process requiring domain experts to read, evaluate, summarise and simplify a long technical document. This takes place in diverse sectors from legal policy to medical research, or the production of news for lay readers from scientific articles.

The popular science website 'Eureka Alert' is one such example, where a team of authors synthesise information from scientific articles and convert these into easy to read summaries for their audience, applying simplification and summarisation. If they only summarised, the articles would be too difficult for a non-expert, containing technical scientific terms and jargon. Similarly, if an author only simplified a scientific article, the format would be too long and dense to engage with. The authors must perform both tasks, as must a system which is intended to automate this process.

Consider the following example snippet from a scientific article:

> Inhibitory luminopsins: genetically-encoded bioluminescent opsins for versatile scalable and hardware-independent optogenetic inhibition

An author for Eureka Alert, may simplify the snippet to read as follows:

> Tools for illuminating brain function make their own light

In the above example the source document consists of many complex and domain specific terms such as 'luminopsins', 'bioluminescent' and 'optogenetic', yet the summary is easy to read and free of domain specific terms.

The main objective of this work is to develop an approach for the combined task of text summarisation and simplification, which will generate a simplified summary of a given document (in this case, a scientific article). We refer to this approach as Hybrid Text Summarisation and Simplification or HTSS. In this work, we proposed a novel hybrid model for the combined task of text summarisation and text simplification. For this task there is no such corpus readily available, so we created a new parallel corpus for the combined task of summarisation and simplification, for this we scraped simplified summaries from Eurika Alert[1], then processed each simplified summary and linked it back to the original scientific paper.

Our key contributions are as follows: First we create a parallel corpus of 5204 articles for the combined task of text summarisation and simplification from Eureka Alert (Section 3). Secondly, we propose a composite loss function for the task of summarisation and simplification (Section 3.2). For this task, we have extended the pointer generator (See, Liu, & Manning, 2017) architecture to perform simplification at the same time as performing summarisation. We have tailored the loss function of the pointer generator model by incorporating the easiness score. For this we used a lookup table of words and their corresponding binary score as hard or easy. Third, we propose a novel HTSS architecture for the joint task of summarisation and simplification (Section 3.2). Fourth, we propose a new evaluation measure for the combined tasks of summarisation and simplification (Section 3.3). We evaluate this against a corpus collected from Eureka Alert and demonstrate that our model outperforms two reasonable baselines (one specialised in simplification, the other in summarisation). Finally, we provide all code, models and corpora to the NLP scientific community for future research.[2]

The rest of the paper is structured as follows. Section 2 presents the literature review. Section 3 presents data and the proposed model, followed by the discussion on the results in Section 4. Finally, Section 5 presents concluding remarks and future directions.

2. Related work

Automatic document summarisation is the process of automatically compressing a document's contents whilst preserving the meaning. There are two main and widely studied approaches for automatic document summarisation, extractive and abstractive summarisation.

2.1. Extractive summarisation

In extractive document summarisation, a summary is obtained by picking important sections from the source document. Work on extractive summarisation stretches back to 1958 (Baxendale, 1958; Luhn, 1958), where automatic methods for generating abstracts of technical documents were first proposed. It was demonstrated that the importance of a word can be determined by its frequency. Similarly, a sentence is more informative, if it contains more important words (Luhn, 1958). Similarly, Baxendale (1958) investigated the position of sentences in a source document and found that, 85% of paragraphs contain an important sentence at the beginning and 7% of paragraphs contain an important sentence at the end.

Later, Edmundson (1969) examined 400 documents and created their summaries manually. He used 2 features from previous work, word frequency (Luhn, 1958) and positional importance (Baxendale, 1958). Edmundson proposed two new features: 1) cue words, and 2) structural features. Their automated approach generated an accurate summary 44% of time.

More recently extractive summarisation has been used in a supervised manner (Collins et al., 2017), especially at the sentence level. One method of extracting sentences is sentence regression (Zopf, Loza Mencía, & Fürnkranz, 2018), which predicts supervised utility scores at the sentence level. Extractive summarisation has also been performed using deep learning (Kinugawa & Tsuruoka, 2017). A recent comparison of algorithms for extractive summarisation (Mackie, McCreadie, Macdonald, & Ounis, 2014) demonstrates that the sumBasic algorithm (Nenkova & Vanderwende, 2005) was highly effective for summarisation of microblogs. Extractive summarisation may benefit from the identification of structured elements of texts (Filatova & Hatzivassiloglou, 2004), especially in domain specific language such as medical (Shardlow et al., 2018) or news text (Thompson, Nawaz, McNaught, &

[1] https://www.eurekalert.org/ .
[2] https://github.com/slab-itu/HTSS/ .

Ananiadou, 2017). Van Lierde and Chow (2019) argue that extractive summarisation often suffers from producing redundant summaries, thus, use of a graph based model and querying the text can be leveraged to address the problem of redundancy in the output summary. Single document summarisation can be extended to multi-document summarisation by incorporating sentiment lexicon and querying the text, which leads to an improvement in performance of semantic meaning of sentiment (Abdi, Shamsuddin, & Aliguliyev, 2018). Recently the effect of Semantic Role Labeling (SRL) using graph on text summarisation has been studied (Mohamed & Oussalah, 2019), demonstrating that semantic representation of input text improves the performance significantly. In a recent study, a web document and user-comments, tweets on social media about that document was used to enrich the output summary of the given document with new information from the tweets and posts. Furthermore, the authors showed that their method produces relevant summaries. They also noted that irrelevant comments about a document can degrade the performance of the proposed method (Nguyen, Tran, Nguyen, & Nguyen, 2019). More recently, a graph-based approach for text summarisation has been employed (Hark & Karcı, 2020). In this study, the authors form a graph from the input document by counting common words in between sentences, the sentences represent nodes and the links between them represent the edges weighted by the count of common words between two sentences. They removed nodes one-by-one from the graph and measured the graph entropy along with a new entropy proposed by the authors called Karci -entropy. In this way they are left with highly informative sentence in the graph which results in the output summary. According to the authors Karci -entropy method produces informative summary as compared to other classic graph based methods. Additionally, extractive text summarisation can be used applicationally, such as to determine the most informative sentences from hotel reviews (Hu, Chen, & Chou, 2017).

