*International Journal of Engineering & Technology*, 7 (3) (2018) 1366-1371

**International Journal of Engineering & Technology**

*Website: www.sciencepubco.com/index.php/IJET*
*doi: 10.14419/ijet.v7i3.13291*
***Research paper***

# Semantic based neural model approach for text simplification

**Hemanth Somasekar [1] *, Dr. Kavya Naveen [1]**

*[1] RNS Institute of Technology, India*
**Corresponding author E-mail: hemanth.shantha@gmail.com*

**Abstract**

The machine translation systems affect by various difficulties like long-distance dependency and long sentences having complex syntax. Text Summarization (TeSu) and Text Simplification (TeSi) are the important ways of simplifying the text for users who are having the poor reading capability, including non-native speakers, functionally illiterate and children. TeSu produce a brief summary of the main ideas of the text, while TeSi aims to reduce the linguistic complexity of the text and retain the original meaning. In many text generation tasks, sequence-to-sequence model depends upon approaches of TeSu and TeSi achieves more success, recently. Text data have low Semantic Relevance (SR), but the Simplified Text (SiTs) which generate from the Source Text (SoTs) are more similar. The goal of the paper is to work for TeSu and TeSi, for improving the SR between the original texts and the modified texts. The proposed method encouraging high semantic similarity between texts and summaries by implementing SR based Neural model (SRN). The encoder represents the SoT, whereas, the decoder produced the summary representation. During training, the representation provides maximum Similarity Score (SS) and the experi-ments conducted on the approach using two benchmark datasets. The experimental results showed that SNR approach provided better per-formance compared to the existing method in terms of metrics such as readability metrics, human-sentence level evaluation, and Post Editing (PE) evaluation.

***Keywords***: *Use About Five Key Words or Phrases in Alphabetical Order, Separated by Semicolon*

## 1. Introduction

The conversation is quite normal for humans in our daily life, people can talk naturally as human's brain already know how to catch the key information and reply properly. However, this work is difficult for the machine, due to the unnecessary and superfluous words appear frequently in spoken dialogue [1]. Therefore, the technique of summarization played a major part and a lot of research work require in computational linguistics for solving real-world problems as per native linguistic use [2]. With the rapid development of the social network, a large number of users express their views to some hot topics through the network. The texts given by the users are related to society, life, science and technology, entertainment and other fields [3]. The problems such as redundancy and information overload cause by a number of available documents that are difficult to use and find the information effectively and efficiently. The above problems tackled by the process called Document Summarization (DS) and these urgent practical problems overcome by raising new methods [4]. The process of producing the topic-or generic reports compressed a set of documents sharing the similar topics by reducing the document length. The summary can be single-document (SDS) or a Multi-Document (MDS) which depends on the number of documents to be summarized. The reduction of one document into a shorter version is done on SDS whereas, a set of documents is compressed in MDS, moreover, in a cluster of documents, the MDS is used for outlining the information.

The summarization techniques are most useful for retrieving important information from the documents with the help of clustering and also have a vast range of application in many fields such as retrieval and information management [5-6]. The TeSu uses the generic summarization algorithms developed for summarizing the text that is both diverse and concise [7]. The process of summarization performed by two kinds of content such as multi-media content (i.e. transcripts of video) and textual content like books, etc., which make a rise for providing demand for a high-quality summary [8]. The user's semantic information is contained by these text data which are having more valuable resources of information in the era of big data. A number of management are making use of the semantic information on the web for decision-making [9]. Ambiguous words can have multiple meanings, for example, the word "cut" has several meanings such as perforate, slashed or chopped. Getting the right meaning of an ambiguous word is easy for a human, but developing Natural Language Processing (NLP) system for a machine is complicated. However, this can be overcome by incorporating the knowledge that identifies the original word of the uncertain word or called Word Sense Disambiguation (WSD). A classifier is applied for WSD to produce two types of knowledge sources that distinguish senses of the words in a given collection of words. The corpus is the first type that is not labelled with word senses, whereas, the dictionary can be thesaurus and machine-readable which is the second type of knowledge source [10-11].

