# Deep Semantic Similarity Models for Evaluating Text Summaries

**Susmit Wagle**[1]**, Prerna Bharti**[2]**, Priyank Pathak**[3]**, Vivek Gupta**[4]**, Pegah Nokhiz**[5]**, Harish Karnick**[6]
Goldman Sachs[1], Microsoft R&D India[2], New York University[3], University of Utah[4,5], IIT Kanpur[6]
Bangalore[1,2], New York[3], Utah[4,5], Kanpur[6]
{waglesmit.uec21, prernarbharti}@gmail.com, pp1953@nyu.edu, {vgupta, pnokhiz}@cs.utah.edu, hk@iitk.ac.in

## Abstract

ROUGE (Lin, 2004), a lexical matching metric, is the most prominent evaluation metric in text summarization. However, it fails to succeed in capturing the semantics if there is a correlation on the sample level, particularly in abstractive summarization. For instance, ROUGE's performance is unsatisfactory when the dataset contains no news articles (Dohare et al., 2017). In this work, we use a semantic similarity-based measure instead of lexical matching to evaluate summaries which works better than ROUGE. We propose a novel deep learning approach, namely a Convolutional Deep Semantic Similarity Model (CDSSM) based evaluation metric in an end-to-end manner that aligns with human judgments. We also construct a new scientific dataset (Sum-PubMed) to show our metric's superior performance on these types of datasets compared to ROUGE.

## 1. Introduction

Abstractive document summarization is a well-known challenging problem in NLP, so a cheap, repeatable, and fast, automatic evaluation metric plays a pivotal role in this task. However, current evaluation methods for abstractive summarization, such as Recall-Oriented Understudy of Gisting Evaluation (ROUGE), mostly depend on word $n$-gram matching.

Simple word-overlap methods such as ROUGE cannot capture the semantics because several words can be of similar meanings (Lloret et al., 2018; Cohan and Goharian, 2016). Also, capturing the semantics is a hard task due to the words' multiple senses in various contexts, e.g., 'office', which can be a location or software in different contexts. Because of an exponentially larger sample space, brute-force approaches for semantic matching using a word-net dictionary are intractable. Furthermore, these metrics do not align well with human judgments, where the summary is not located at the beginning of the document in a summarization task (Dohare et al., 2017).

On the other hand, a deep semantic similarity model (DSSM) captures the semantic meanings of text efficiently (Gao et al., 2014b). The deep model maps the textual input to its latent semantic representation in the semantic space using a neural network. It assumes that similarity in the semantic space can be inferred as the semantic similarity of the input, which succeeds in several NLP tasks (Gao, 2017). This model has been successful in capturing the semantic similarity in multiple natural language processing tasks, namely web search (Huang et al., 2013; Shen et al., 2014; Palangi et al., 2016), entity linking (Gao et al., 2014b), online recommendations (Gao et al., 2014b), image capturing (Fang et al., 2015), machine translation (Gao et al., 2014a), and question answering (Yih et al., 2015).

Since ROUGE does not align well with human ratings (Dohare et al., 2017), we propose a DSSM using Convolutional Neural Networks (CNN) (Kim, 2014; Zhang and Wallace, 2017; Jiao et al., 2018) to project textual strings to the semantic space for evaluating abstractive summarization. Using CNNs, we benefit from parallelization and hierarchical representations over the input sequences for capturing long-range dependencies (Zhang et al., 2016) compared to chain-structured models such as Recurrent Neural Networks (RNN) (Mikolov et al., 2010), as stated in (Ruseti et al., 2018).

To assess our metric's performance on human judgments and compare it to ROUGE, we construct a scientific dataset (Sum-PubMed) using the PubMed directory, where the top-located sentences are not the summaries. We then show our metric's superior performance compared to ROUGE and its close alignment with human judgments. Our main contributions in this paper are:

1. We endorse previous observations by (Dohare et al., 2017; Cohan and Goharian, 2016) through our evaluation metric and show that ROUGE does not align well with human judgments for a summarization task where the summary is not located at the beginning of the document.

2. To address this issue, we propose a novel model-based automatic evaluation metric for abstractive summarization, which takes the semantical meaning of text rather than the lexical overlapping into account. We use a Convolutional Neural Network DSSM-based (CDSSM) (Gao, 2017) method for summary evaluation.

3. To assess our metric's performance on human judgments and compare it to ROUGE, we construct a scientific summarization dataset (Sum-PubMed) using the PubMed directory where the top-located sentences are not the summaries. In addition, we show that ROUGE does not align well with human judgments on this dataset.

In section (1.), we provided a brief introduction to the problem statement. The remaining parts of the paper are organized as follows: in Section (2.), we discuss related work in summarization. In Section (3.), we discuss a deep semantic similarity model (DSSM). We then move on to our two proposed models in section (4.). Next, we discuss our newly constructed scientific summarization dataset (SumPubMed) in section (5.), followed by experimental results and analysis in section (6.). Finally, we conclude our findings in Section (7.). We have also released the SumPubmed dataset along with the paper. [1]

## 2. Related Work

ROUGE is a well-known $n$-gram matching metric that measures the quality of a summary generated by a system. The pyramid method proposed by (Nenkova et al., 2007) compares the Summarization Count Units (SCUs) between the candidate and the reference. Smatch (Cai and Knight, 2013), another metric, matches the semantic structures, namely the Abstract Meaning Representation (AMR) of two sentences. A recent metric (Ng and Abrecht, 2015; ShafieiBavani et al., 2018b) evaluates the summarizations without human model summaries by utilizing the compositional attributes of corpus-based and lexical resource-based word embeddings. The extended version uses a graph-based summarization (ShafieiBavani et al., 2017; ShafieiBavani et al., 2018a). (Louis and Nenkova, 2013; Peyrard and Eckle-Kohler, 2017; Peyrard et al., 2017; Peyrard and Gurevych, 2018) propose an evaluation metric without reference summaries. Finally, (Genest et al., 2011) uses a deep model for evaluation (two summaries are inputs and supervised average regression scores are used for evaluation), but with simple averaging aggregation, the result is slightly worse than ROUGE-2's recall.

However, these metrics do not employ an unsupervised DNN framework based on several levels of similarities in latent semantic representations. Also, these metrics' performance is not assessed on scientific articles. It should also be noted that constructing domain-specific embeddings is ineffective due to a few statistical insights (jargon frequencies) in the domain-specific corpora for the jargon (Pilehvar and Collier, 2016).

## 3. Deep Semantic Similarity Model

DSSM computes the semantic similarity between two strings by mapping them into a common latent semantic space using deep neural networks. It is a dual-branch network with tied weights and branches merged at the top layer. The bottom-most layer is the input layer, where the text is passed as a sequence of words/a chunk. [2] The non-linear transformation layers can be either a feed-forward neural network, Convolutional Neural Network, or a seq2seq model such as an RNN. The output of the non-linear layer represents the common latent semantic space where cosine similarity[3] between the latent representations capture the similarity between the strings.