2.2. *Abstractive summarisation*

Abstractive summarisation leverages deep neural sequence to sequence networks to produce novel summarisations based on an input text (Chen & Bansal, 2018; Liu, Flanigan, Thomson, Sadeh, & Smith, 2015). These methods have the potential to generate long fluent passages of text, but suffer from non-factual generation in the output sentences. The use of Generative Adversarial Networks (Goodfellow et al., 2014) has been proposed to alleviate this problem (Liu et al., 2018). More advancement in abstractive text summarisation incorporates the granularity level of the output summary provided and controlled by the user (Azmi & Altmami, 2018). Rather than using one summarisation model, the aggregation of many summarisation models is also possible (Mehta & Majumder, 2018). In recent work, abstractive text summarisation was extended to multi-document summarisation (Barros, Lloret, Saquete, & Navarro-Colorado, 2019), in this work the author used the narrative approach to summarise news documents based on events from different news documents, the entities were arranged on a timeline in an ordered manner, then one sentence for each event is produced based on the information obtained from the documents. According to the authors it was found that the proposed approach produces a better summary as compared to state of the art abstractive approaches. Moreover, abstractive text summarisation had been combined with the extractive method by introducing diversification and content selection from the source document. In this work, the authors used three decoders to produce three different output summaries for a single input document. They scored each section of the source text, called ‘focus’. The authors reported better results as compared to previous works (Cho, Seo, & Hajishirzi, 2019). In Liu and Lapata (2019), Yang and Mirella used pre-trained encoders from BERT (kentonbert & Toutanova, 0000) for abstractive and extractive text summarisation, the authors showed that a pre-trained encoder with fine tuning of the decoder can enhance the capability of the text summarisation framework. Recently the effect of self pertaining transformers with supervised objectives for text summarisation was studied (Dong et al., 2019a). Later, the pre-training objectives were tailored for text summarisation (Zhang, Zhao, Saleh, & Liu, 2019), the authors also used masking in the input source text and mark it as a target text as in extractive models. The recent ProphetNet (Yan et al., 2020) also used self supervised manner of masking input text for the task of text summarisation, in this work the authors also used bi-grams future prediction in the decoder part of their model.

The ROUGE metric (Lin & Hovy, 2003) has been used widely for the evaluation of extractive and abstractive summarisation, however there are some limits such as strict word to word comparison of reference and generated sequences favours extractive summarisation, therefor the it must be applied with careful consideration of the meaning of the results (Schluter, 2017).

2.3. *Pointer-generator network*

The Pointer generator network is the combination of abstractive and extractive models (Gehrmann, Deng, & Rush, 2018). It uses encoder-decoder sequence to sequence model with attention, at the output. It uses two distributions: one is the distribution of the input source vocabulary and the other is the vocabulary of the decoder output. These two distributions are combined to make the final distribution. The output is then sampled from the final distribution. The key idea is to either keep the generated word by decoder in the final summary or copy words from input based on comparing their probability score.

The pointer generator network introduced by Vinyals, Fortunato, and Jaitly (2015) has been adopted for numerous tasks such as Neural Machine Translation (Gulcehre, Ahn, Nallapati, Zhou, & Bengio, 2016), summarisation (Gehrmann et al., 2018; Nallapati, Zhou, dos Santos, Gulçehre, & Xiang, 2016), and language modeling.

In Vinyals et al. (2015), the authors proposed pointer generator network and applied the pointer network to three non trivial challenging problems: calculating planner convex hulls, traveling salesman problem and delaunay triangulation’s. The main finding of this study is content-based input attention. This attention is different from the attention proposed in Bahdanau, Cho, and Bengio (2015).

The authors in See et al. (2017) applied the pointer generator networks to the task of text summarisation. In this work they used pointers to copy words from input so that the accuracy of the summary is retained, while generating new words from generative model to make the summary abstractive in nature. Another contribution of this work is, they used a coverage mechanism to avoid the repetition in the output summary.

2.4. Automated text simplification

Text simplification is an open problem in NLP. Early approaches started out by addressing issues of technical manual writing (Hoard, Wojcik, & Holzhauser, 1992) and assisting stroke survivors to read (Carroll, Minnen, Canning, Devlin, & Tait, 1998). There are three typical methods of text simplification found in the literature: syntactic simplification, lexical simplification and neural text simplification (NTS).

Syntactic simplification focuses on rewriting grammatical structures to transform constructs such as the passive voice, or long lists into more understandable structures (Siddharthan, 2014). This contrasts to lexical simplification, where a pipeline approach is typically used (Shardlow, 2014b). This pipeline consists of complex word identification, substitution generation, word sense disambiguation and reranking to select a replacement synonym. The pipeline has been shown to have a significant error rate (Shardlow, 2014a).

At the intersection of lexical and syntactic simplification, open source machine translation software can be used to generate both lexical and syntactic simplifications concurrently. Early attempts used Phrase-based statistical machine translation software (Li, Li, Qiang, & Yuan, 2018; Wubben, van den Bosch, & Krahmer, 2012), whereas more recent efforts have used neural machine translation (Nisioi, Štajner, Ponzetto, & Dinu, 2017b), for which the difficulty of the output can be controlled (Agrawal & Carpuat, 2019; Marchisio, Guo, Lai, & Koehn, 2019; Nishihara, Kajiwara, & Arase, 2019).

Simplification has previously been incorporated as part of the summarisation pipeline, where it is necessary to generate texts for children (Macdonald & Siddharthan, 2016), or for clinical summaries (Acharya et al., 2019). Our methods vary from these as we are using the pointer generator model with an improved loss function to model simplification.

Further recent developments in the field of text simplification include work to develop the scoring methodology for text simplification systems, leading to a new standardising package called EASSE (Alva-Manchego et al., 2019). Further work in supervised simplification (Dong et al., 2019b) has seen the prediction of explicit operations for simplification, whereas work in unsupervised text simplification has focussed on building systems that learn simplifications based solely on the text content of corpora (Surya et al., 2019).

3. Data and methods

Text summarisation models require parallel corpora of source and target texts, where the source is a long text document and the target is its summary. Manual creation of parallel corpora for text summarisation is time consuming and requires domain experts to read and synthesise the source documents.

In our approach, we automatically harvested a parallel corpus of summaries from the Eureka Alert website and corresponding scientific journal articles. Eureka alert is a website where researchers, bloggers and other domain experts take scientific documents and manually create a summarised and easy to read version. The summary is publicly available through the website of Eureka Alert.

We first extracted 227,590 simplified easy to read versions of scholarly articles from Eureka Alert. Each summarised document is linked to the corresponding journal article through its DOI. The DOIs allowed us to get PDFs, however parsing these PDFs yielded significant text cohesion errors, so we filtered for only those articles for which we could get an XML version, yielding 5204 summary-article pairs from the following journals: PLOS-ONE, Nature Communication, and Scientific Reports. The data is characterized by the following six attributes: Eureka Title Simplified, Eureka Text Simplified, Paper Title, Full Paper XML, Paper Journal and Paper DOI.

3.1. Approaches

In this section, we present the pointer generator model and our proposed adaptations to make it suitable for the joint task of text summarisation and simplification, as shown in Fig. 1.

3.1.1. Pointer generator model

Our implementation for the pointer generator model follows that of See et al. (2017). That is, an encoder with a single layer bi-directional LSTM, a decoder with single layer uni-directional LSTM, and a soft switch $\boldsymbol{P_{sw}}$, which is the probability $\in [0, 1]$ of the newly generated word by the decoder from the distribution of the vocabulary.