Today, the level of writing in their independent language relate to the synaptic structures from the levels of paragraphs, phrases, clauses, and sentences by the process of Semantic analysis. The invariant meanings convert from the elements of figurative speech and idiom, which are cultural in the analysis of semantic information [12-13]. Without knowledge sources, it is hard for either people or machines to recognize the correct sense. According to knowledge or information source, several WSD techniques are ranging as supervised or unsupervised techniques. This paper proposed an SRN to improve the SR between SoTs and generated SiTs for TeSu and TeSi. A component for evaluating similarity is introduced to calculate the relevance of SoTs and Generated Texts

(GT), and the SS is maximized to encourage high SR between SoTs and SiTs. The research work also introduced a self-gated encoder to better represent a long redundant text. Section 2 explains the related works, whereas the proposed framework described in Section 3. The experiments are conducted on databases and the evaluation results represented in Section 4, finally, the conclusion is made in Section 5.

## 2. Literature review

Some existing methods related to our work described in the below section. The methodology used in the papers discussed and also their drawbacks detailed below.

L. Wang et al. [14] developed a Sentiment Related Index (SRI) for measuring the lexical elements between the associations in a specific domain using the bridge as a domain-independent feature. They proposed a sentiment classification algorithm as SentiRelated in cross-domain depends upon SRI, for analysing the polarity for short texts. The algorithms were validated on two typical datasets and the experimental results showed that the proposed SentiRelated algorithm was effective for analysing the short text polarity. The method trained classification model according to the vast data on the offline part, on the other hand, short deadlines are presented in the online analytics.

A. Abdi, et al., [15] employed the combination of summarization and sentiment approaches by implementing the QMOS method. The paper transformed the query-based MDS from the lexicon-based method of opinion, which expressed in the reviews. If the sentence was not included in a sentiment lexicon, the sentiment score of a word was determined by the QMOS by using Semantic Sentiment approach. The redundancy was reduced by employing the query expansion approach and greedy algorithm and the lexical gaps were filled for similar contexts which were expressed using different wordings. While comparing the two sentences, the method was not able to distinguish between an active sentence and a passive sentence.

N. K. Nagwani, [16] implemented an approach which was used for large text summarization by employing MapReduce technology. For summarizing the large text collection, the technique was presented using topic modeling with Latent Dirichlet Allocation (LDA) and semantic similarity based clustering over MapReduce framework. The modular implementation of MDS was provided by the task of summarization which was performed in four stages. The MapReduce framework reduced time complexity and provided better scalability for summarizing a large number of text documents. The framework was unable to provide the support for multilingual TeSu in different languages.

S. Xiong et al., [17] proposed a joint sentiment-topic model Word-pair Sentiment-Topic Model (WSTM) for detecting the topics and sentiments by considering the text sparse problem. The two words had the same topic in a word-pair, moreover, all words had the same sentiment polarity in a sentence which was held in the generative process of WSTM. The Chinese product review datasets were used for experimental evaluation, the results showed that the WSTM accurately identified the document-level sentiment in addition to learn high-quality topics. The method needed another effective topic model for filtering common topics because the method consumes more time for filtering which was directly affected the overall performance.

S. Song et al., [18] developed constructed new sentences for exposing more grained fragments than semantic phrases, employed an LSTM-CNN based ATS framework (ATSDL). The ATSDL consisted of two main steps, namely the summarization of GT using deep learning and the extraction of phrases from source sentences. The experiments were conducted on datasets such as CNN and DailyMail. The results evaluated that the ATSDL provided better performance in terms of both syntactic and semantic structure. The requirements of the syntactic structure were difficult by the sequence of keywords to solve them that was the major drawback of the approach.

## 3. Proposed methodology

The goal is to improve the SR between SoTs and SiTs. So, the proposed model of this paper encourages high similarity between their representations. Figure 1 represents the block diagram of our proposed model. The model consists of three components namely, encoder, decoder and a Similarity Function (SF). The SoTs compresses into semantic vectors by the encoder, whereas the generation of summaries and the production of semantic vectors of these generated summaries done by using decoder. The SF. evaluates the relevance of the vector of semantic SoTs and summaries. The main objective is to increase the SS, so that the summaries have high SR to SoTs.

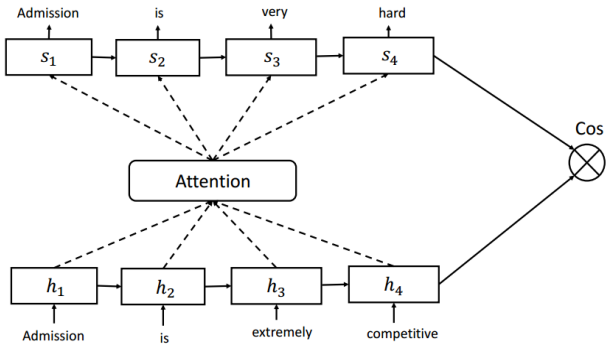


**Fig. 1:** The Block Diagram of the Proposed Methodology.

Where, $s_1, s_2, s_3, s_4$ represents as decoder, $h_1, h_2, h_3, h_4$ describe as encoder and *Cos* is defined as Cosine Similarity.