### 3.1. Deep Semantic Similarity Model's Basic Layers

We will first describe the common basic layers in DSSM frameworks. This framework is used in many NLP tasks, shown in Table 1. These layers are placed one after another in the following order:

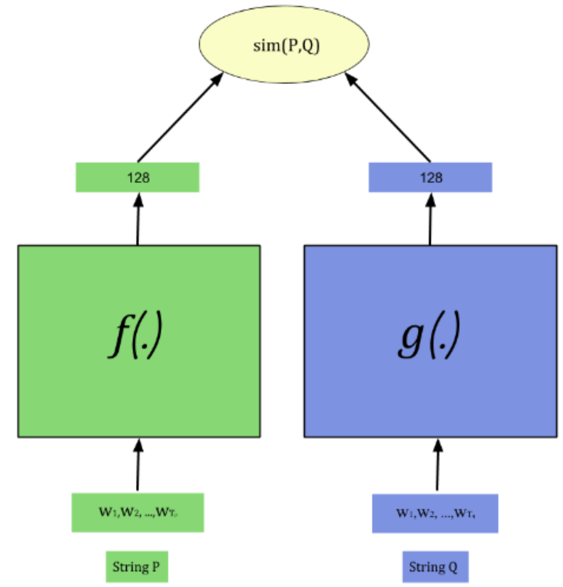


Figure 1: General DSSM Architecture (Gao, 2017)

**Data Processing and Input Layer:** We first preprocess the textual input as described in (Shen et al., 2014). We use the *Word Indexing* input layer, which maps each word to an index. The input layer converts the list of input word tokens of given strings to a list of indexes.

**Embedding Layer:** We then create an embedding layer which maps each index to a continuous dense embedding. The weights of an embedding layer are initialized with random values and are learned during the joint training. We can also initialize them to some pre-defined values such as pre-trained word embeddings. We have initialized the vocabulary words' embeddings with the average of the embeddings of all words in the vocabulary. All punctuation is removed, and the input is converted to lower case letters. Only alphanumeric characters are allowed, and the input string is white-space tokenized. We do not perform any data stemming. A input token string $S$ of size $N$ is passed through an embedding layer which returns an $N \times D$ matrix, where each row $(i)$ represents the continuous dense embedding of token $(i)$.

**Convolution and Max Over Time Pooling Layer:** Both functions $f(.)$ and $g(.)$ are featured in Figure 1. We use a convolution layer (Kim, 2014) with multiple filters followed by a max-over-time pooling operation (Collobert et al., 2011) to handle variable string lengths. Let $S$ be a string of length $N$ and $v_i \in R^K$ be the $K$-dimensional word vector representation of the $i^{th}$ word in the string. We can represent $S$ as:

$$S_{1:N} = v_1 \oplus v_2 \oplus .... \oplus v_N \tag{1}$$

here $\oplus$ is the concatenation operator. In the convolution operation, we have a filter $W \in R^{HK}$, which is applied[4]

[1] https://goo.gl/7WGZW7 [2] Either character embedding or word embedding are used. [3] Any other valid distance or similarity metric could be used.

[4] Convolution is a simple matrix multiplication operation.

Table 1: DSSM in various NLP tasks (Gao, 2017)

| **Task** | **String P** | **String Q** |
|---|---|---|
| Web Search | Search Query | Document Set |
| Entity Linking | Entity mention and context | Entity and its corresponding page |
| Online Recommendation | Document reading | Interesting things/other docs |
| Image Captioning | Image | Text |
| Machine Translation | Text string of language A | Translation in language B |
| Question Answering | Question | Answer |
| Summarization (our model) | Story | Summary |

to a word window of size $H$ to produce a new feature. For example, feature $C_i$ is generated using window of words $V_{i:i+H-1}$,

$$C_i = \phi(W.V_{i:i+H-1} + b) \tag{2}$$

where $b \in R$ is the bias term and $\phi$ is the activation function. Sliding this filter over string $S$ grants a feature map $C \in R^{N-H+1}$

$$C = [C_1, C_2, ..., C_{N-H+1}] \tag{3}$$

Then a max-over-time pooling operation (Collobert et al., 2011) is applied over this feature map $C$, which grants us the top feature $\hat{C}$ for filter $W$.

$$\hat{C} = \max\{C\} \tag{4}$$

This naturally helps us in handling variable length strings. In general, for a window size $H$, we apply multiple filters to get multiple different max-pooled features.

**Dense and Soft-max Layer:** Finally, a fully connected dense layer with a soft-max layer on a non-linear activation function was applied. There are several choices for an activation function; however, we choose a *sigmoid* function. We also use a neural network dropout regularizer, introduced by (Srivastava et al., 2014) to handle over-fitting.

Furthermore, a neuron of a dense layer gives us an output $y$, which is the weighted summation of its inputs. The value of $y$ can range from $-\infty$ to $+\infty$, which is not useful for model training. Therefore, we use an activation function which bounds the value of the output of a neuron. There are many choices for an activation function; however, some widely used functions are:

$$\sigma(y) = \frac{1}{1+e^{-y}} \in [0,1] \tag{5}$$

$$tanh(y) = \frac{e^{2y}-1}{e^{2y}+1} \in [-1,1] \tag{6}$$

$$ReLU(y) = \max(0, y) \in [0, \infty) \tag{7}$$

Dropout, introduced by (Srivastava et al., 2014), is a technique to handle over-fitting. It prevents the neural network from over co-adapting. During training, it drops a neuron's output which means the output of the neuron is set to zero, with probability $p$ during forward-back propagation passes. At testing time, $p$ is set to 1. In all our experiments, we apply activation and dropout over the output of every convolution and dense layer.

# 4. Our Proposed Models

We will now discuss our two proposed CDSSM frameworks for automatic summary evaluations.

**Single-layer Model (SL):** In this model, we use the CNN model for each CDSSM subnet. However, because of the heterogeneity between the story (original document) and summary texts in their lengths and styles, the branches have their own set of layer weights, window sizes, and the number of filters. The complete architecture of our model with all components is shown in Figure 2.

*The Story subnet* projects the document and *the Summary subnet* projects the corresponding document summary to the common intermediate feature space. The output of both subnets is passed through a common, fully connected (dense) layer to obtain the corresponding latent representation. We then use the cosine similarity between the latent representations to capture the semantic similarity between the document and the summary. We use a discriminative training approach similar to (Gao et al., 2014b; Shen et al., 2014) on the unsupervised data to train our model. We only have the story-summary pair in our training data. Each training sample consists of a story-summary pair (positive-pair) and *m*, a hyper-parameter indicating the count of other story documents as negative samples. The model is trained as a classifier that learns to assign soft-max probabilities of true class (positive pair) as 1 and the rest as 0. At the test time, we remove the soft-max layer and only consider the cosine similarity as our output.

**Multi-layer with $k$-max Pooling (ML):** In the single-layered CNN model, we use a simple max-over-time pooling to select only the top features. However, the drawback is that for two different summaries, it will pool similar features corresponding to a general topic in common in both summaries. Also, it cannot capture whether a relevant feature in a row occurs multiple times or just once. Moreover, it neglects the position of a top feature in the row. Thus, to grade summaries on a finer level, we need to have a hierarchical architecture. In the bottom layers, we can have a large window to cover larger contexts, and as we go up the hierarchy, the context becomes more focused and smaller. We use a simplified version of the model defined by (Kalchbrenner et al., 2014), which uses $k$-max-over-time pooling operations on both subnets. Each convolution layer also uses multiple window sizes with multiple filters. We show the architecture of our model in Figure 3.