The encoder produces a sequence of hidden states h_i by taking words w_i from the input source document. During training, at time step t the decoder produces summary word w_t by taking the hidden states sequence h_i produced by the encoder and the previous word w_{t-1} from the reference summary, while at test time that previous word w_{t-1} comes from the decoder output, produced in the previous time step. The attention distribution is derived following the procedure presented in Bahdanau et al. (2015), by the encoder state h_i, and the decoder state s_t. The formulation of the presented model is as follows:

$$e^t = v^T tanh(W_h h_i + W_s s_t + b_{atn}) \tag{1}$$

$$a^t = softmax(e^t) \tag{2}$$

Where b_{atn}, W_s, and W_h are learnable parameters. The decoder uses the attention distribution as a probability distribution of the

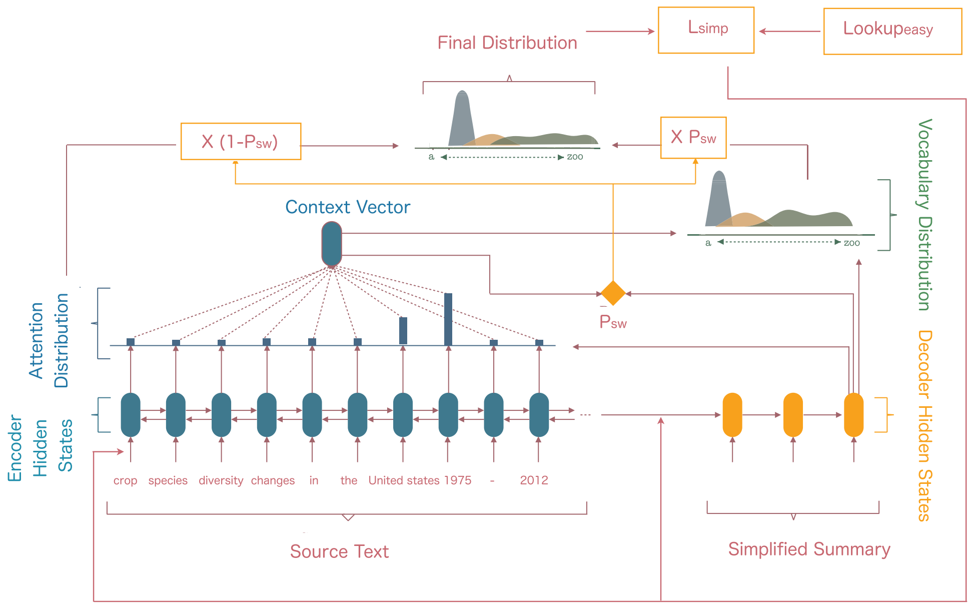

Fig. 1. Proposed HTSS Abstractive Simplified Architecture.

source words for producing the next word. Alternatively, the attention guides the decoder to look back to the source document probability distribution while generating new words. Next the context vector h_t^* is produced by taking the weighted sum of the encoder hidden states with the attention distribution:

$$h_t^* = \sum_i a_i^t h_i \tag{3}$$

The context vector also known as the taught vector is the representation of the input source for the current step. To compute the distribution of the vocabulary P_{vocab}, the decoder state s_t is concatenated with the taught vector h_t^*, and then the concatenated result is passed through two linear layers.

$$X = CONCAT(s_t, h_t^*) \tag{4}$$

$$P_{\mathrm{vocab}} = softmax(V'(VX + b) + b') \tag{5}$$

Where b, b', V, and V' are learnable parameters. And P_{vocab} is the probability distribution of all words present in the vocabulary.

$$P_{\mathrm{sw}} = \sigma\left(w_{h^*}^T h_t^* + w_s^T s_t + w_x^T x_t + b_{ptr}\right) \tag{6}$$

Where scalar b_{ptr}, vectors w_x, w_s, and w_{h^*} are learnable parameters. The σ is the sigmoid function, and P_{sw} is a switching probability. P_{sw}, is further used decide words' selection from source to the target while using the attention distribution a^t or generate a new word by sampling from P_{vocab}. For each input text document, the union of all words of the source document and the vocabulary is taken to form an extended vocabulary. The following probability distribution using the extended vocabulary is obtained:

$$P(w) = P_{\mathrm{sw}} P_{\mathrm{vocab}}(w) + (1 - P_{\mathrm{sw}}) \sum_{i:w_i=w} a_i^t \tag{7}$$

P_{vocab} is zero if w is out of vocabulary word, also $\sum_{i:w_i=w} a_i^t$ is zero if w is not present in the source text document.

At training time, for a time step t the loss can be computed by calculating the negative log likelihood of the target word w_t^*, which appears at t.

$$\mathcal{L}_{\mathrm{t}} = -log(P(\mathrm{w}_{\mathrm{t}}^*)) \tag{8}$$

and the total loss of the sequence can be computed as:

$$\mathcal{L} = \frac{1}{T} \sum_{t=0}^{T} \mathcal{L} \tag{9}$$

Sequence to sequence models suffers from the problem of generating repeated tokens in the output, and hence leads to non-sensical results. To overcome this problem, we built upon the implementation of See et al. (2017). In the coverage mechanism, we sum the attention distribution for all decoder previous time steps, and obtain a coverage vector c^t as follows:

$$c^t = \sum_{t'=0}^{t-1} a^{t'} \tag{10}$$

The coverage vector is the distribution of all words present in the source document representing words' weights received so far. Initially, the value of c^0 is set to zero and increases with the respect to the number of processed documents. To re-compute the attention using the coverage vector as an additional parameter, Eq. (1) becomes:

$$e_i^t = v^T tanh(w_h h_i + w_s s_t + w_c c_i^t + b_{atn}) \tag{11}$$

Where w_c is a learnable parameter having the same length as v. This keeps the attention mechanism updated with all previous decisions taken with the help of c_t. This make the attention mechanism able to avoid the repetition of attending the same locations. Using this strategy, the coverage loss can be computed as follows:

$$\mathcal{L} \quad = \sum_i min(a_i^t, c_i^t) \tag{12}$$

Incorporating the coverage loss weighted by the hyper-parameter λ, the loss function is updated as follows:

$$\mathcal{L} \ = -log(P(w_t^*)) + \lambda \sum_i min(a_i^t, c_i^t) \tag{13}$$

3.2. Proposed HTSS abstractive simplified architecture

We extended the pointer generator model with an improved loss function, which further tailored it for the combined task of text summarisation and simplification.