### 3.1. Self-gated encoder

The goal of the complex text encoder is to provide a series of dense representation of SoTs for the decoder and the SR component. In the previous work, the complex text encoder is a two-layer uni-directional Long Short-term Memory Network (LSTM), which produces the dense representation $\{h_1, h_2, ...., h_N\}$ from the SoT $\{x_1, x_2, ...., x_N\}$.

However, in TeSu and TeSi, SoTs are usually very long and noisy. Therefore, some encoding information at the beginning of the texts vanish until the end of the texts, which leads to bad representations of the texts. Bi-directional LSTM is an alternative to deal with the problem, but it needs double time to encoder the SoTs, and it does not represent the middle of the texts well when the texts are too long. To solve the problem, we proposed a self-gated encoder to better represent a long text.

In TeSu and TeSi, some words or information in the SoTs are unimportant, so they need to be simplified or discarded. Therefore, the SRN approach introduces a self-gated encoder, which can reduce the unnecessary information and enhance the important information to represent a long text. The basic structure of the Self-gated encoder represented in figure 2.

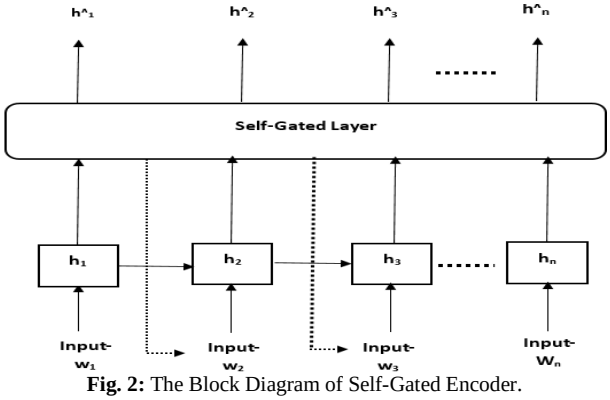


**Fig. 2:** The Block Diagram of Self-Gated Encoder.

Self-gated encoder tries to measure the importance of each word, and decide how much information is reserved as the representation of the texts. At each time step, every upcoming word $x_t$ is fed into the LSTM cell, which outputs the dense vector $h_t$.

$$h_t = f(x_t, h_{t-1}) \tag{1}$$

Where, $f$ is the LSTM function, and $h_t$ is the output vector of the LSTM cell. A Feed-Forward Neural Network (FFNN) used to measure the importance and decide how much information is reversed.

$$\beta_t = sigmoid(g(h_t)) \tag{2}$$

Where $g$ is the FFNN, and $\beta_t$ measures the proportion of the reserved information. Finally, the reversed information is computed by multiplying $\beta_t$.

$$\hat{h}_t = \beta_t . h_t \tag{3}$$

$$h_t = e_{t+1} \tag{4}$$

Where $\hat{h}_t$ is the representation at the $t_{th}$ time step, and $\hat{e}_{t+1}$ is the input embedding of $x_{t+1}$ at the $t+1_{th}$ time step.

### 3.2. Simplified text decoder

The goal of the SiT decoder is to generate a series of simplified words from the dense representation of SoTs. In SRN model, the dense representations of the SoTs are fed into an attention layer to generate the context vector $c_t$.

$$c_t = \sum_{i=1}^{N} \alpha_{t_i} \hat{h}_i \tag{5}$$

$$\alpha_{t_i} = \frac{e^{g(s_t, \hat{h}_i)}}{\sum_{j=1}^{N} e^{g(s_t, \hat{h}_j)}} \tag{6}$$

Where $s_t$ is the dense representation of generated simplified computed by a two-layer LSTM.