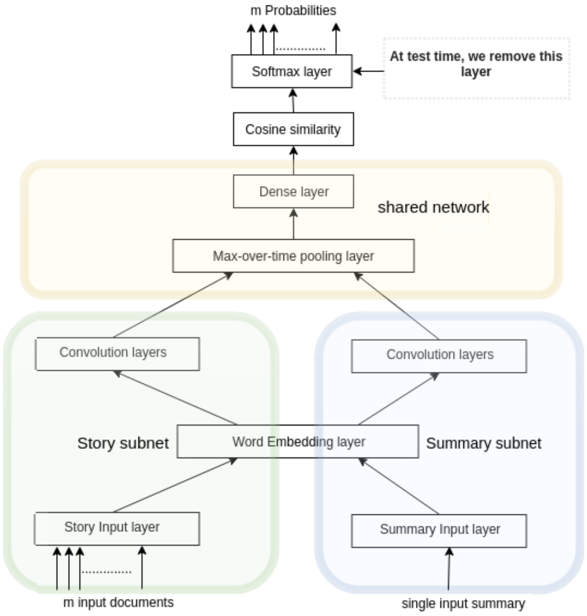


Figure 2: First Model's SL-CDSSM Architecture

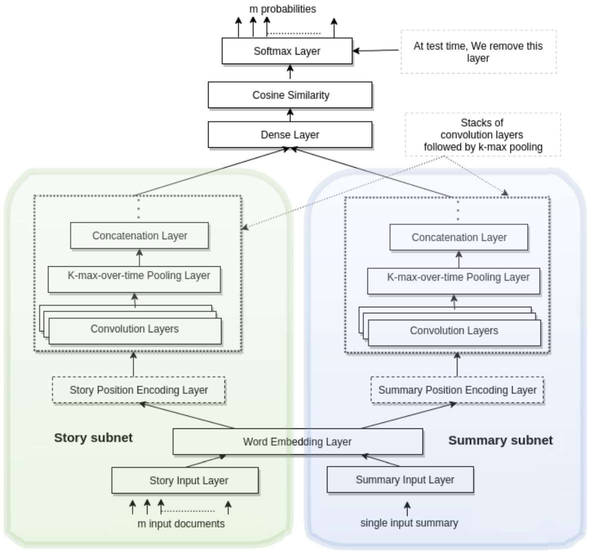


Figure 3: Second Model's ML-CDSSM Architecture

**Position Encoding (MLW):** A convolutional neural network with $k$-max pooling, preserves some order of the top-$k$ features, but it is still insensitive to the words' positions. To capture coherence in a summary, we inject additional information of absolute/relative positions in our model using positional encoding (Vaswani et al., 2017) on the input embeddings in both subnets. The positional embeddings have the same dimension as the word embeddings to enable their aggregation. The story and summary position encoding layers are shown in Figure 3 (dashed boxes).

## 5. New Dataset (Sum-PubMed)

The main problem with the news dataset is that the entire summary of documents can be found in the first few lines. Also, in CNN Dailymail, news highlights are considered as golden summaries.

Table 2: Single Layer CDSSM architecture

| Hyper Parameter | Value |
|---|---|
| # Convolution layers | 1 |
| Layer-1 | [(3,100),(4,100),(5,100)] |
| Dropout | 0.8 |
| Word embedding size | 300 |
| Dense layer input size | 300 |
| Dense layer output size | 128 |

Therefore, we create a new summarization dataset named Sum-PubMed which is a dataset of scientific articles in the biomedical domain due to the following reasons: 1) Sum-PubMed has longer documents, 2) Sum-PubMed is constructed of documents with scientific terms, and 3) The first few lines in the dataset are not the summaries. Each document consists of two parts, an abstract and a full text. An abstract can be considered as the summary of the full text. In this dataset, each document is a research paper resulting in very large document sizes. Thus, it is difficult to pre-process this dataset with available resources. So, we create our own multi-stage pre-processing pipeline. For each sentence in the raw text we use our own re-gex script which a) replaces citations and digits with <cit> and <dig> labels, b) removes figures, tables, captions, and related lines, and c) removes acknowledgment and references from the text. We then use a $SAX\ parser$, which converts each sentence in the raw text to a structured $XML$. We choose the documents that have fewer than 2,000 words after applying the extractive summarization using the text-rank algorithm (33% retained). Furthermore, to ensure that the summary and the document have some named entities, we perform a noun intersection that uses a noun dictionary and a standard NER/ABNER.

The final statistics of the dataset is shown in Table 5. The Sum-PubMed dataset is publicly available for research purposes. [5] Recently, (Cohan et al., 2018) released a PubMed based summarization dataset; however, unlike our dataset no extensive pre-processing pipeline was applied to clean the text in their approach. Moreover, our summary length and vocabulary size is much larger compared to the previous dataset.

**Sum-PubMed Dataset Construction Example:** Firstly, the research document in the raw form has two parts: front and body. The front part of the document is basically the abstract, which has three subsections: background, results, and conclusion, shown in Figure 4. The body part of the document is basically the text, which has three subsections: background, results, and conclusion, as shown in Figure 5. The raw text contains figures, tables, citations, digits, acknowledgments and references, as shown in Figure 5. We cleaned them by using reg-ex (Figure 6). The $xml$ file is formed from the cleaned text shown in Figure 7. Next, we form two folders: abstract, which contains the abstract part and the text folder which is of the text part of all research documents. Now the abstract and text are ready to be used for running seq2seq models. The cleaned files are shown in Figure 8 and Figure 9. Figure 10 shows the Sum-PubMed example, i.e., the article and the summary.

[5] `https://goo.gl/7WGZW7`

Table 3: Multi-layer CDSSM architecture

| | Hyper Parameter | Value |
|---|---|---|
| Story subnet | # Convolution layers | 4 |
| | Layer-1 | [(5, 30), (6, 30), (7, 40), (8, 40), (9, 30), (10, 30)], 200 |
| | Layer-1 | [(4, 20), (5, 30), (6, 30), (7, 20)], 100 |
| | Layer-3 | [(3, 20), (4, 30), (5, 20)], 50 |
| | Layer-4 | [(2, 20), (3, 20)], 25 |
| Summary subnet | Layer-1 | [(1, 20), (2, 30), (3, 30), (4, 20)], 10 |
| General | Dropout Prob | 0.8 |
| | Word embedding size | 300 |
| | position encoding size | 300 |
| | Dense layer input size | 1000 |
| | Dense layer output size | 300 |
| | Optimizer | Adam |
| | Initial Learning Rate | 0.001 |
| | Batch size | 60 |
| | # Negative samples | 3 |

Table 4: Model ML2 Architecture

| | Hyper Parameter | Value |
|---|---|---|
| Story subnet | # Convolution layers | 7 |
| | Layer-1 | [(10, 30), (11, 30), (12, 30)], 1280 |
| | Layer-1 | [(7, 30), (8, 30), (9, 30)], 640 |
| | Layer-3 | [(5, 40), (6, 40), (7, 40)], 320 |
| | Layer-4 | [(4, 40), (5, 40), (6, 40)], 160 |
| | Layer-5 | [(3, 40), (4, 40), (5, 40)], 80 |
| | Layer-6 | [(4, 30), (5, 30)], 40 |
| | Layer-7 | [(3, 30), (4, 30)], 20 |
| Summary subnet | # Convolution layers | 4 |
| | Layer-1 | [(5, 30), (6, 30), (7, 30)], 80 |
| | Layer-2 | [(3, 40), (4, 40), (5, 40)], 40 |
| | Layer-3 | [(3, 30), (4, 30)], 20 |
| Common Parameters | Dropout Prob | 0.8 |
| | Word embedding size | 300 |
| | Dense layer input size | 1200 |
| | Dense layer output size | 128 |
| | Optimizer | SGD |
| | Learning Rate | 0.1 |
| | Batch size | 40 |
| | # Negative samples | 3 |

Table 5: Sum-PubMed Dataset Statistics

| Statistic | Story | Summary |
|---|---|---|
| Max Length | 2000 | 647 |
| Avg Length | 1410 | 279 |
| Vocab size | 1,98,570 | |
| Available in word2vecs | 47,154 | |
| # of documents | 18,991 | |

## 6. Experimental Results

**Baselines:** We compare our model's performance with three ROUGE variants, namely L, 1, and 2. In our experiments, we judge the quality of an evaluation method ($e$) by measuring the similarity with human ratings based on the *mean squared error (MSE)* between human scores and the automatic evaluation scores (human vs. ROUGE and human vs. CDSSM). We compare the human score with ROUGE's precision, recall, and F1-score. We use MSE to efficiently compute the absolute closeness of a score given by evaluation method ($e$). Before computing MSE we normalize CDSSM, ROUGE, and human scores for a fair comparison. In our model, we feed the summaries/original documents to summary/story subnets to obtain the scores, respectively.