In order to enforce the model to learn the generation of the simplified summary. We calculated the easiness score for each summary and hence calculated $\mathcal{L}$ as the simplification loss:

$$\mathcal{L} \quad = \sum_t^T Lookup_{\text{easy}}(W_t^*) \tag{14}$$

Where $Lookup_{\text{easy}}$ is taken from a lookup table, which contains vocabulary words, each annotated for complexity. Each word has a binary label: zero if it is easy, one if it is complex. We tagged 49,000 words using groups of masters students proficient in English. Each word was tagged by three students and the tag was chosen based on the agreement of at least 2 annotators.

The updated loss can be computed by aggregating the likelihood loss, coverage loss and the simplification loss as follows:

$$\mathcal{L} \ = -log(P(w_t^*)) + \lambda \sum_i min(a_i^t, c_i^t) + \beta \sum_t Lookup_{\text{easy}}(W_t^*) \tag{15}$$

where β is a hyperparameter which can be tuned experimentally, by deciding, how much we pay attention to the task of simplification. The overall loss for the whole sequence can be computed as:

$$\mathcal{L} = \frac{1}{T} \sum_{t=0}^{T} \mathcal{L} \tag{16}$$

3.3. Evaluation

Finally, to quantify how much a model perform on each specific task, an evaluation measure is required. We used Rouge-1 (Lin & Hovy, 2003) for evaluating summarisation, while SARI is used to measure the simplification of our model. We used the implementaiton of SARI (Xu, Napoles, Pavlick, Chen, & Callison-Burch, 2016) from the EASSE package (Alva-Manchego et al., 2019). To unify the measures and evaluate the joint task of summarisation and simplification on a common platform, we used the harmonic mean of Rouge-1 and SARI to evaluate the combined task of summarisation and simplification. We call this measure: **C**ombined **S**ummarisation and **S**implification score (CSS), the formulation of the harmonic mean is as follows:

$$CSS_\gamma = \frac{(\gamma^2 + 1)R_1 \times SARI}{\gamma^2 R_1 + SARI} \tag{17}$$

Where R_1 is Rouge-1 score, and SARI is the simplification score. We set $\gamma = 1$ and the formula simplify as:

$$CSS_1 = \frac{2R_1 \times SARI}{R_1 + SARI} \tag{18}$$

The use of the γ parameter allows future studies to vary the importance that is paid to the task of summarisation (e.g., $\gamma > 1$) or simplification (e.g., $\gamma < 1$). However in our study we only evaluate the case where equal weight is given to both tasks.

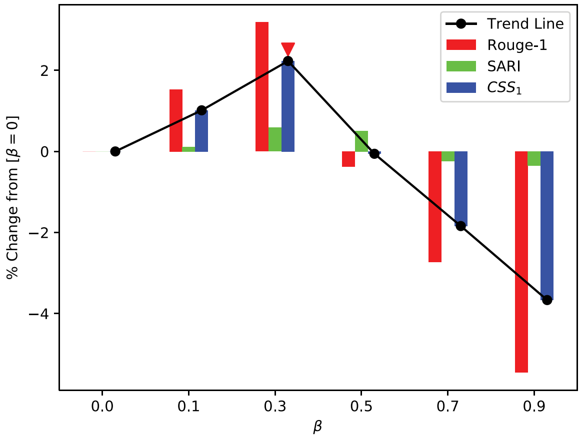

Fig. 2. Rouge-1, SARI and CSS_1 scores in relation to β.

3.4. Baseline systems

As a fair comparison, we selected one system for text simplification called Neural Text Simplification (NTS) (Nisioi et al., 2017b) and another for Abstractive Text Summarisation (ATS) (Nikolov et al., 2018). The NTS, is an encoder-decoder sequence-to-sequence model with beam search, the encoder consists of two LSTM layers and 500 hidden units, and the decoder consists of two layers of LSTM with global attention. ATS is an end to end data driven sequence-to-sequence model consisting of two layers of LSTM encoders with 1000 hidden units and 500 dimension of word embedding. We ran experiments using these two systems on our dataset and report the results on SARI, ROUGE-1 and CSS_1.

4. Results and discussion

We carried out our experiments, using a Linux based machine with two Titan 1080 GPUs. We used Python 3.7 as a scripting language. For the implementation of neural network based modules we used PyTorch (Paszke et al., 2017) – a deep learning framework. During our experiments, we used a learning rate of 0.001, hidden state of 256 units, and batch size of 164. The value of λ was set to 0.3, as in prior work (See et al., 2017). However, for the value of β, we ran a set of experiments to empirically find the optimal value according to CSS_1 score, see Fig. 2.

4.1. HTSS in relation to NTS and ATS

It is clear from Table 1 that our HTSS method vastly outperforms the baseline systems across all three measures. We were surprised to see in our results that although our system is trained and designed for the joint task of simplification and summarisation, it performs well across the Rouge and SARI metrics when compared to the baselines. In addition, our system greatly outperforms the baseline systems on CSS_1 score. It was surprising to see that the NTS system performed well on the ROUGE metrics, beating ATS (which is specialised for the summarisation task) on ROUGE-1 and ROUGE-L, but not on ROUGE-2. ROUGE-2 measures the overlap of bigrams between source and target text, and is significantly lower than the other 2 metrics in our evaluation, indicating that this was a harder task all round. In addition, it was surprising to see that the ATS system, although containing no innate simplification knowledge, performed well on the SARI baseline, getting a slightly lower, but equivalent performance to the NTS system.

Fig. 2 shows the difference in score obtained at different values of β, when compared to $\beta = 0$. We can see that some increase is observed across all 3 metrics for $\beta = 0.1$ and $\beta = 0.3$, but that this begins to decrease from $\beta = 0.5$ and onwards. According to this graph we empirically selected $\beta = 0.3$ as our threshold to report all our other results with. The main effect of increasing β past 0.3 is to inhibit the ROUGE-1 score. This is to be expected as β is increasing the amount of weight being given to the simplification loss, causing the model to focus more on the simplification task and less on summarisation.

In Fig. 3, we have shown the distribution of our data instances according to their SARI and ROUGE scores as a heatmap. It is clear in these figures that there is some significant variation in the score, with the majority of the scores centered around the means

Table 1
Evaluation of our proposed HTSS Abstractive Simplified model with benchmarks.