In this way, $c_t$ *and* $s_t$ respectively represent the information of SoTs and the target texts at the $t^{th}$ time step. To predict the $t^{th}$ word, the decoder uses $c_t$ *and* $s_t$ to generate the probability distribution of the candidate words:

$$p_t(y \vee x) = softmax(W_{s_t}) \tag{7}$$

$$\hat{s}_t = tanh(W_c[s_t; c_t]) \tag{8}$$

Where $W$ *and* $W_c$ is the parameter matrix of the output layer. Finally, the word with the highest probability is predicted:

$$y_t = argmax_y, p_t(y \text{ '} \vee \text{x}) \tag{9}$$

### 3.3. Semantic relevance

Our goal is to calculate the SR of SoTs and GTs given the source semantic vector $V_t$ and the generated semantic vector $V_s$. Here, we use Cosine Similarity (CS) to measure the SR, which is represented by a dot product and magnitude:

$$cos(V_s, V_t) = \frac{V_s . V_t}{\|V_s\| \|V_t\|} \tag{10}$$

SoTs and GTs share the same language, so it is reasonable to assume that their semantic vectors are distributed in the same space. The distance between two vectors is calculated by the CS in the same space.

With the SR metric, the problem is getting the semantic vector $V_s$ and $V_t$. The representation of text or sentences includes many methods namely reserving the last state of LSTM or mean the pooling of LSTM output. In our model, select the last state of the encoder as the representation of SoTs:

$$V_s = \hat{h}_N \tag{11}$$

The semantic vector of a summary is got by feeding them into the encoder as well, but the method loss more time because of the encoding process. The information for both SoT and generated summaries are contained by the last output of the decoder $\hat{S}_M$. We simply evaluate the semantic vector of the summary with the equation below:

$$V_s = \hat{S}_M - \hat{h}_N \tag{12}$$

The method is effective for representing a span of words, which is proved by previous work, without encoding the words once more.

### 3.4. Training

The model parameter is given as θ and x is an input text, the model produces corresponding summary y and semantic vector $V_s$ and $V_t$. The objective is to minimize the loss function.

$$L = -p(y \vee x; \theta) - \lambda cos(V_s, V_t) \tag{13}$$

Where $p(y \vee x; \theta)$ is the conditional probability of summaries given SoTs, and is calculated by the encoder-decoder model. $cos(V_s, V_t)$ is CS of semantic vectors $V_s$ and $V_t$. This term tries to maximize the SR between SoT and target output.

## 4. Experimental outcome

The SRN approach provided an experimental evaluation in three ways such as readability of the SiTs by using the palette of metrics of standard readability. The method evaluated the human sentence-level of simplified sentences namely simplicity, preservation of meaning and grammaticality with the original sentences. At last, the framework calculated the PE for measuring the time taken by the human to correct the errors produced by the TeSu and TeSi systems.

### 4.1. Database description

The simplification component approach implemented for simplifying the words with real-world events rather than the texts of descriptive structures with very few events. However, the existing simplification systems do not focus on events and were built using the English Wikipedia-Simple English Wikipedia (EW–SEW) corpus. Hence, the SRN method aimed to compare with those systems based on a collection of new stories (News dataset), the approach also evaluated Wikipedia articles (WiKi dataset) as well. The method considered 100 documents maintained by both datasets with at least 10 sentences each.

## 4.2. Evaluation metrics

The performance of the approach calculated by the following evaluation metrics like readability metrics, human-sentence level evaluation, and PE evaluation. The following sections describe the evaluation metrics:

### 4.2.1. Readability evaluation

The readability of simplification is measured by the approach with the help of Average Sentence Length (ASL) and the formula for this three readability are as follows:

- Flesch Reading Ease (FRE): The combination of ASL and the Average Number of Syllables (ASN) in a word are computed by the FRE score:

$$FRE = 206.835 - (1.015 \times ASL) - (84.6 \times ASW) \quad (14)$$

The more readable text depends on the high score of FRE.

- Fog Index (FOG): The ASL are combining with the average number of words that are more than two syllables (LEW) containing 100 words in textual fragments by using FOG.

$$FOG = 0.4 \times (ASL + LWS) \quad (15)$$

The low value of FOG index describes a number of readable texts.

- SMOG Grading score (SMOG): The SMOG scores are taken into consideration only the polysyllabic words in 30-sentences are average in long textual segments.

$$SMOG = 3 + \sqrt{polysyllable_{count}} \quad (16)$$

The SMOG scores not affected by the average sentence length of the SiT like other metrics, the evaluation of lexical complexity calculated by the SMOG scores. The more readable texts indicated by lower the values of SMOG.

The readability scores for the original texts obtained on the News and WiKi datasets presented in Table 1 and 2 and their simplified versions by four different systems automatically. The scores obtained for all the documents and the values reported for the original texts. The simplifications performed by systems such as SRN with the existing methods like the systems of Lexico-Syntactic (LS), LexEv and EvLex TeSi.