**Datasets:** In this work, we use two datasets: *DUC-2001* consists of 60 sets of documents, 30 for training/30 for testing. Each set consists of 10 documents. Each document of a set contains a story for the same specific topic/event with human annotations for all peer summaries. The annotators assign various scores such as readability, coverage, etc. when presented a pair of summaries where the golden (human-written) summaries and the system-generated ones

Table 6: MSE results for DUC-2001

| MSE | | Gram | Chs | Org | Cov |
|---|---|---|---|---|---|
| Rouge-1 | R | 0.378 | 0.224 | 0.244 | 0.058 |
| | P | 0.344 | 0.204 | 0.220 | 0.064 |
| | F | 0.606 | 0.375 | 0.418 | **0.053** |
| Rouge-2 | R | 0.382 | 0.226 | 0.248 | 0.057 |
| | P | 0.348 | 0.206 | 0.224 | 0.063 |
| | F | 0.608 | 0.376 | 0.420 | 0.054 |
| Rouge-L | R | 0.385 | 0.227 | 0.249 | 0.056 |
| | P | 0.347 | 0.206 | 0.223 | 0.063 |
| | F | 0.608 | 0.376 | 0.420 | **0.053** |
| CDSSM | ML | 0.096 | 0.165 | 0.127 | 0.410 |
| | MLW | 0.071 | **0.142** | 0.114 | 0.414 |
| | SL | **0.042** | 0.150 | **0.059** | 0.472 |

are unknown to the annotators. Since the number of documents in DUC-2001 for the training purpose is very small, an additional 30,000 document-summary pairs from the CNN Dailymail dataset was used to train our models. This dataset is also from the news domain, so it is of almost similar document-summary sizes compared to DUC-2001. Due to resource constraints, we pruned the documents to 33% of their original size. We applied a text-rank algorithm (Mihalcea and Tarau, 2004) to get the top 33% sentences.

For our second dataset, we use the newly constructed dataset (Sum-PubMed) described in section (5.)

**Results and Analysis:** We have summarized the results in Table 6 for DUC-2001 (R, P and F are recall, precision and F1-score). The scores for peer summaries were provided on the scale of 0 to 4 (4 is the best). The scores are based on two aspects: *Readability and Coverage (Cov)*. The readability score consists of three scores: *Grammaticality (Gram), Cohesion (Chs), and Organization (Org)*. Grammaticality captures the overall grammar of the peer summary. Cohesion checks for the flow of information in the sentences of peer summaries. Also, organization concentrates on the high-level arrangement of ideas in the peer summary, while the coverage score measures the information covered by the peer summary of the document. We outperformed all ROUGE metrics in grammaticality, cohesion, and organization except the coverage because of document compression into lower-dimensional representations.

Table 7: MSE results for Sum-PubMed

| MSE | | Non-Re | Coh | Read | IOF |
|---|---|---|---|---|---|
| Rouge-1 | R | 0.144 | 0.122 | 0.120 | 0.088 |
| | P | 0.079 | 0.061 | 0.058 | **0.045** |
| | F | 0.114 | 0.094 | 0.092 | 0.067 |
| Rouge-2 | R | 0.334 | 0.299 | 0.294 | 0.242 |
| | P | 0.272 | 0.240 | 0.234 | 0.194 |
| | F | 0.311 | 0.277 | 0.272 | 0.223 |
| Rouge-L | R | 0.261 | 0.231 | 0.228 | 0.180 |
| | P | 0.211 | 0.183 | 0.177 | 0.146 |
| | F | 0.260 | 0.230 | 0.227 | 0.181 |
| CDSSM | ML1 | **0.048** | **0.050** | **0.051** | 0.074 |
| | ML2 | **0.048** | 0.053 | 0.057 | 0.077 |

For the Sum-Pubmed dataset, we evaluated our approach with two position-encoded multi-layered CNN models where the first model's (ML1) network size is similar to the model (1,4) used for DUC-2001 and the second model (ML2) has more layers (4,7) than ML1. Refer to Tables 2, 3 and 4 for model architecture details on single layer (SL for DUC 2001), the small multi-layer (MLW for DUC 2001, M1 for Sum-PubMed), and the large multi-layer (M2 for Sum-PubMed) CDSSMs used in our experiments. [6] Human annotations were conducted for 50 documents to find the correlation with the human ratings. 10 annotators were randomly assigned to pairs of summaries such that for each pair, we had 3 human ratings. Each annotator was asked to rate the summary pairs on a scale of 1 to 10 on the following four attributes: *Non-Repetition and no factual Redundancy (Non-Re)* where there should not to be any redundancy in the factual information and no repetition of sentences is allowed. *Coherence (coh)* means the arguments have to be connected rationally so that the reader/listener can observe consecutive sentences on one (or related) concept(s). *Readability (Read)* where criteria such as the spelling, correct grammar, understandability, etc. are measured. Lastly, *Informativeness, Overlap and Focus (IOF)* indicates the amount of information in one summary covered by the information in the other summary (using key-phrases/keywords). We summarized our results in the Table 7.

We outperform all ROUGE metrics in all categories except the IOF. We assume a similar reason to coverage's inferior results for IOF's poor performance. We discover that a single-layer model performs better for shorter documents like DUC-2001 since the *ML* models overfit the small amount of data. However, the multi-layer model outperforms *SL* on larger documents such as Sum-PubMed. In addition, position encoding helps in the overall evaluation due to the reasons mentioned in section (4.)

**Analysis and Discussion:** Our method performs better compared to ROUGE since it is a neural network model that learns to capture the semantic meaning of the inputs. In the Sum-PubMed dataset, the abstraction level in the golden summaries is much higher than CNN Dailymail. Hence, ROUGE fails to capture the semantic meaning of the inputs and thus performs poorly for scientific documents.