Models	Rouge-1	Rouge-2	Rouge-L	SARI	$\mathbf{CSS_1}$
HTSS = $\mathcal{PG} + \mathcal{L}$	21.938	03.21	17.171	37.650	27.7225
NTS	14.30	2.81	9.37	32.98	19.9498
ATS	12.51	3.02	7.35	32.53	18.070

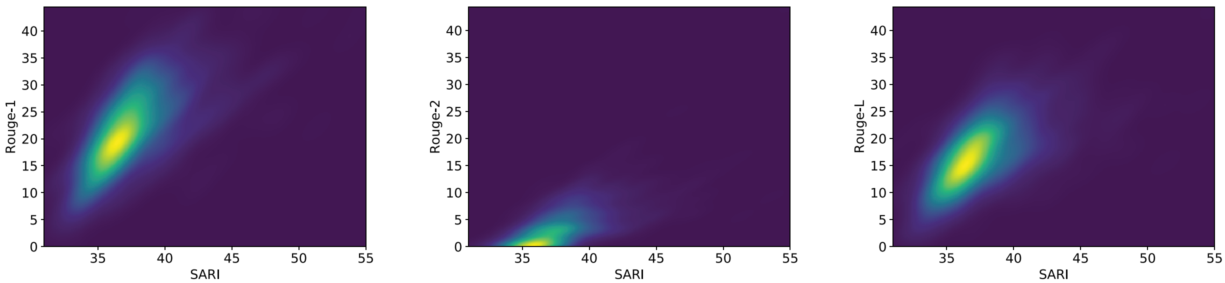

Fig. 3. SARI in relation to rouge-1, 2 and L scores $\beta = 0.3$.

reported in Table 1. We can see that SARI and ROUGE scores vary together indicating that a sentence that scores well according to SARI will also score well according to ROUGE

4.2. Discussion on the instances generated by HTSS

Furthermore, we have provided a few examples of our system's output. The top five examples according to CSS_1 score are presented in Table 2. These show that our system is capable of generating fluent and coherent text which successfully summarises and simplifies the input sentences. We have also presented the worst five examples according to CSS_1 score in Table 3. These examples show where our system has tried and failed to simplify a text. We can see in the first example in this table, that sometimes complex technical terminology may be a barrier to successful simplification and summarisation, as the system has placed many 'UNK' tokens in the output, indicating it did not have an appropriate word in its vocabulary to replace for the given input. The second example demonstrates text where the style may be unfamiliar to the system, containing some foreign words and several numbers and

Table 2
Top five instances generated by our model.

S.No.	Input Text	Output Text
1	fishing down nutrients on coral reefs 'fishing is widely considered a leading cause of biodiversity loss in marine environments but the potential effect on ecosystem processes such as nutrient fluxes is less explored. here we test how fishing on caribbean coral reefs influences biodiversity and ecosystem functions provided by the fish community that is fish-mediated nutrient capacity. specifically we modelled five processes of nutrient storage (in biomass) and supply (via excretion) of nutrients as well as a measure of their multifunctionality onto 143 species of coral reef fishes across 110 coral reef fish communities.	a new study led by researchers at the university of california san diego has found that coral larvae are able to restore the impact of coral reefs around the world. view more credit: university of california san diego.
2	cognitive improvement after mild traumatic brain injury measured with functional neuroimaging during the acute period functional neuroimaging studies in mild traumatic brain injury (mtbi) have been largely limited to patients with persistent post-concussive symptoms utilizing images obtained months to years after the actual head trauma. we sought to distinguish acute and delayed effects of mild traumatic brain injury on working memory functional brain activation patterns less than 72 hours after mild traumatic brain injury (mtbi) and again one-week later. we hypothesized that clinical and fmri measures of working memory would be abnormal in symptomatic mtbi patients assessed less than 72 hours.	new cognitive improvement after mild traumatic brain injury, this is a depiction of the traumatic brain cell. credit: university of illinois researchers at the university of california san diego school of medicine and the university of illinois have developed a new method to analyze traumatic brain injury.
3	effectiveness of electronic reminders to improve medication adherence in tuberculosis patients: a cluster-randomised trial mobile text messaging and medication monitors (medication monitor boxes) have the potential to improve adherence to tuberculosis (tb) treatment and reduce the need for directly observed treatment (dot) but to our knowledge they have not been properly evaluated in tb patients. we assessed the effectiveness of text messaging and medication monitors to improve medication adherence in tb patients. in a pragmatic cluster-randomised trial 36 districts/counties.	electronic reminders to prevent tuberculosis treatment, this is a study published in plos medicine at the university of hygiene academy of sciences nizhny novgorod academy of sciences
4	differential toxicity of antibodies to the prion protein antibodies against the prion protein prpc can antagonize prion replication and neuroinvasion and therefore hold promise as possible therapeutics against prion diseases. however the safety profile of such antibodies is controversial. it was originally reported that the monoclonal antibody d13 exhibits strong	a new insights into how the prion protein [UNK] this is a microfilaria of wuchereria bancrofti from the patient. credit: anant madabhushhi a deep-learning prion protein ubiquitous.
5	low cloud cover-adjusted ultraviolet b irradiance is associated with high incidence rates of leukemia: study of 172 countries there are 52 380 cases of leukemia and 24 090 deaths from it in the us annually. its causes are unknown and no preventive strategies have been implemented. we hypothesized that leukemia is due to.	leukemia are benefit to reduce ultraviolet b corvallis ore. Researchers at the university of california san diego school of medicine have developed a new evidence

Table 3
Five worst instances generated by our model.

S. no.	Input Text	Output Text
1	spacer-free bodipy fluorogens in antimicrobial peptides for direct imaging of fungal infection in human tissue fluorescent antimicrobial peptides are promising structures for in situ real-time imaging of fungal infection. here we report a fluorogenic probe to image aspergillus fumigatus directly in human pulmonary tissue. we have developed a fluorogenic trp-bodipy amino acid with a spacer-free c-c linkage between trp and a bodipy fluorogen which shows remarkable fluo- rescence enhancement in hydrophobic microenvironments. the incorporation of our fluorogenic amino acid in short antimicrobial peptides does not impair their selectivity for fungal cells and enables rapid and direct fungal imaging without any washing.	antimicrobial peptides does peptides does not sensitive [UNK] for [UNK] [UNK] this is a schematic illustration of the fungal cell cell. view credit: [UNK] [UNK] [UNK] [UNK] [UNK] peptides does not respond to a raw maze maze [UNK] [UNK] and colleagues at the university of [UNK] have developed.
2	advancing medical professionalism in us military detainee treatment citation: rubenstein ls allen sa guze pa (2016) advancing medical professionalism in us military detainee treatment. plos med 13(1): published: january 5, 2016 copyright: 2016 rubenstein et al. this is an open access article distributed under the terms of the creative commons attribution license which permits unrestricted use distribution and reproduction in any medium provided the original author and source are credited funding: no funding was received for this work.	medical competing us competing medical us [UNK] this is a medical article published in plos medicine at the hebrew university of jerusalem have developed a way to assess the fate of medical chart in plant-based foods.
3	exceptional preservation of eye structure in arthropod visual predators from the middle jurassic vision has revolutionized the way animals explore their environment and interact with each other and rapidly became a major driving force in animal evolution. however direct evidence of how ancient animals could perceive their environment is extremely difficult to obtain because internal eye structures are almost never fossilized. here we reconstruct with unprecedented resolution the three-dimensional structure of the huge compound eye of a 160-million-year-old thylacocephalan arthropod from the la voulte exceptional fossil biota in se france. this arthropod had about 18 000 lenses on each eye which.	[UNK] jurassic vision vision uncovered in outh asia [UNK] this is a reconstruction of the middle jurassic vision vision view [UNK] [UNK] visual jurassic jurassic vision vision vision in the middle jurassic vision has been likely according to a study published december 19, [UNK] in the
4	long-term urban carbon dioxide observations reveal spatial and temporal dynamics related to urban characteristics and growth recent efforts to reduce greenhouse gas emissions have focused on cities due to intensive emissions viable policy levers and interested stakeholders. atmospheric observations can be used to independently evaluate emissions but suitable networks are sparse. we present a unique decadal record of atmospheric co2 from five sites with contrasting urban characteristics that show divergent trends in co2 emissions across a city. comparison with population growth reveals a nonlinear relationship that may reflect how urban form affects co2 emissions. four state-of-the-art global-scale emission inventories capture.	[UNK] co2 co2 shaped spatial carbon dioxide into atmosphere [UNK] this is an artists depiction of atmospheric urban carbon dioxide [UNK] [UNK] [UNK] [UNK] [UNK] atmospheric carbon dioxide [UNK] atmospheric carbon dioxide to [UNK] atmospheric carbon dioxide into the atmosphere according to a study published june.
5	changes in intake of fruits and vegetables and weight change in united states men and women followed for up to 24 years: analysis from three prospective cohort studies current dietary guidelines recommend eating a variety of fruits and vegetables. however based on nutrient composition some particular fruits and vegetables may be more or less beneficial for maintaining or achieving a healthy weight. we hypothesized that greater consumption of fruits and vegetables with a higher fiber content or lower glycemic load would be more strongly associated with a healthy weight. we examined the association between change in intake of specific fruits.	[UNK] changes in fruits and vegetables men [UNK] women are pictured. [UNK] [UNK] women are vital in dramatic dramatic dramatic dramatic of fruits and fruits and fruits and [UNK] according to a study published december 14, [UNK] in the open-access journal plos one by [UNK]

