Text summarization evaluation using semantic probability distributions

Anh Le
Henry M. Gunn High School
Palo Alto, California USA
al50837@pausd.us

Fred Wu
Dept. of Computer Science
W. Virginia State Univ. WV. USA
heng.wu@wvstateu.edu

Lan Vu
Broadcom, Inc.
Palo Alto, California USA
lan.vu@broadcom.com

Thanh Le
UEH University
Hochiminh City, Vietnam
lntmail@yahoo.com

***Abstract*—Most popular methods for evaluation of automatic summarized text content employ some protocol that requires gold-standard summary, usually made by human, for validating the summarized text content based on some content comparison. These evaluation methods are however unable to function in case human-made summaries are not available, or improperly functioning when these summaries are in poor quality. In this paper, we proposed SESP, a novel evaluation method using content based approach. SESP applies advanced text tokenization methods and semantic based similarity metrics to generate semantic probability distributions of text contents. The probability distributions are then used for evaluating summarized text content given the original text document. We showed that SESP functions without a need for gold-standard summaries, but yet achieving better performance compared with the state of the art methods that require human-made summaries.**

Keywords—text summarization, lemmatization, part of speech, content-based approach, probability divergence.

I. INTRODUCTION

Automatic Text Summarization (ATS) refers to the task that utilizes techniques of Natural Language Processing (NLP) to automatically produce the shorter piece of text for a given original text document while keeping the source information content. Current state of the art ATS includes methods using extractive approach and those using abstractive one. Given a text document, extractive methods analyze the document, searching for most important passages, then using them to generate summarized content for the original document. Abstractive methods, on the other hand, employs some particular Natural Language Understanding (NLU) in NLP, including grammars and lexicons, to first understand the original document, then to generate summarized content that best describes the document. The summarized contents generated by abstractive methods therefore may not exactly consist of the verbatim sentences of the most important passages in the source document. While most of ATS models are based on extractive approach, recently developed abstractive based models have shown their significant advantages, particularly those using generative AI approach, e.g. chatGPT based models. That motivates a need for effective methods for evaluation of ATS models.

Methods for ATS model evaluation can be broadly classified into two categories: intrinsic and extrinsic. Extrinsic evaluation assesses summarized content based on its utility in given application context; e.g. the relevance, reading comprehension, etc. It can involve a comparison with the original document or summary written by a human expert, measuring how many main ideas of the document are covered by the summarized content. Intrinsic evaluation, on the other hands directly assesses the summarized content for the coherence, faithfulness, linguistic and content quality [1]. Both categories are however facing the problem of defining proper validation factors, metrics and their usage. The most popular metrics for intrinsic evaluation, to date, are precision, recall and F-score, which measure the overlap between summarized contents and human-made summaries.

Intrinsic evaluation methods, thanks to their advantages in terms of computational time and annotation cost, have been used by the great majority of research papers on ATS model evaluation [2] [3]. Lin [2] proposed content based metrics which measures the number of overlapping textual units (n-gram word sequences) between the ATS content and the gold-standard summaries for the model evaluation. Ng et al. [4] extended the work of [2] by using a word embedding technique to make possible soft lexicon matching that allows approximate similarity among tokens. Peyrard et at. [5] combined the metrics of [2] and [4] to build better metrics for ATS evaluation. Zhao et al. [6] applied a distance measurement on n-gram embedding pooled from BERT representation to define new semantic metrics to measure semantic distance between ATS contents and golden summaries. Zhang et al. [7], on the other hand, proposed an alignment method on token level to bring a better similarity measurement for ATS contents and human-made summaries.

Among the state of the art (SOTA) intrinsic evaluation methods, ROUGE is most widely used due to its high correlation with the manual assessment process [2]. Its key feature of supporting different methods for both text content representation and similarity definition encourages the research community to contribute. However, ROUGE and the similar methods mainly use n-gram technique which is known suffering from capturing linguistic patterns. Most of them do not allow semantic similarity among tokens which unfortunately reduces the performance of evaluation metrics. In addition, all of them require gold-standard summaries for evaluation process, and are therefore inapplicable when lacking human help.