## 7. Conclusion

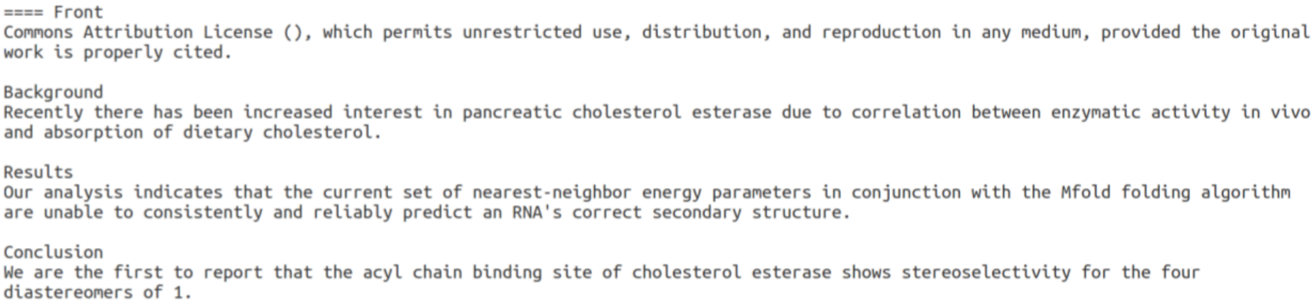
We argue that ROUGE does not align well with human scores for abstractive summarization, in particular, in the scientific domain. Thus, we propose CDSSMs as an evaluation metric. Our proposed models' evaluation scores align well with human scores due to their smaller MSEs compared to ROUGE. Specifically, our metric outperforms ROUGE on our newly constructed scientific dataset.

## 8. Acknowledgment

This project was supported by the Research-I foundation, CSE Department, IIT Kanpur [7] under master thesis project.

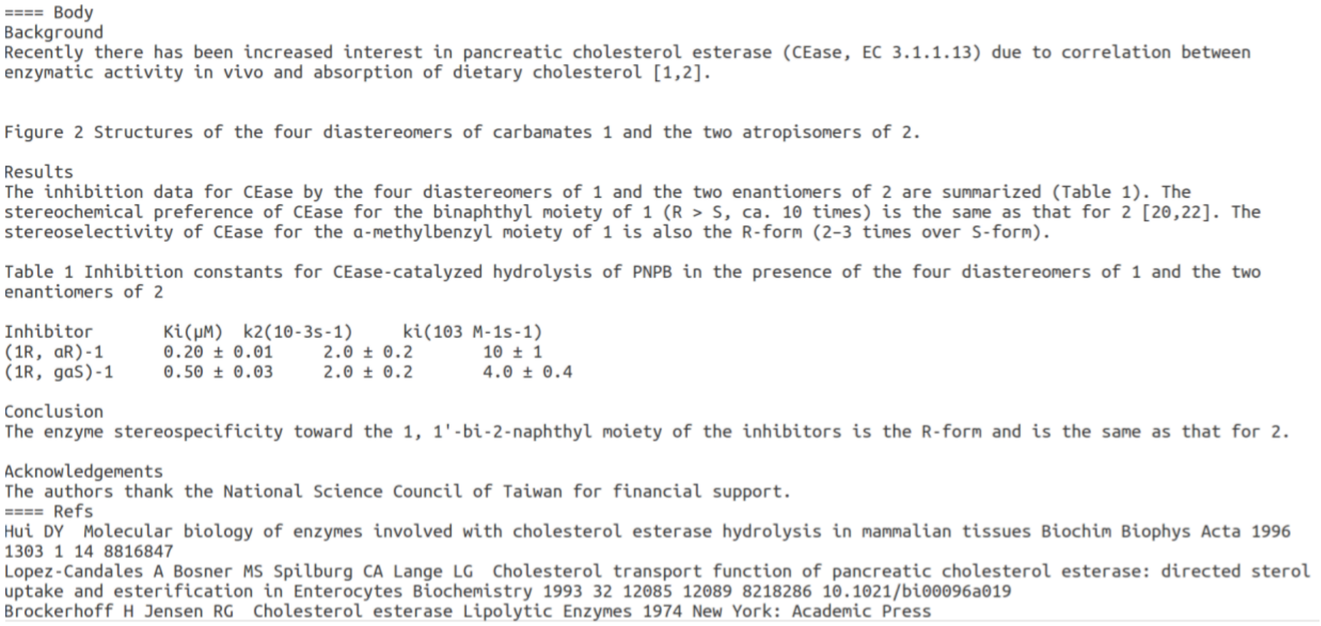
[6] Numbers in parenthesis are the number of filters and the corresponding filter sizes, followed by final embedding dimensions.

[7] https://www.cse.iitk.ac.in/users/rif/



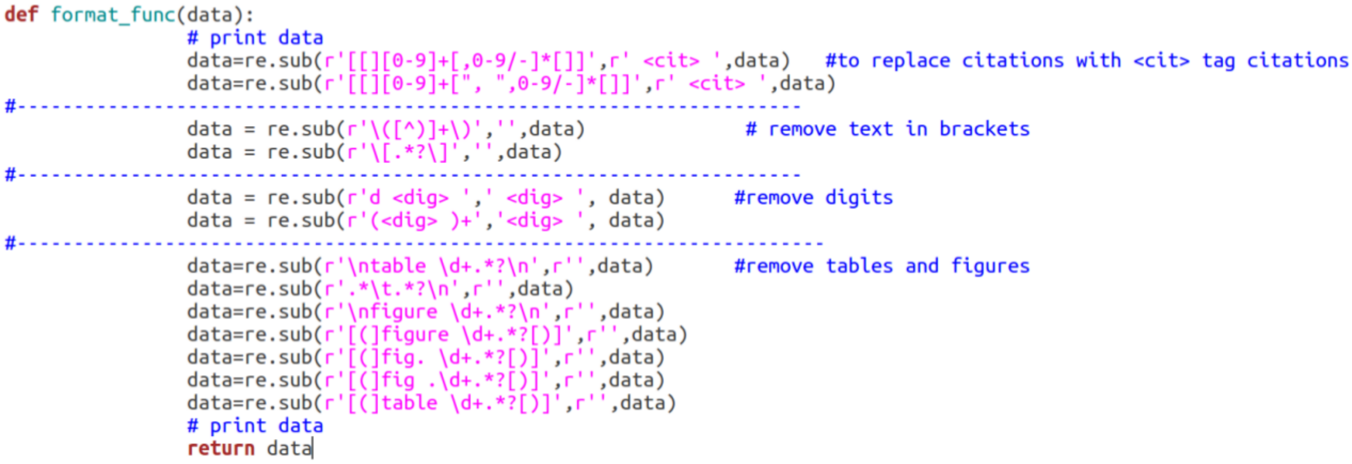
```
==== Front
Commons Attribution License (), which permits unrestricted use, distribution, and reproduction in any medium, provided the original
work is properly cited.

Background
Recently there has been increased interest in pancreatic cholesterol esterase due to correlation between enzymatic activity in vivo
and absorption of dietary cholesterol.

Results
Our analysis indicates that the current set of nearest-neighbor energy parameters in conjunction with the Mfold folding algorithm
are unable to consistently and reliably predict an RNA's correct secondary structure.

Conclusion
We are the first to report that the acyl chain binding site of cholesterol esterase shows stereoselectivity for the four
diastereomers of 1.
```