punctuation symbols. This causes the system to give some repeated output at the beginning and then simply repeat a non-relevant portion of the original text. The third and fourth examples present similar mixtures of unfamiliar words and styles, which have also caused many 'UNK' tokens and elements of repeated text. This is to be expected in any neural language generation system and we expect that future systems will benefit from developments in the field of deep learning.

4.3. Theoretical and practical implications

In this section, we present a number of theoretical and practical implications of this study. The first and foremost crucial theoretical implication is the proposal of a hybrid summarisation and simplification architecture. We present a novel loss function that aggregates the likelihood loss, coverage loss, and simplification loss for the combined task of summarisation and simplification. Furthermore, we empirically fine-tune the values of the β and λ hyper-parameters to get the best results. The second theoretical implication of our study is the introduction of a new metric called (CSS_1), which combines SARI and Rouge and demonstrates the effectiveness of the model in relation to the benchmarked state-of-the-art models.

This research combines two critical and recently desired natural language processing tasks (summarisation and simplification). To the best of our knowledge, these tasks are often discussed in separate streams of literature. The abstractive text summarisation task, adopted in this research, seeks to reduce the content of the text by preserving the original underlying message. Similarly, the text simplification task adopted in this research helps second language learners with limited vocabulary and knowledge of in-depth domain-specific terminologies by providing them a simplified conversion of the text.

Both the tasks of summarisation and simplification are desired in ever-increasing textual information over the web and many other sources, like news articles, scientific literature, and blogs. The proposed models in this study can provide a human-readable synopsis of large-textual content. Our research can also be leveraged for search engines to summarise and simplify retrieved documents and present the summaries to the user so that the user can have a quick idea about the document in a short time. Another important implication of our research lies in the development of the technical report or user-manuals suited for non-technical readers and stakeholders.

Last but not least, our research fulfills the appetite of scholars, students, and professionals by providing them engaging yet simplified text to enable them to study cross domain-specific content in order to produce innovative and productive multi-disciplinary solutions. To study domain-specific work, a non-domain expert must employ a domain-specific expert to breakdown the technical terminology into a simplified summary. While some primitive tools exist to provide manually converted understandable summary documents for scientific/technical articles, or the translation of technical terminology, to this date, there is no such automated tool that can conveniently convert any provided scientific/technical literature into an easily understandable and readable summary text. Tapping on this proposed HTSS framework, we seek to build automated tools for the broader readability and outreach of scientific knowledge.

5. Concluding remarks

We present the first attempt at using the pointer generator model for the combined task of simplification and summarisation by incorporating a novel measure of simplification into the loss function. We have provided a new corpus of simplified and summarised articles, on which we have derived a new evaluation measure, CSS_1. We expect that our resources will be of great value to the simplification and summarisation communities. In future work, we would seek to extend our model by tuning for both the lambda and beta parameters jointly. This would allow us to better understand the optimal parameters for out model. In addition, we will seek to improve the loss function by incorporating further features that indicate the complexity of a word such as length, frequency, concreteness and polysemy. We will also consider how to expand the vocabulary of our model to help it to avoid putting 'UNK' tokens into the output.

CRediT authorship contribution statement

Farooq Zaman: Data curation, Investigation, Writing - original draft, Writing - review & editing. **Matthew Shardlow:** Writing - original draft, Writing - review & editing. **Saeed-Ul Hassan:** Conceptualization, Methodology, Investigation, Writing - original draft, Writing - review & editing. **Naif Radi Aljohani:** Writing - original draft, Writing - review & editing. **Raheel Nawaz:** Methodology, Writing - original draft.

References

Abdi, A., Shamsuddin, S. M., & Aliguliyev, R. M. (2018). Qmos: Query-based multi-documents opinion-oriented summarization. *Information Processing & Management, 54*(2), 318–338.

Acharya, S., Boyd, A. D., Cameron, R., Lopez, K. D., Martyn-Nemeth, P., Dickens, C., ... Di Eugenio, B. (2019). *Incorporating personalization features in a hospital-stay summary generation system. Proceedings of the 52nd Hawaii international conference on system sciences*.

Agrawal, S., & Carpuat, M. (2019). *Controlling text complexity in neural machine translation. Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics1549–1564. https://doi.org/10.18653/v1/D19-1166.

Alva-Manchego, F., Martin, L., Scarton, C., & Specia, L. (2019). *EASSE: Easier automatic sentence simplification evaluation. Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (emnlp-ijcnlp): System demonstrations*. Hong Kong, China: Association for Computational Linguistics49–54. https://doi.org/10.18653/v1/D19-3009 https://www.aclweb.org/anthology/D19-3009415.

Azmi, A. M., & Altmami, N. I. (2018). An abstractive arabic text summarizer with user controlled granularity. *Information Processing & Management, 54*(6), 903–921.

Bahdanau, D., Cho, K., & Bengio, Y. (2015). *Neural machine translation by jointly learning to align and translate. 3rd international conference on learning representations, ICLR 2015*.