In this research, we proposed SESP, a novel method for ATS model evaluation. SESP applies POS (Part Of Speech) in NLP to tokenize and to convert text contents into semantic probability distributions. Distribution divergence measures are used to compute the sematic similarity between ATS content and original text document for ATS model evaluation. SESP does not require human-made summaries. We showed that SESP outperformed ROUGE on some real-world datasets.

2769-5654/23/$31.00 ©2023 IEEE
DOI 10.1109/CSCI62032.2023.00038

II. TEXT SUMMARIZATION

A. Text summarization

Automatic text summarization (ATS) is a process of finding a subset of as few as possible sentences or passages from a given document, but consists of as much as possible information content of the document. There exists two types of text summarization: extractive and abstractive. Extractive methods usually reuse sentences and passages from the original document for building ATS content. The earliest work on extractive summarization was done by Luhn [8]. This method utilizes statistical analysis of word distributions to select the most important words, sentences for ATS content. TextRank (Mihalcea et al.) [9] on the other hand converts text content into graph. The graph model is used for ranking words and sentences of the original document. Ranking results are employed for generating ATS content. Similar to that of [9], LexRank method (Erkan et al.) [10] also relies on document graph model to search for the most important sentences using concept of sentence salience. In a different approach, Steinberger et al. [11] proposed an LSA based model to capture the key concepts in original document for determining the most important sentences. Selected sentences are used for ATS content.

Extractive method have the advantage of preserving factual information, the summarized content however can be hard to read. This is where the abstractive methods step in. These methods rely on NLU techniques to encode information content of the document, and to decode the information for ATS content. ChatGPT [12], for example, is a modern abstractive ATS model. Abstractive methods can produce better ATS contents in terms of coherence, they may however fall short of faithfulness.

B. Co-Selection evaluation metrics

The main evaluation metrics of co-selection are precision, recall and F-score. Precision score determines what fraction of the sentences selected by ATS model are correct, and is computed, as in (1), using the number of sentences occurring in both ATS content and gold-standard summary divided by the number of sentences in the ATS content. Recall score, on the other hand, determines what proportion of the sentences chosen by humans are selected by the ATS model, and is defined, as in (2), by the number of sentences occurring in both ATS content and gold-standard summary divided by the number of sentences in the gold-standard summary.

Denote by X and Y, the automatic summarized content and the gold-standard summary, respectively.

$$Precision(X,Y) = \frac{|X \cap Y|}{|X|} \quad (1)$$

$$Recall(X,Y) = \frac{|X \cap Y|}{|Y|} \quad (2)$$

F-score is a composite measure that combines precision and recall. The basic way of how to compute the F-score is to count a harmonic average of precision and recall:

$$F_{score} = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (3)$$

More flexible combination form of F-score, which allows to choose favoring either precision or recall, and is defined in (4).

$$F_{score} = \frac{(\beta^2+1) \times Precision \times Recal}{\beta^2 \times Precision + Recall} \quad (4)$$

where β is a weighting factor that favors precision when $\beta > 1$, or else, when $\beta < 1$, favors recall.

C. ROUGE

ROUGE method employs gold-standard summaries to automatically validate summarized text contents based on n-gram overlaps [2]. The overlapping information is then used to compute evaluation metrics.

1) ROUGE-N

Traditional ROUGE method basically uses unigram (1-gram) and bigram (2-gram) overlaps [2], described below.

Denote by X and Y, the automatic summarized content with m tokens, and the gold-standard summary with n token, respectively. Let ROUGE-N be the ROUGE method which is unigram ROUGE, for n = 1, or bigram ROUGE for n = 2.

$$Precision(X,Y) = \frac{\sum_{S \in X} \sum_{gramN \in S} Count_{match}(gramN)}{\sum_{S \in X} \sum_{gramN \in S} Count(gramN)} \quad (5)$$

$$Recall(X,Y) = \frac{\sum_{S \in X} \sum_{gramN \in S} Count_{match}(gramN)}{\sum_{S \in Y} \sum_{gramN \in S} Count(gramN)} \quad (6)$$

ROUGE-N however has limitation in that can arbitrarily break some linguistic patterns in the text. Information content in the text therefore can be lost.