**Table 1:** Results of the readability evaluation for News dataset

| Database | Methods | FRE | FOG | SMOG | ASL |
|---|---|---|---|---|---|
| News | Original Text | 58.3 ± 10.9 | 13.4 ± 3.1 | 11.6 ± 2.2 | 24.8 ± 12.8 |
| | LS [19] | 66.9 ± 9.4 | 9.9 ± 2.5 | 9.7 ± 1.7 | 13.0 ± 8.7 |
| | LexEV [20] | 74.5 ± 9.4 | 7.4 ± 2.2 | 7.7 ± 1.6 | 7.6 ± 2.9 |
| | EvLeX [20] | 74.7 ± 10.0 | 7.3 ± 2.2 | 7.7 ± 1.6 | 7.5 ± 3.0 |
| | SRN | 78.8 ± 11.2 | 7.6 ± 3.6 | 7.9 ± 3.2 | 7.6 ± 3.9 |

**Table 2:** Results of the Readability Evaluation for Wiki Dataset

| Database | Methods | FRE | FOG | SMOG | ASL |
|---|---|---|---|---|---|
| WiKi | Original Text | 47.9 ± 11.7 | 15.5 ± 2.6 | 13.4 ± 1.9 | 27.0 ± 13.6 |
| | LS [19] | 55.6 ± 10.8 | 12.5 ± 2.3 | 11.4 ± 1.5 | 17.1 ± 10.7 |
| | LexEV [20] | 58.8 ± 17.5 | 10.1 ± 3.3 | 9.2 ± 1.4 | 9.1 ± 3.5 |
| | EvLeX [20] | 59.8 ± 15.6 | 9.9 ± 2.9 | 9.1 ± 1.5 | 9.2 ± 3.4 |
| | SRN | 61.5 ± 18.9 | 12.5 ± 4.6 | 9.5 ± 2.0 | 10.2 ± 4.8 |

The WiKi dataset provides the readability indices with the help of SMOG and FOG, the readable texts significantly produced by SRN approach. The ASL is the largest term provided by the difference between the systems. The simplified sentences transformed from the event mention of the original text, therefore this approach produced significantly average shorter sentences.

### 4.2.2. Human sentence-level evaluation

The semantic and grammaticality aspects of the SiT do not take the account of readability scores. Hence, the evaluation of complement readability calculated with the metrics like meaning preservation, simplicity, and grammaticality on the sentence level in the TeSu and TeSu. The cognitive effort on annotators imposed by evaluating the properties of semantic and syntactic of text, the entire documents sentences performed by the human evaluations.

From the two datasets, the approach selected 50 sentences randomly and the performance of simplification evaluations is measured by two annotators independently produced by three systems in terms of their:

- Grammaticality (Gram), i.e. the grammatical correctness;
- Simplicity (Simp), which resembles the simplified sentences in a simple way, i.e., both in terms of its vocabulary and its syntactic structure.
- Meaning preservation (MP), describes how well the original meaning is preserved by simplified sentences.

The arithmetic mean of the three assigned scores calculated from the scores for each simplification (Avg), i.e. $Avg = (Gram + Simp + MP)/3$. Table 3 presented the values of Inter-Annotator Agreement (IAA), which measured in terms of quadratic Cohen's kappa for both datasets.

**Table 3:** The IAA for Human Sentence-Level Evaluation

| Dataset | Gram | Simp | MP |
|---|---|---|---|
| News | 0.778 | 0.615 | 0.590 |
| WiKi | 0.640 | 0.537 | 0.656 |

The IAA agreement values are observed, after that the framework calculated the average scores of the system with the help of two annotators. The table 4 and 5 describe the calculation values of the human sentence-level for the two databases.

**Table 4:** The News Dataset Validate the Results of Human Sentence-Level

| Database | Methods | Gram | Simp | MP | Avg |
|---|---|---|---|---|---|
| News | Original Text | 4.96 ± 0.13 | 3.31 ± 0.74 | - | - |
| | LS [19] | 3.59 ± 0.78 | 3.42 ± 0.91 | 3.74 ± 1.16 | 3.56 ± 0.75 |
| | LexEV [20] | 4.23 ± 0.85 | 4.49 ± 0.70 | 3.57 ± 0.92 | 4.10 ± 0.71 |
| | EvLeX [20] | 4.25 ± 0.77 | 4.49 ± 0.68 | 3.55 ± 0.96 | 4.10 ± 0.69 |
| | SRN | 4.45 ± 0.85 | 4.58 ± 0.59 | 4.16 ± 0.90 | 4. 25 ± 0.71 |