Figure 4: Front part of the example document test.txt



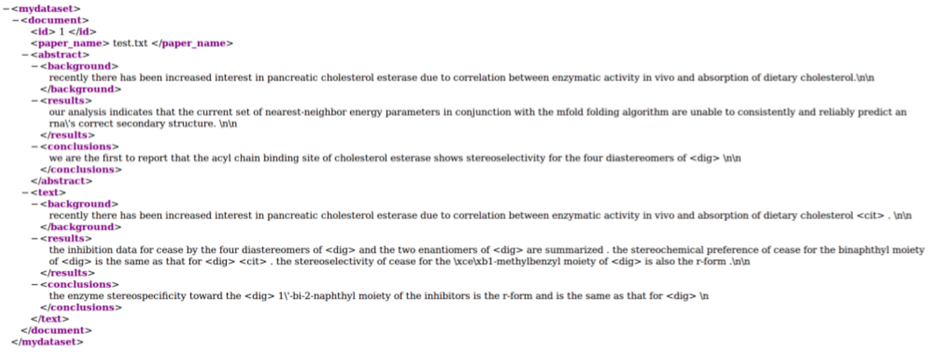
```
==== Body
Background
Recently there has been increased interest in pancreatic cholesterol esterase (CEase, EC 3.1.1.13) due to correlation between
enzymatic activity in vivo and absorption of dietary cholesterol [1,2].


Figure 2 Structures of the four diastereomers of carbamates 1 and the two atropisomers of 2.

Results
The inhibition data for CEase by the four diastereomers of 1 and the two enantiomers of 2 are summarized (Table 1). The
stereochemical preference of CEase for the binaphthyl moiety of 1 (R > S, ca. 10 times) is the same as that for 2 [20,22]. The
stereoselectivity of CEase for the α-methylbenzyl moiety of 1 is also the R-form (2-3 times over S-form).

Table 1 Inhibition constants for CEase-catalyzed hydrolysis of PNPB in the presence of the four diastereomers of 1 and the two
enantiomers of 2

Inhibitor       Ki(μM)  k2(10-3s-1)     ki(103 M-1s-1)
(1R, αR)-1      0.20 ± 0.01     2.0 ± 0.2       10 ± 1
(1R, gαS)-1     0.50 ± 0.03     2.0 ± 0.2       4.0 ± 0.4

Conclusion
The enzyme stereospecificity toward the 1, 1'-bi-2-naphthyl moiety of the inhibitors is the R-form and is the same as that for 2.

Acknowledgements
The authors thank the National Science Council of Taiwan for financial support.
==== Refs
Hui DY  Molecular biology of enzymes involved with cholesterol esterase hydrolysis in mammalian tissues Biochim Biophys Acta 1996
1303 1 14 8816847
Lopez-Candales A Bosner MS Spilburg CA Lange LG  Cholesterol transport function of pancreatic cholesterol esterase: directed sterol
uptake and esterification in Enterocytes Biochemistry 1993 32 12085 12089 8218286 10.1021/bi00096a019
Brockerhoff H Jensen RG  Cholesterol esterase Lipolytic Enzymes 1974 New York: Academic Press
```

Figure 5: Body part of the example document test.txt



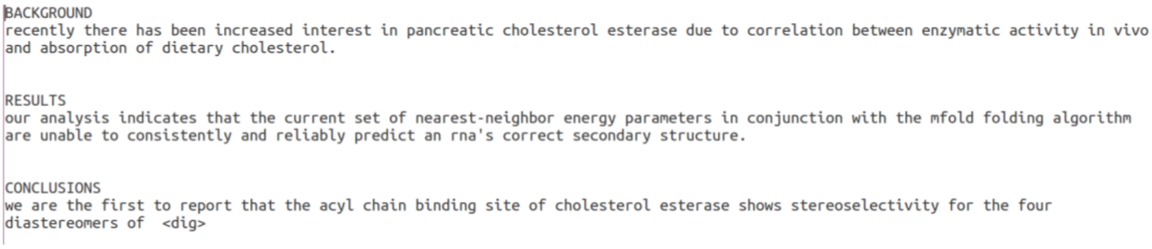
```
def format_func(data):
                # print data
                data=re.sub(r'[[][0-9]+[,0-9/-]*[]]',r' <cit> ',data)   #to replace citations with <cit> tag citations
                data=re.sub(r'[[][0-9]+[", ",0-9/-]*[]]',r' <cit> ',data)
#------------------------------------------------------------------
                data = re.sub(r'\([^)]+\)','',data)              # remove text in brackets
                data = re.sub(r'\[.*?\]','',data)
#------------------------------------------------------------------
                data = re.sub(r'd <dig> ',' <dig> ', data)      #remove digits
                data = re.sub(r'(<dig> )+','<dig> ', data)
#--------------------------------------------------------------------
                data=re.sub(r'\ntable \d+.*?\n',r'',data)       #remove tables and figures
                data=re.sub(r'.*\t.*?\n',r'',data)
                data=re.sub(r'\nfigure \d+.*?\n',r'',data)
                data=re.sub(r'[(]figure \d+.*?[)]',r'',data)
                data=re.sub(r'[(]fig. \d+.*?[)]',r'',data)
                data=re.sub(r'[(]fig .\d+.*?[)]',r'',data)
                data=re.sub(r'[(]table \d+.*?[)]',r'',data)
                # print data
                return data
```

Figure 6: The function which applies all the reg-ex expressions on the text string. These regex expressions clean the text



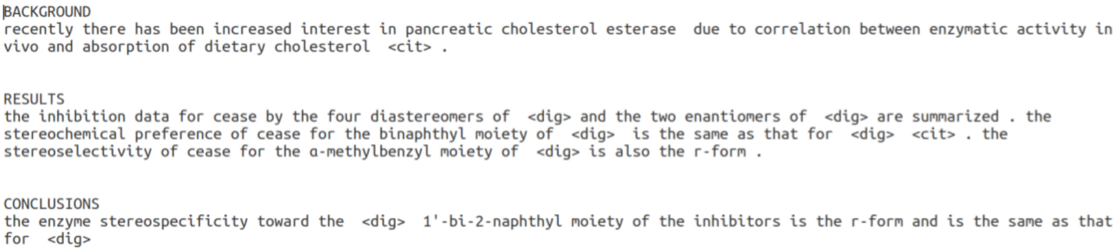
```
−<mydataset>
  −<document>
      <id> 1 </id>
      <paper_name> test.txt </paper_name>
    −<abstract>
      −<background>
          recently there has been increased interest in pancreatic cholesterol esterase due to correlation between enzymatic activity in vivo and absorption of dietary cholesterol.\n\n
        </background>
      −<results>
          our analysis indicates that the current set of nearest-neighbor energy parameters in conjunction with the mfold folding algorithm are unable to consistently and reliably predict an
          rna\'s correct secondary structure. \n\n
        </results>
      −<conclusions>
          we are the first to report that the acyl chain binding site of cholesterol esterase shows stereoselectivity for the four diastereomers of <dig> \n\n
        </conclusions>
      </abstract>
    −<text>
      −<background>
          recently there has been increased interest in pancreatic cholesterol esterase due to correlation between enzymatic activity in vivo and absorption of dietary cholesterol <cit> . \n\n
        </background>
      −<results>
          the inhibition data for cease by the four diastereomers of <dig> and the two enantiomers of <dig> are summarized . the stereochemical preference of cease for the binaphthyl moiety
          of <dig> is the same as that for <dig> <cit> . the stereoselectivity of cease for the \xce\xb1-methylbenzyl moiety of <dig> is also the r-form .\n\n
        </results>
      −<conclusions>
          the enzyme stereospecificity toward the <dig> 1\'-bi-2-naphthyl moiety of the inhibitors is the r-form and is the same as that for <dig> \n
        </conclusions>
      </text>
    </document>
  </mydataset>
```