2) ROUGE-L

ROUGE-L is another popular variant of ROUGE. It is an update of traditional ROUGE method using longest common subsequences instead of n-grams in evaluation procedure. A subsequence of length k is a list of k increasing indices corresponding to the tokens in a given text. A common subsequence among an automatic summarized content X, and the gold-standard summary Y is a pair of subsequences $(i_1,\ldots, i_k)$ and $(j_1,\ldots, j_k)$ such that $X[i_l] = Y[j_l]$, $l=1,\ldots,k$; $X[i_l]$ and $Y[j_l]$ are tokens in X and Y. The longest one of such common subsequences can be determined efficiently using a Viterbi algorithm for sequence alignment [13], or any similar dynamic programming algorithm. The precision and recall for ROUGE-L are defined as in (7) and (8).

$$ROUGE\_L_{Precision} = \frac{LCS(X,Y)}{m} \quad (7)$$

$$ROUGE\_L_{Recall} = \frac{LCS(X,Y)}{n} \quad (8)$$

Where LCS(.) is the function that determines the length of the longest common sequence between automatic summarized content X, and the gold-standard summary Y.

ROUGE, similar to other co-selection based evaluation methods, ignores the fact that two sentences can contain the same information even if they are written differently. In addition, these methods do not allow synonym words. This

limitation degrades their performance, particularly in measuring similarity between ATS contents and gold-standard summaries.

III. PROPOSED METHOD

We proposed SESP, an ATS evaluation method that utilizes POS (Part Of Speech) based tokenization technique with semantic similarity for semantic probability distribution representation of information content in text document. Probability divergence measures are used to determine the similarity between ATS content and original document.

A. POS based tokenization

POS based tokenization technique of SESP aims at turning text document into informative facts. Each SESP fact is defined to be a tuple of tokens that are connected with linguistic relational patterns or rules, where the root is an object and its dependents are either attributes or actions of the object itself. Table 1 shows a list of rules that define possible tuples for facts.

Each sentence may have multiple facts of which each holds a piece of information content of the sentence. For instance, given the sentence S: "Autonomous cars shift insurance liability toward manufacturers".

TABLE 1 LIGUISTIC RELATIONAL RULES

Dependent	Root
	Noun
Adjective	Noun
Adjective	Proper noun
Adjectival modifier	Noun
Adjectival modifier	Proper noun
Verb	Nominal subject
Verb	Subject passive

Utilization of POS based tokenization technique and the concept of informative facts in SESP allows to address a common problem of SOTA methods in determining proper n-gram for text content tokenization, particularly, in case where the text content is token-sensitively consistent. For instance, given two sentences S1 and S2 [2] as followings.

S1. police kill the gunman

S2. the gunman kill police

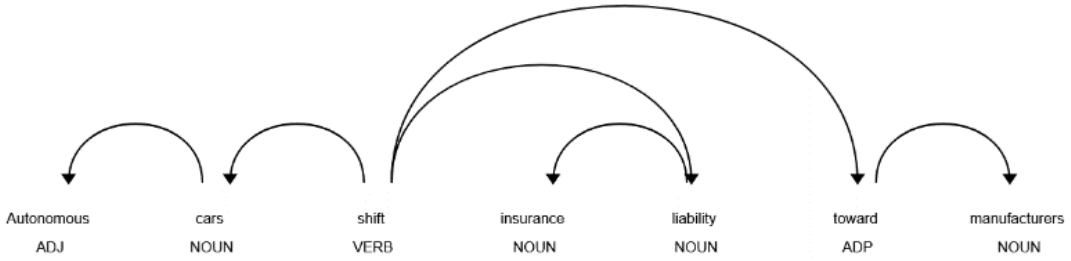

Fig. 1 POS based tokenization

Fig. 1 shows the root tokens with POS information and relevant facts of the sentence S. The word "car", of which the POS is of noun, is a root token and can join with the word 'autonomous' which is ADJ to build a fact "autonomous car". Hence, sentence S has a list of facts as followings: 'autonomous car', 'car', 'autonomous car shift', 'insurance liability', 'liability', 'shift insurance liability', 'manufacturer', 'toward manufacturer'.