**Table 5:** The Results of the Evaluation of Human Sentence-Level for Wiki Dataset

| Database | Methods | Gram | Simp | MP | Avg |
|---|---|---|---|---|---|
| WiKi | Original Text | 4.95 ± 0.15 | 3.33 ± 0.60 | - | - |
| | LS [19] | 4.15 ± 0.80 | 3.58 ± 0.84 | 3.96 ± 1.22 | 3.90 ± 0.70 |
| | LexEV [20] | 4.35 ± 0.66 | 4.30 ± 0.65 | 3.58 ± 1.02 | 4.07 ± 0.77 |
| | EvLeX [20] | 4.35 ± 0.63 | 4.30 ± 0.60 | 3.54 ± 1.02 | 4.05± 0.64 |
| | SRN | 4.47 ± 0.59 | 4.60 ± 0.57 | 4.20 ± 0.99 | 4. 27 ± 0.84 |

According to the results from the above tables, the proposed SRN framework provided better results compared to the other compet-

ing systems on the two datasets such as News and WiKi databases.

### 4.2.3. Post-editing evaluation

The SRN approach measured the PE time necessary to correct errors in grammaticality and meaning preservation. The SiT has these errors as very frequent; hence, an additional method is introduced before presenting to the final users, i.e. PE method. In this case, the calculation of PE time seems like an additional evaluation method. The framework finds the evaluation particularly suitable for TeSu and TeSi system, which performs any kind of content reduction without omission of sentence part that can often change the intended meaning.

From the two datasets, the method randomly selected 10 documents, the annotator doesn't edit the simplified versions of the same original documents ensured by the framework. PE requires less time for the next simplification compare to the first simplification because of the familiarity of the annotator. The tables 6 and 7 represented the results for evaluating the PE values on databases such as News dataset and WiKi dataset. The graphical representation of PE seconds/documents values described in the below figures 3 (News dataset) and 4 (WiKi dataset).

**Table 6:** Post Editing Results

| | SRB | | | LS [19] | | |
|---|---|---|---|---|---|---|
| Article ID | sec/doc | #Sent | sec/sent | sec/doc | #Sent | sec/sent |
| 616 | 60 | 7 | 6.9 | 170 | 13 | 13.1 |
| 2602 | 125 | 11 | 9.6 | 454 | 17 | 26.7 |
| 13552 | 259 | 14 | 15.3 | 456 | 25 | 18.2 |
| 15937 | 256 | 10 | 22.1 | 291 | 21 | 13.9 |
| 16437 | 150 | 7 | 16.2 | 185 | 15 | 12.3 |
| 2513 | 69 | 4 | 10.2 | 97 | 11 | 8.8 |
| 7958 | 165 | 9 | 12.4 | 209 | 12 | 17.4 |
| 16443 | 442 | 14 | 21.8 | 398 | 19 | 20.9 |
| 18877 | 134 | 8 | 11.7 | 254 | 20 | 12.7 |
| 23913 | 65 | 6 | 7.5 | 102 | 10 | 10.2 |
| Average | 175.7 | 9.8 | 13.0 | 261 | 15.2 | 15.4 |

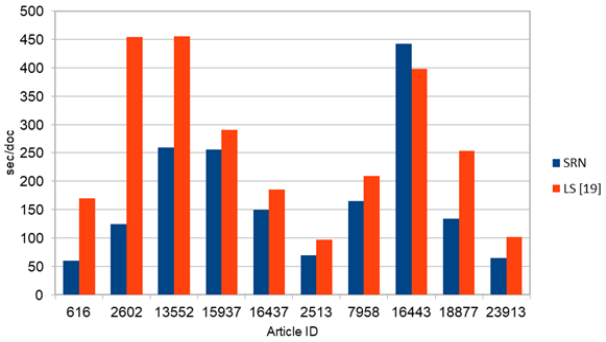


**Fig. 3:** Graphical Representation of News Dataset.

**Table 7:** Post-Editing Results

| | SRN | | | LS [19] | | |
|---|---|---|---|---|---|---|
| Article | sec/doc | #Sent | sec/sent | sec/doc | #Sent | sec/sent |
| Afghanistan | 320 | 13 | 20.7 | 498 | 25 | 19.9 |
| Alan turning | 354 | 18 | 15.4 | 434 | 36 | 12.1 |
| Am. English | 409 | 11 | 27.5 | 350 | 21 | 16.7 |
| Angola | 238 | 10 | 16.4 | 419 | 26 | 16.1 |
| Atom | 185 | 7 | 22.4 | 203 | 20 | 10.15 |
| Bottle | 400 | 12 | 27.6 | 320 | 21 | 15.2 |
| City | 320 | 13 | 23.1 | 363 | 26 | 14.0 |
| Food | 275 | 7 | 303.7 | 481 | 28 | 17.2 |
| France | 115 | 4 | 20.4 | 579 | 23 | 25.2 |
| Glass | 174 | 8 | 20.7 | 340 | 21 | 16.2 |
| Average | 284.2 | 11.2 | 21.9 | 398.7 | 24.7 | 16.1 |