Figure 7: XML file: text.xml



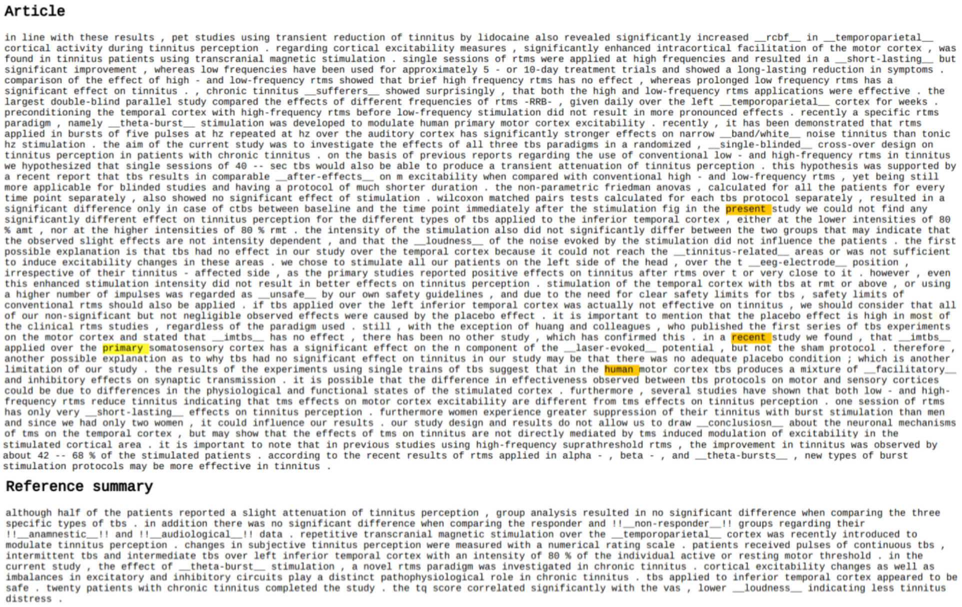
```
BACKGROUND
recently there has been increased interest in pancreatic cholesterol esterase due to correlation between enzymatic activity in vivo
and absorption of dietary cholesterol.


RESULTS
our analysis indicates that the current set of nearest-neighbor energy parameters in conjunction with the mfold folding algorithm
are unable to consistently and reliably predict an rna's correct secondary structure.


CONCLUSIONS
we are the first to report that the acyl chain binding site of cholesterol esterase shows stereoselectivity for the four
diastereomers of  <dig>
```

Figure 8: The preprocessed abstract file: abstract 1.txt



```
BACKGROUND
recently there has been increased interest in pancreatic cholesterol esterase  due to correlation between enzymatic activity in
vivo and absorption of dietary cholesterol  <cit> .


RESULTS
the inhibition data for cease by the four diastereomers of  <dig> and the two enantiomers of  <dig> are summarized . the
stereochemical preference of cease for the binaphthyl moiety of  <dig>  is the same as that for  <dig>  <cit> . the
stereoselectivity of cease for the α-methylbenzyl moiety of  <dig> is also the r-form .


CONCLUSIONS
the enzyme stereospecificity toward the  <dig>  1'-bi-2-naphthyl moiety of the inhibitors is the r-form and is the same as that
for  <dig>
```

Figure 9: The preprocessed text file: text 1.txt



```
Article

in line with these results , pet studies using transient reduction of tinnitus by lidocaine also revealed significantly increased __rcbf__ in __temporoparietal__
cortical activity during tinnitus perception . regarding cortical excitability measures , significantly enhanced intracortical facilitation of the motor cortex , was
found in tinnitus patients using transcranial magnetic stimulation . single sessions of rtms were applied at high frequencies and resulted in a __short-lasting__ but
significant improvement , whereas low frequencies have been used for approximately 5 - or 10-day treatment trials and showed a long-lasting reduction in symptoms .
comparison of the effect of high - and low-frequency rtms showed that brief high frequency rtms has no effect , whereas prolonged low frequency rtms has a
significant effect on tinnitus . , chronic tinnitus __sufferers__ showed surprisingly , that both the high and low-frequency rtms applications were effective . the
largest double-blind parallel study compared the effects of different frequencies of rtms -RRB- , given daily over the left __temporoparietal__ cortex for weeks .
preconditioning the temporal cortex with high-frequency rtms before low-frequency stimulation did not result in more pronounced effects . recently a specific rtms
paradigm , namely __theta-burst__ stimulation was developed to modulate human primary motor cortex excitability . recently , it has been demonstrated that rtms
applied in bursts of five pulses at hz repeated at hz over the auditory cortex has significantly stronger effects on narrow __band/white__ noise tinnitus than tonic
hz stimulation . the aim of the current study was to investigate the effects of all three tbs paradigms in a randomized , __single-blinded__ cross-over design on
tinnitus perception in patients with chronic tinnitus . on the basis of previous reports regarding the use of conventional low - and high-frequency rtms in tinnitus
we hypothesized that single sessions of 40 -- sec tbs would also be able to produce a transient attenuation of tinnitus perception . this hypothesis was supported by
a recent report that tbs results in comparable __after-effects__ on m excitability when compared with conventional high - and low-frequency rtms , yet being still
more applicable for blinded studies and having a protocol of much shorter duration . the non-parametric friedman anovas , calculated for all the patients for every
time point separately , also showed no significant effect of stimulation . wilcoxon matched pairs tests calculated for each tbs protocol separately , resulted in a
significant difference only in case of ctbs between baseline and the time point immediately after the stimulation fig in the present study we could not find any
significantly different effect on tinnitus perception for the different types of tbs applied to the inferior temporal cortex , either at the lower intensities of 80
% amt , nor at the higher intensities of 80 % rmt . the intensity of the stimulation also did not significantly differ between the two groups that may indicate that
the observed slight effects are not intensity dependent , and that the __loudness__ of the noise evoked by the stimulation did not influence the patients . the first
possible explanation is that tbs had no effect in our study over the temporal cortex because it could not reach the __tinnitus-related__ areas or was not sufficient
to induce excitability changes in these areas . we chose to stimulate all our patients on the left side of the head , over the t __eeg-electrode__ position ,
irrespective of their tinnitus - affected side , as the primary studies reported positive effects on tinnitus after rtms over t or very close to it . however , even
this enhanced stimulation intensity did not result in better effects on tinnitus perception . stimulation of the temporal cortex with tbs at rmt or above , or using
a higher number of impulses was regarded as __unsafe__ by our own safety guidelines , and due to the need for clear safety limits for tbs , safety limits of
conventional rtms should also be applied . if tbs applied over the left inferior temporal cortex was actually not effective on tinnitus , we should consider that all
of our non-significant but not negligible observed effects were caused by the placebo effect . it is important to mention that the placebo effect is high in most of
the clinical rtms studies , regardless of the paradigm used . still , with the exception of huang and colleagues , who published the first series of tbs experiments
on the motor cortex and stated that __imtbs__ has no effect , there has been no other study , which has confirmed this . in a recent study we found , that __imtbs__
applied over the primary somatosensory cortex has a significant effect on the n component of the __laser-evoked__ potential , but not the sham protocol . therefore ,
another possible explanation as to why tbs had no significant effect on tinnitus in our study may be that there was no adequate placebo condition ; which is another
limitation of our study . the results of the experiments using single trains of tbs suggest that in the human motor cortex tbs produces a mixture of __facilitatory__
and inhibitory effects on synaptic transmission . it is possible that the difference in effectiveness observed between tbs protocols on motor and sensory cortices
could be due to differences in the physiological and functional states of the stimulated cortex . furthermore , several studies have shown that both low - and high-
frequency rtms reduce tinnitus indicating that tms effects on motor cortex excitability are different from tms effects on tinnitus perception . one session of rtms
has only very __short-lasting__ effects on tinnitus perception . furthermore women experience greater suppression of their tinnitus with burst stimulation than men
and since we had only two women , it could influence our results . our study design and results do not allow us to draw __conclusiosn__ about the neuronal mechanisms
of tms on the temporal cortex , but may show that the effects of tms on tinnitus are not directly mediated by tms induced modulation of excitability in the
stimulated cortical area . it is important to note that in previous studies using high-frequency suprathreshold rtms , the improvement in tinnitus was observed by
about 42 -- 68 % of the stimulated patients . according to the recent results of rtms applied in alpha - , beta - , and __theta-bursts__ , new types of burst
stimulation protocols may be more effective in tinnitus .

Reference summary

although half of the patients reported a slight attenuation of tinnitus perception , group analysis resulted in no significant difference when comparing the three
specific types of tbs . in addition there was no significant difference when comparing the responder and !!__non-responder__!! groups regarding their
!!__anamnestic__!! and !!__audiological__!! data . repetitive transcranial magnetic stimulation over the __temporoparietal__ cortex was recently introduced to
modulate tinnitus perception . changes in subjective tinnitus perception were measured with a numerical rating scale . patients received pulses of continuous tbs ,
intermittent tbs and intermediate tbs over left inferior temporal cortex with an intensity of 80 % of the individual active or resting motor threshold . in the
current study , the effect of __theta-burst__ stimulation , a novel rtms paradigm was investigated in chronic tinnitus . cortical excitability changes as well as
imbalances in excitatory and inhibitory circuits play a distinct pathophysiological role in chronic tinnitus . tbs applied to inferior temporal cortex appeared to be
safe . twenty patients with chronic tinnitus completed the study . the tq score correlated significantly with the vas , lower __loudness__ indicating less tinnitus
distress .
```