Linguistic relational patterns found in a sentence may overlap. Some of them may be redundant and uninformative. For instance, the tuple of (subject, verb, object) when applied onto the sentence "Autonomous cars shift insurance liability" will create two overlapping facts: "Autonomous cars shift" and "shift insurance liability".

In order to reduce the number of facts as well as to eliminate redundant and uninformative ones, one can simply update the list of linguistic rules. For example, we proposed removing the last relational rule in Table 1 which is (verb, subject passive).

POS based tokenization of SESP will convert the two sentences S1 and S2 into token vectors V1 and V2 respectively as below:

V1 = ['police', 'police kill', 'the gunman', 'gunman',
'kill the gunman']

V2 = ['the gunman', 'gunman', 'the gunman kill',
'police', 'kill police']

Since V1 and V2 are very much different in terms of token elements, their representations in terms of probability distributions will be much different too.

B. Word embedding and token semantic similarity

In NLP, word embedding is about representation of words, facts, or even more, of sentences and documents, in an effective way to support further text analyses. The representation usually uses a real-valued vector that encodes both word meaning and word dictionary information such as type, synonyms and antonyms. Among of the most popular word embedding methods, GloVe (Global Vectors for Word Representation) is widely used these days. Glove works based on matrix factorization techniques using word-context matrix. A large

matrix of co-occurrence information is constructed by counting every word to see how frequently it is found in some "context" (relevant words) in a large corpus. A large corpus can be either all the documents made available on Wikipedia, or all the posts on Twitter. Fig. 2 shows an example on co-occurrence of the word w_k with the word w_i and w_j.

Probability and Ratio	$k = solid$	$k = gas$	$k = water$	$k = fashion$
$P(k\|ice)$	1.9×10^{-4}	6.6×10^{-5}	3.0×10^{-3}	1.7×10^{-5}
$P(k\|steam)$	2.2×10^{-5}	7.8×10^{-4}	2.2×10^{-3}	1.8×10^{-5}
$P(k\|ice)/P(k\|steam)$	8.9	8.5×10^{-2}	1.36	0.96

Very small or large:
solid is related to ice but not steam, or
gas is related to steam but not ice

close to 1:
water is highly related to ice and steam, or
fashion is not related to ice or steam.

Fig. 2 Example of word co-occurrence matrix

A neural network model was used to learn the co-occurrence matrix in the context of the selected corpus, resulting in the best representation, real-valued vector, of every word for the given context. Use of Glove word-embedding method allows both compact representation of words, relevant information and possible sematic similarity measurement between words and facts. For instance, it allows measuring how close, in terms of semantic similarity, the two different words are. This not only eliminates the problem of matching different words, but also enhances the process of measuring the similarity between ATS contents and the original document.

C. Probability distribution similarity

Probability distributions of the parsed tokens of text document are used to approximate information content of the document. Definition of semantic similarity between probability distributions is essential to compute the distance between AST contents and gold-standard summaries.

1) Kullback Leibler divergence

The Kullback Leibler (KL) divergence between two probability distributions P and Q is defined as in .

$$KL(P||Q) = \sum_w P_P(w) log_2(\frac{P_P(w)}{P_Q(w)}) \quad (9)$$

KL(.) is defined as the average number of bits wasted by coding samples belonging to P using another distribution Q, an approximate of P. In SESP method, P and Q stand for two probability distributions of words from ATS and the original document contents. KL divergence is not symmetric. The divergences of ATS content to gold-standard and gold-standard to ATS content are therefore not identical. In addition, there might be the case where $P_Q(w)$ divergence is undefined when $P_P(w) > 0$ but $P_Q(w) = 0$, resulting in undefinition of KL(.). A simple smoothing function in (10) can help with the problem.

$$p(w) = \frac{C+\delta}{N+\delta+B} \quad (10)$$

where C is the word count, N is the number of tokens, B = 1.5|V| and V is input vocabulary, δ was set to small value of 0.0005 to avoid shifting to much probability to unseen events.