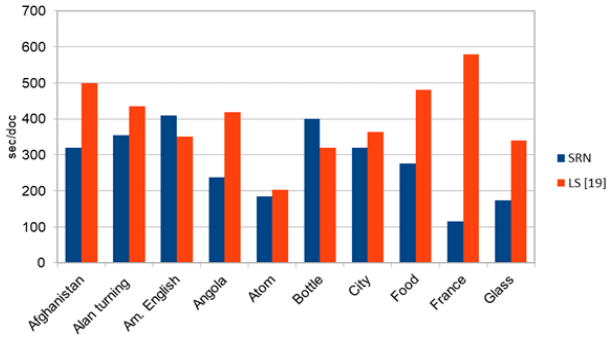


**Fig. 4:** The Result of PE Values for Wiki Dataset.

The two systems provided the simplified documents that differ greatly in the total number of sentences, the method produced PE results both in time per sentence (sec/sent) and in per-document time (sec/doc). The difference in the representation of generated sentences is especially noticeable in the Wiki dataset, which contains many descriptive sentences that not refers a single event. The simplifications provided by SRN, the Average PE (APE) time per document (sec/doc) is shorter on both datasets. In news dataset, the APE time for sentence (sec/sent) is shorter for simplify the values produced by SRN, whereas, on the Wiki dataset, the result of the APE is also shorter for the output of the LS system. The reason behind for that is the system (WiKi) copying various original sentences to the output without any changes and provides less content reduction.

## 5. Conclusion

In our routine lives, many texts encountered (e.g. Wikipedia articles or news articles) can be syntactical, semantically, or lexically complex for more audiences, especially users with non-native speakers, autism spectrum disorders, cognitive disabilities, etc. At once, these texts suffer some difficulties for various NLP tasks and tools, e.g. summarisation, parsing, and machine translation. Though there are various guidelines for how to write easy-to-read documents, manual simplification of already existing articles is costly and time-consuming and can't keep up with the pace with which new texts are being published. This created the need for TeSu and TeSi, the SiTs must have high SR to the SoTs. However, current sequence-to-sequence models tend to produce grammatical and coherent SiTs regardless of the SR to SoTs. The summary created by a sequence-to-sequence model (Seq2seq) is similar to the SoT literally, but it has low SR. In this approach, the main aim is to improve the SR between SoTs and generated SiTs for TeSu and TeSi for achieving this goal, the paper proposes an SRN. The component similarity introduced in this paper to calculate the relevance of SoTs and GTs. During training, the framework maximizes the SS to encourage high SR between SoTs and SiTs. In order to represent better, a long SoT, the SRN method introduced a self-gated attention encoder to memory the input text. The experimental results stated that this approach provided better results compared to the existing methods. The SRN method simplifies the small text in this work, further by improving the stability of parameters like readability metrics, human-sentence level evaluation, PE evaluation and also simplify the larger texts by using other techniques as a future work. In the present research work, TeSu work performed in single document, so the time complexity occurred for summarizing the whole documents. In future, this work can be extend using query-based method in multiple document and it may improve performance of document summarization in different fields like opinion and biomedical summarization.

## References


[1] A Ou YY, Kuan TW, Paul A, Wang JF, & Tsai AC (2017), "Spoken dialog summarization system with HAPPINESS/SUFFERING factor recognition," *Frontiers of Computer Science*, Vol. 11, No. 3, pp. 429-443. https://doi.org/10.1007/s11704-016-6190-2.

[2] Singh J, Singh G, Singh R, & Singh P (2018), "Morphological Evaluation and Sentiment Analysis of Punjabi Text using Deep Learning Classification," *Journal of King Saud University-Computer and Information Sciences*. https://doi.org/10.1016/j.jksuci.2018.04.003.