Barros, C., Lloret, E., Saquete, E., & Navarro-Colorado, B. (2019). Natsum: Narrative abstractive summarization through cross-document timeline generation. *Information Processing & Management, 56*(5), 1775–1793.

Baxendale, P. B. (1958). Machine-made index for technical literature an experiment. *IBM Journal of Research and Development, 2*(4), 354–361.

Carroll, J., Minnen, G., Canning, Y., Devlin, S., & Tait, J. (1998). *Practical simplification of english newspaper text to assist aphasic readers. In proc. of AAAI-98 workshop on integrating artificial intelligence and assistive technology*7–10.

Chen, Y.-C., & Bansal, M. (2018). *Fast abstractive summarization with reinforce-selected sentence rewriting. Proceedings of the 56th annual meeting of the association for computational linguistics (volume 1: Long papers)*. Melbourne, Australia: Association for Computational Linguistics675–686. https://doi.org/10.18653/v1/P18-1063 https://www.aclweb.org/anthology/P18-1063.

Cho, J., Seo, M., & Hajishirzi, H. (2019). *Mixture content selection for diverse sequence generation. Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*3112–3122.

Collins, E., Augenstein, I., & Riedel, S. (2017). *A supervised approach to extractive summarisation of scientific papers. Proceedings of the 21st conference on computational natural language learning (CONLL 2017)*195–205.

Dong, L., Yang, N., Wang, W., Wei, F., Liu, X., Wang, Y., ... Hon, H.-W. (Yang, Wang, Wei, Liu, Wang, Gao, Zhou, Hon, 2019a). *Unified language model pre-training for natural language understanding and generation. Advances in neural information processing systems*13042–13054.

Dong, Y., Li, Z., Rezagholizadeh, M., & Cheung, J. C. K. (Li, Rezagholizadeh, Cheung, 2019b). *EditNTS: An neural programmer-interpreter model for sentence simplification through explicit editing. Proceedings of the 57th annual meeting of the association for computational linguistics*. Florence, Italy: Association for Computational