2) Jensen Shannon Divergence

The Jensen Shannon (JS) divergence incorporates the idea that the distance between two probability distributions cannot be very different from the average of distances from their mean distribution. JS(.) is formally defined as in (11).

$$JS(P||Q) = \frac{1}{2}[KL(P||A) + KL(Q||A)] \quad (11)$$

where $A = \frac{P+Q}{2}$ is the mean distribution of P and Q.

In contrast to KL divergence, JS divergence is symmetric and is always defined. JS divergence is therefore used for computing the distance between semantic probability distributions. Since JS(.) is always defined, semantic similarity between any pair of text contents should be always defined.

D. SESP algorithm

- inputs: A (ATS content), D (original document)
- output: semantic similarity score (larger the better)

Step

1) Create the gold-standard tokens using D
2) Generate probability distributions P_A, P_D for the two text contents, A and D, respectively using gold-standard tokens
3) Return JS(P_A, P_D) using (11)
4) Stop

IV. Experiments

A. Datasets

To benchmark our proposed method (SESP) we used two datasets: CNN/Daily Mail [14] and SRAdb. CNN/Daily Mail dataset consists of more than 300,000 news articles from CNN and the Daily Mail newspaper, published between 2007 and 2015. We used the latest published version of the dataset. The test split from the dataset, consisting of 11,490 articles, was used.

SRAdb dataset contains the metadata of the Sequence Read Archive (SRA), publicly made available by the National Center for Biotechnology Information (NCBI) [15]. The metadata are about 331,837 research studies submitted to NCBI. However, there are only 16,929 studies that provide detail and brief information on their research experiments.

B. Test procedure and evaluation measures

For each document and its human-made summary, we applied ATS methods in section II.A, including TextRank (Rnk) [9], LSA based method [11], Luhn [8], LexRank (Lex) [10], and ChatGPT-2 (GPT2) [12] to generate ATS contents for the document. We only ran GPT2 on SRAdb because GPT2 is time consuming. We then applied SESP on every pair of the document and each of ATS contents created the ATS algorithms. For ROUGE-1, ROUGE-2 and ROUGE-L, we did the same but replaced original document by the gold-standard summary. Output scores were averaged by ATS algorithm and ATS evaluation method across the dataset. Final benchmark results were reported in section IV.C.

C. Experimental results

TABLE 2 EVALUATION OF ATS METHODS USING CNN-DAILYMAIL

	Rnk	Lsa	Luhn	Lex	Corr.
SESP	0.390	0.345	**0.400**	0.381	
ROUGE-1	**0.315**	0.266	0.303	0.302	0.903
ROUGE-2	**0.114**	0.078	0.106	0.099	0.925
ROUGE-L	**0.288**	0.242	0.275	0.274	0.899

Table 2 shows performance results of SESP and the ROUGE methods. The column "Corr." indicates a high correlation between SESP and ROUG ones. The highest correlation is of 0.925, which is between SESP and ROUGE-2. The correlation is slightly lower when comparing SESP with ROUGE-1, which is 0.903; and with ROUGE-L, which is 0.899. SESP however differed from the ROUGE methods in selecting the best ATS method; SESP picked Luhn while all ROUGE methods selected TextRank method. However, Luhn was also reported by a recent reseach work [16] as the best ATS method for CNN-DailyMail dataset.

D. NCBI SRAdb - metadata for RNAseq data

SRAdb dataset contains the metadata of high-throughput next-generation sequencing data publicly made available by NCBI. In addition to extractive method, we also employed an abstractive method (GPT2) to generate ATS contents for testing purpose.

TABLE 3 EVALUATION OF ATS METHODS USING SRADB

Method	Rnk	Lsa	Luhn	Lex	GPT2	Corr.
SESP	**0.750**	0.724	0.748	0.725	0.646	
ROUGE-1	**0.801**	0.777	0.793	0.774	0.678	0.996
ROUGE-2	**0.766**	0.729	0.763	0.733	0.575	0.997
ROUGE-L	**0.800**	0.775	0.791	0.773	0.675	0.996

Table 3 shows the performance results of SESP and the ROUGE methods. Information in the column "Corr" indicates that SESP had very high correlation with ROUGE ones, and most correlated with ROUGE-2 method. For the best ATS method, SESP picked TextRank which was also selected by all the ROUGE algorithms.