[3] Zhang S, Wei Z, Wang Y, & Liao T (2018), "Sentiment analysis of Chinese micro-blog text based on extended sentiment dictionary," *Future Generation Computer Systems*, Vol. 81, pp. 395-403. 8 https://doi.org/10.1016/j.future.2017.09.048.

[4] Zhang Y, Xia Y, Liu Y, & Wang W (2015), "Clustering sentences with density peaks for multi-document summarization," *Proceedings of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1262-1267.

[5] Canhasi E, & Kononenko I (2016), "Weighted hierarchical archetypal analysis for multi-document summarization," *Computer Speech & Language*, Vol. 37, pp. 24-46. https://doi.org/10.1016/j.csl.2015.11.004.

[6] Canhasi E, & Kononenko I (2014), "Multi-document summarization via archetypal analysis of the content-graph joint model," *Knowledge and information systems*, Vol. 41, no. 3, pp. 821-842. https://doi.org/10.1007/s10115-013-0689-8.

[7] Raposo F, Ribeiro R, & de Matos DM (2015), "On the application of generic summarization algorithms to music," *IEEE Signal Processing Letters*, vol. 22, no. 1, pp. 26-30. https://doi.org/10.1109/LSP.2014.2347582

[8] Bhargava R, Sharma Y, & Sharma G (2016), "Atssi: Abstractive text summarization using sentiment infusion," *Procedia Computer Science*, vol. 89, pp. 404-411. https://doi.org/10.1016/j.procs.2016.06.088.

[9] Chen MY, Huang TC, Shu Y, Chen CC, Hsieh TC, & Yen NY (2018), "Learning the Chinese Sentence Representation with LSTM Autoencoder," *Proceedings of the Companion of the Web Conference*, pp. 403-408. https://doi.org/10.1145/3184558.3186355.

[10] Yahaya MF, Rahman NA, Bakar ZA, & Hasmy H (2017), "Evaluation On Knowledge Extraction and Machine Learning in Resolving Malay Word Ambiguity," *Journal of Fundamental and Applied Sciences*, Vol. 9, no. 5S, pp. 115-130. https://doi.org/10.4314/jfas.v9i5s.10.

[11] Eddington CM, & Tokowicz N (2015), "How meaning similarity influences ambiguous word processing: The current state of the literature," *Psychonomic bulletin & review*, Vol. 22, no. 1, pp. 13-37. https://doi.org/10.3758/s13423-014-0665-7.

[12] Gautam G, & Yadav D (2014), "Sentiment analysis of twitter data using machine learning approaches and semantic analysis," *Proceedings of the Contemporary computing (IC3)*, pp. 437-442.

[13] Xia Y, Cambria E, Hussain A, & Zhao H (2015), "Word polarity disambiguation using bayesian model and opinion-level features," *Cognitive Computation*, Vol. 7, no. 3, pp. 369-380. https://doi.org/10.1007/s12559-014-9298-4.

[14] Wang L, Niu J, Song H, & Atiquzzaman M (2018), "SentiRelated: A cross-domain sentiment classification algorithm for short texts through sentiment related index," *Journal of Network and Computer Applications*, Vol. 101, pp. 111-119. https://doi.org/10.1016/j.jnca.2017.11.001.

[15] Abdi A, Shamsuddin SM, & Aliguliyev RM, "QMOS: Query-based multi-documents opinion-oriented summarization," *Information Processing & Management*, Vol. 54, no. 2, pp. 318-338.

[16] Nagwani NK (2015), "Summarizing large text collection using topic modeling and clustering based on MapReduce framework," *Journal of Big Data*, Vol. 2, no. 1, pp. 6. https://doi.org/10.1186/s40537-015-0020-5.

[17] Xiong S, Wang K, Ji D, & Wang B (2018), "A short text sentiment-topic model for product reviews," *Neurocomputing*, Vol. 297, pp. 94-102. https://doi.org/10.1016/j.neucom.2018.02.034.

[18] Song S, Huang H, & Ruan T (2018), "Abstractive text summarization using LSTM-CNN based deep learning," *Multimedia Tools and Applications*, pp. 1-19.

[19] Mandya AA, Nomoto T, & Siddharthan A (2014), "Lexico-syntactic text simplification and compression with typed dependencies," *Proceedings of the 25th International Conference on Computational Linguistics*.

[20] Štajner S, & Glavaš G (2017), "Leveraging event-based semantics for automated text simplification," *Expert systems with applications*, Vol. 82, pp. 383-395. https://doi.org/10.1016/j.eswa.2017.04.005.