Figure 10: Sum-PubMed dataset example showing the article and the summary

## 9. References

Cai, S. and Knight, K. (2013). Smatch: an evaluation metric for semantic feature structures. In *ACL*.

Cohan, A. and Goharian, N. (2016). Revisiting summarization evaluation for scientific articles. *CoRR*, abs/1604.00400.

Cohan, A., Dernoncourt, F., Kim, D. S., Bui, T., Kim, S., Chang, W., and Goharian, N. (2018). A discourse-aware attention model for abstractive summarization of long documents. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, volume 2, pages 615–621.

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., and Kuksa, P. (2011). Natural language processing (almost) from scratch. *J. Mach. Learn. Res.*, 12:2493–2537, November.

Dohare, S., Karnick, H., and Gupta, V. (2017). Text summarization using abstract meaning representation. *arXiv preprint arXiv:1706.01678*.

Fang, H., Gupta, S., Iandola, F., Srivastava, R. K., Deng, L., Dollár, P., Gao, J., He, X., Mitchell, M., Platt, J. C., et al. (2015). From captions to visual concepts and back. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1473–1482.

Gao, J., He, X., Yih, W.-t., and Deng, L. (2014a). Learning continuous phrase representations for translation modeling. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 699–709.

Gao, J., Pantel, P., Gamon, M., He, X., and Deng, L. (2014b). Modeling interestingness with deep neural networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2–13.

Gao, J. (2017). Introduction to deep learning for natural language processing (tutorial at deeplearning2017 summer school in bilbao). July.

Genest, P.-E., Gotti, F., and Bengio, Y. (2011). Deep learning for automatic summary scoring.

Huang, P.-S., He, X., Gao, J., Deng, L., Acero, A., and Heck, L. P. (2013). Learning deep structured semantic models for web search using clickthrough data. In *CIKM*.

Jiao, X., Wang, F., and Feng, D. (2018). Convolutional neural network for universal sentence embeddings. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2470–2481. Association for Computational Linguistics.

Kalchbrenner, N., Grefenstette, E., and Blunsom, P. (2014). A convolutional neural network for modelling sentences. 1, 04.

Kim, Y. (2014). Convolutional neural networks for sentence classification. In *EMNLP*.

Lin, C.-Y. (2004). Rouge: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*.

Lloret, E., Plaza, L., and Aker, A. (2018). The challenging task of summary evaluation: an overview. *Language Resources and Evaluation*, 52(1):101–148.

Louis, A. and Nenkova, A. (2013). Automatically assessing machine summary content without a gold standard. *Computational Linguistics*, 39(2):267–300.

Mihalcea, R. and Tarau, P. (2004). Textrank: Bringing order into texts.

Mikolov, T., Karafiát, M., Burget, L., Černockỳ, J., and Khudanpur, S. (2010). Recurrent neural network based language model. In *Eleventh Annual Conference of the International Speech Communication Association*.

Nenkova, A., Passonneau, R., and McKeown, K. (2007). The pyramid method: Incorporating human content selection variation in summarization evaluation. *ACM Trans. Speech Lang. Process.*, 4(2), May.

Ng, J.-P. and Abrecht, V. (2015). Better summarization evaluation with word embeddings for rouge. *arXiv preprint arXiv:1508.06034*.

Palangi, H., Deng, L., Shen, Y., Gao, J., He, X., Chen, J., Song, X., and Ward, R. (2016). Deep sentence embedding using long short-term memory networks: Analysis and application to information retrieval. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, 24(4):694–707.

Peyrard, M. and Eckle-Kohler, J. (2017). A principled framework for evaluating summarizers: Comparing models of summary quality against human judgments. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 26–31.

Peyrard, M. and Gurevych, I. (2018). Objective function learning to match human judgements for optimization-based summarization. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, volume 2, pages 654–660.

Peyrard, M., Botschen, T., and Gurevych, I. (2017). Learning to score system summaries for better content selection evaluation. In *Proceedings of the Workshop on New Frontiers in Summarization*, pages 74–84.

Pilehvar, M. T. and Collier, N. (2016). Improved semantic representation for domain-specific entities. In *Proceedings of the 15th Workshop on Biomedical Natural Language Processing*, pages 12–16.

Ruseti, S., Dascalu, M., Johnson, A. M., McNamara, D. S., Balyan, R., McCarthy, K. S., and Trausan-Matu, S. (2018). Scoring summaries using recurrent neural networks. In *International Conference on Intelligent Tutoring Systems*, pages 191–201. Springer.

ShafieiBavani, E., Ebrahimi, M., Wong, R., and Chen, F. (2017). A semantically motivated approach to compute rouge scores. *arXiv preprint arXiv:1710.07441*.

ShafieiBavani, E., Ebrahimi, M., Wong, R., and Chen, F. (2018a). A graph-theoretic summary evaluation for rouge. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 762–767. Association for Computational Linguistics.

ShafieiBavani, E., Ebrahimi, M., Wong, R., and Chen, F.

(2018b). Summarization evaluation in the absence of human model summaries using the compositionality of word embeddings. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 905–914.

Shen, Y., He, X., Gao, J., Deng, L., and Mesnil, G. (2014). A latent semantic model with convolutional-pooling structure for information retrieval. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 101–110. ACM.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15(1):1929–1958, January.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. *CoRR*, abs/1706.03762.

Yih, W.-t., Chang, M.-W., He, X., and Gao, J. (2015). Semantic parsing via staged query graph generation: Question answering with knowledge base. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1321–1331. Association for Computational Linguistics.

Zhang, Y. and Wallace, B. C. (2017). A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification. In *IJCNLP*.

Zhang, R., Lee, H., and Radev, D. R. (2016). Dependency sensitive convolutional neural networks for modeling sentences and documents. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1512–1521.