Benchmarks using the two datasets in section IV.A show that SESP outperformed the ROUGE methods. In addition, SESP does not require human-made summaries for its evaluation procedure. SESP is therefore more practical when applying for real-world problems.

V. CONCLUSIONS

We introduced SESP, a novel method for evaluation of automatic text summarization models. SESP applies advanced text tokenization methods and semantic similarity measures to properly parsing text data and to build highly accurate probability distributions of text contents. Using content probability distributions, SESP is able to validate ATS contents without a need for human-made summaries. We showed that SESP outperformed ROUGE, the most popular evaluation method for ATS models, on two real-world datasets, SRAdb and CNN/DailyMail from DUC.

Our future work includes additional benchmarks of SESP on more challenging datasets from DUC and TAC, and application of SESP in linguistic educational software.

REFERENCES

[1] G. J. Sparck Jones K., "Evaluating natural language processing," *Lecture notes in computer science,* 1996.

[2] C.-Y. Lin, "Rouge: A package for automatic evaluation of summries," *Text Summarization Branches Out,* 2004.

[3] W. K. B. M. C. X. R. S. D. R. Alexander R. Fabbri, "SummEval: Re-evaluating Summarization Evaluation," *Association for Computational Linguistics,* 2021.

[4] V. A. Jun-Ping Ng, "Better summarization evaluation with word embeddings for ROUGE," *Empirical Methods in Natural Language Processing,* p. 1925–1930, 2015.

[5] T. B. I. G. Maxime Peyrard, "Learning to score system summaries for better content selection," *Workshop on New Frontiers in Summarization,* pp. 74-84, 2017.

[6] M. P. F. L. Y. G. C. M. M. S. E. Wei Zhao, "MoverScore: Text generation evaluating with contextualized embeddings and earth mover distance," *EMNLPIJCNLP,* p. 563–578, 2019.

[7] V. K. F. W. K. Q. W. Y. A. Tianyi Zhang, "Bertscore: Evaluating text generation with BERT," *International Conference on Learning Representation,* 2020.

[8] L. H. P., "The Automatic Creation of Literature Abstracts," *IBM Journal of Research and Development,* vol. 2, pp. 159-165, 1958.

[9] P. T. Rada Mihalcea, "TextRank: Bringing Order into Text," *Empirical Methods in Natural Language Processing,* p. 404–411, 2004.

[10] D. R. R. Günes Erkan, "LexRank: graph-based lexical centrality as salience in text summarization," *Journal of Artificial Intelligence Research,* vol. 22, pp. 457-479, 2004.

[11] K. J. Josef Steinberger, "Using Latent Semantic Analysis in Text Summarization and Summary Evaluation," *7th International Conference ISIM,* 2004.

[12] OpenAI, "Text generated by ChatGPT," 15 Dec 2022. [Online]. Available: https://chat.openai.com.

[13] A. D. J. E.-S. Boraq Madi, "Textline alignment on the image domain," *International Journal on Document Analysis and Recognition,* vol. 25, no. 4, pp. 415-427, 2022.

[14] T. K. E. G. L. E. W. K. M. S. P. B. Karl Moritz Hermann, "Teaching Machines to Read and Comprehend," *Advances in Neural Information Processing Systems,* pp. 7-12, 2015.

[15] "The NCBI Sequence Read Archive," [Online]. Available: http://www.ncbi.nlm.nih.gov/sra.

[16] C. M. N. K. Nikolaos Giarelis, "Abstractive vs. Extractive Summarization: An experimental review," *Applied Science,* 2023.

[17] P. G. A. A. P. L. G. N. Manik Bhandari, "Re-evaluating Evaluation in Text Summarization," *Empirical Methods in Natural Language Processing,* pp. 9347-9359, 2020.

[18] D. G. Y. G. H. R. R. P. M. B. Y. A. I. D. Ori Shapira, "Crowdsourcing lightweight pyramids for manual summary evaluation," *Association for Computational Languistic: Human Language Technologies,* 2019.