Linguistics3393–3402. https://doi.org/10.18653/v1/P19-1331 https://www.aclweb.org/anthology/P19-1331.
Edmundson, H. P. (1969). New methods in automatic extracting. *Journal of the ACM (JACM), 16*(2), 264–285.
Filatova, E., & Hatzivassiloglou, V. (2004). *Event-based extractive summarization. Text summarization branches out*. Barcelona, Spain: Association for Computational Linguistics104–111 https://www.aclweb.org/anthology/W04-1017.
Gehrmann, S., Deng, Y., & Rush, A. (2018). *Bottom-up abstractive summarization. Proceedings of the 2018 conference on empirical methods in natural language processing*. Brussels, Belgium: Association for Computational Linguistics4098–4109. https://doi.org/10.18653/v1/D18-1443 https://www.aclweb.org/anthology/D18-1443.
Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... Bengio, Y. (2014). *Generative adversarial nets. Advances in neural information processing systems*2672–2680.
Gulcehre, C., Ahn, S., Nallapati, R., Zhou, B., & Bengio, Y. (2016). *Pointing the unknown words. Proceedings of the 54th annual meeting of the association for computational linguistics (volume 1: Long papers)*. Berlin, Germany: Association for Computational Linguistics140–149. https://doi.org/10.18653/v1/P16-1014 https://www.aclweb.org/anthology/P16-1014560.
Hark, C., & Karcı, A. (2020). Karcı summarization: A simple and effective approach for automatic text summarization using karcı entropy. *Information Processing & Management, 57*(3), 102187.
Hoard, J. E., Wojcik, R., & Holzhauser, K. (1992). *An automated grammar and style checker for writers of simplified english*. Dordrecht: Springer Netherlands278–296).
Hu, Y.-H., Chen, Y.-L., & Chou, H.-L. (2017). Opinion mining from online hotel reviews–a text summarization approach. *Information Processing & Management, 53*(2), 436–449.
Kenton, J. D. M.-W. C., & Toutanova, L. K.. Bert: Pre-training of deep bidirectional transformers for language understanding,.
Kinugawa, K., & Tsuruoka, Y. (2017). *A hierarchical neural extractive summarizer for academic papers. JSAI international symposium on artificial intelligence*. Springer339–354.
Kriz, R., Sedoc, J., Apidianaki, M., Zheng, C., Kumar, G., Miltsakaki, E., & Callison-Burch, C. (2019). *Complexity-weighted loss and diverse reranking for sentence simplification. 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)*. Minneapolis, Minnesota: Association for Computational Linguistics3137–3147. https://doi.org/10.18653/v1/N19-1317 https://www.aclweb.org/anthology/N19-1317.
Li, T., Li, Y., Qiang, J., & Yuan, Y.-H. (2018). Text simplification with self-attention-based pointer-generator networks. In L. Cheng, A. C. S. Leung, & S. Ozawa (Eds.). *Neural information processing* (pp. 537–545). Cham: Springer International Publishing.
Lin, C.-Y., & Hovy, E. (2003). *Automatic evaluation of summaries using n-gram co-occurrence statistics. Proceedings of the 2003 human language technology conference of the north American chapter of the association for computational linguistics*150–157 https://www.aclweb.org/anthology/N03-1020.
Liu, F., Flanigan, J., Thomson, S., Sadeh, N., & Smith, N. A. (2015). *Toward abstractive summarization using semantic representations. Proceedings of the 2015 conference of the north American chapter of the association for computational linguistics: Human language technologies*. Denver, Colorado: Association for Computational Linguistics1077–1086. https://doi.org/10.3115/v1/N15-1114 https://www.aclweb.org/anthology/N15-1114.
Liu, L., Lu, Y., Yang, M., Qu, Q., Zhu, J., & Li, H. (2018). *Generative adversarial network for abstractive text summarization. Thirty-second AAAI conference on artificial intelligence*.
Liu, Y., & Lapata, M. (2019). *Text summarization with pretrained encoders. Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*3721–3731.
Luhn, H. P. (1958). The automatic creation of literature abstracts. *IBM Journal of Research and Development, 2*(2), 159–165.
Macdonald, I., & Siddharthan, A. (2016). *Summarising news stories for children. Proceedings of the 9th international natural language generation conference*. Edinburgh, UK: Association for Computational Linguistics1–10. https://doi.org/10.18653/v1/W16-6601 https://www.aclweb.org/anthology/W16-6601.
Mackie, S., McCreadie, R., Macdonald, C., & Ounis, I. (2014). Comparing algorithms for microblog summarisation. In E. Kanoulas, M. Lupu, P. Clough, M. Sanderson, M. Hall, A. Hanbury, & E. Toms (Eds.). *Information access evaluation. multilinguality, multimodality, and interaction* (pp. 153–159). Cham: Springer International Publishing.
Marchisio, K., Guo, J., Lai, C.-I., & Koehn, P. (2019). *Controlling the reading level of machine translation output. Proceedings of machine translation summit XVII volume 1: Research track*. Dublin, Ireland: European Association for Machine Translation193–203 https://www.aclweb.org/anthology/W19-6619610.
Mehta, P., & Majumder, P. (2018). Effective aggregation of various summarization techniques. *Information Processing & Management, 54*(2), 145–158.
Mohamed, M., & Oussalah, M. (2019). Srl-esa-textsum: A text summarization approach based on semantic role labeling and explicit semantic analysis. *Information Processing & Management, 56*(4), 1356–1372.
Nallapati, R., Zhou, B., dos Santos, C., Gulçehre, Ç., & Xiang, B. (2016). *Abstractive text summarization using sequence-to-sequence RNNs and beyond. Proceedings of the 20th SIGNLL conference on computational natural language learning*. Berlin, Germany: Association for Computational Linguistics280–290. https://doi.org/10.18653/v1/K16-1028 https://www.aclweb.org/anthology/K16-1028.
Nenkova, A., & Vanderwende, L. (2005). *The impact of frequency on summarization. Microsoft research, redmond, Washington, Tech. Rep. MSR-TR-2005101*.
Nguyen, M.-T., Tran, V. C., Nguyen, X. H., & Nguyen, L.-M. (2019). Web document summarization by exploiting social context with matrix co-factorization. *Information Processing & Management, 56*(3), 495–515.
Nikolov, N. I., Pfeiffer, M., & Hahnloser, R. H. (2018). *Data-driven summarization of scientific articles. Proc. of the 7th international workshop on mining scientific publications, lrec 2018*.
Nishihara, D., Kajiwara, T., & Arase, Y. (2019). *Controllable text simplification with lexical constraint loss. Proceedings of the 57th annual meeting of the association for computational linguistics: Student research workshop*. Florence, Italy: Association for Computational Linguistics260–266. https://doi.org/10.18653/v1/P19-2036 https://www.aclweb.org/anthology/P19-2036605.
Nisioi, S., Štajner, S., Ponzetto, S. P., & Dinu, L. P. (Štajner, Ponzetto, Dinu, 2017a). *Exploring neural text simplification models. Proceedings of the 55th annual meeting of the association for computational linguistics (volume 2: Short papers)*. Vancouver, Canada: Association for Computational Linguistics85–91. https://doi.org/10.18653/v1/P17-2014 https://www.aclweb.org/anthology/P17-2014.
Nisioi, S., Štajner, S., Ponzetto, S. P., & Dinu, L. P. (Štajner, Ponzetto, Dinu, 2017b). *Exploring neural text simplification models. Proceedings of the 55th annual meeting of the association for computational linguistics (volume 2: Short papers)*85–91.
Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., ... Lerer, A. (2017). *Automatic differentiation in pytorch. NIPS 2017 Autodiff Workshop*.
Schluter, N. (2017). *The limits of automatic summarisation according to ROUGE. Proceedings of the 15th conference of the European chapter of the association for computational linguistics: Volume 2, short papers*. Valencia, Spain: Association for Computational Linguistics41–45 https://www.aclweb.org/anthology/E17-2007.
See, A., Liu, P. J., & Manning, C. D. (2017). *Get to the point: Summarization with pointer-generator networks. Proceedings of the 55th annual meeting of the association for computational linguistics (volume 1: Long papers)*. Vancouver, Canada: Association for Computational Linguistics1073–1083. https://doi.org/10.18653/v1/P17-1099 https://www.aclweb.org/anthology/P17-1099440.
Shardlow, M. (2014a). *Out in the open: Finding and categorising errors in the lexical simplification pipeline. Proceedings of the ninth international conference on language resources and evaluation (LREC-2014)*. Reykjavik, Iceland: European Languages Resources Association (ELRA)1583–1590 http://www.lrec-conf.org/proceedings/lrec2014/pdf/479_Paper.pdf.
Shardlow, M. (2014b). A survey of automated text simplification. *International Journal of Advanced Computer Science and Applications, 4*(1), 58–70.
Shardlow, M., Batista-Navarro, R., Thompson, P., Nawaz, R., McNaught, J., & Ananiadou, S. (2018). Identification of research hypotheses and new knowledge from scientific literature. *BMC Medical Informatics and Decision Making, 18*(1), 46.
Siddharthan, A. (2014). A survey of research on text simplification. *ITL-International Journal of Applied Linguistics, 165*(2), 259–298.
Surya, S., Mishra, A., Laha, A., Jain, P., & Sankaranarayanan, K. (2019). *Unsupervised neural text simplification. Proceedings of the 57th annual meeting of the association for computational linguistics*. Florence, Italy: Association for Computational Linguistics2058–2068. https://doi.org/10.18653/v1/P19-1198 https://www.aclweb.org/anthology/P19-1198.
Thompson, P., Nawaz, R., McNaught, J., & Ananiadou, S. (2017). Enriching news events with meta-knowledge information. *Language Resources and Evaluation, 51*(2), 409–438.

Van Lierde, H., & Chow, T. W. (2019). Query-oriented text summarization based on hypergraph transversals. *Information Processing & Management, 56*(4), 1317–1338.

Vinyals, O., Fortunato, M., & Jaitly, N. (2015). *Pointer networks. Advances in neural information processing systems*2692–2700.

Wubben, S., van den Bosch, A., & Krahmer, E. (2012). *Sentence simplification by monolingual machine translation. Proceedings of the 50th annual meeting of the association for computational linguistics (volume 1: Long papers)*. Jeju Island, Korea: Association for Computational Linguistics1015–1024 https://www.aclweb.org/anthology/P12-1107.

Xu, W., Napoles, C., Pavlick, E., Chen, Q., & Callison-Burch, C. (2016). Optimizing Statistical Machine Translation for Text Simplification. *Transactions of the Association for Computational Linguistics, 4*, 401–415. https://doi.org/10.1162/tacl_a_00107 https://www.mitpressjournals.org/doi/abs/10.1162/630tacl_a_00107.

Yan, Y., Qi, W., Gong, Y., Liu, D., Duan, N., Chen, J., Zhang, R., & Zhou, M. (2020). Prophetnet: Predicting future n-gram for sequence-to-sequence pre-training. arXiv:2001.04063,.

Zhang, J., Zhao, Y., Saleh, M., & Liu, P. J. (2019). Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. arXiv:1912.08777,.

Zopf, M., Loza Mencía, E., & Fürnkranz, J. (2018). *Which scores to predict in sentence regression for text summarization? Proceedings of the 2018 conference of the north American chapter of the association for computational linguistics: Human language technologies, volume 1 (long papers)*. New Orleans, Louisiana: Association for Computational Linguistics1782–1791. https://doi.org/10.18653/v1/N18-1161 https://www.aclweb.org/anthology/N18-1161.