NEWSFARM: the Largest Chinese Corpus for Long News Summarization

Anonymous ACL submission

Abstract

Recently, driven by a large number of datasets, the field of natural language processing(NLP) has developed rapidly. However, the lack of large-scale and high-quality Chinese datasets is still a critical bottleneck for further research on automatic text summarization. To close this gap, we searched Chinese news websites of domestic and abroad media, designed the algorithm ***HSS***(hidden topic, semantic, and syntactic) to crawl and filter these records to construct ***NEWSFARM***. ***NEWSFARM*** is the largest highest quality Chinese long news summarization corpus, containing more than 200K Chinese long news and summaries written by professional editors or authors, which are all released to the public. Based on the corpus, we calculated the static metrics and designed many experiments with the baseline models. By comparing with the common datasets, the experiment results show that the high quality of our dataset and training effect of the models, which not only demonstrates the usefulness and challenges of the proposed corpus for automatic text summarization but also provides a benchmark for further research.

1 Introduction

Automatic text summarization is one of the central problems in Natural Language Processing(NLP), posing two aspects challenges mainly about understanding and generation. After years of deep learning development, the quality of models has significantly improved, especially in some data-driven models, such as sequence-to-sequence architecture(Nallapati et al., 2016b; Rush et al., 2015; See et al., 2017a), transfomer(Vaswani et al., 2017), bert(Devlin et al., 2018), bart(Lewis et al., 2019), GPT-3(Brown et al., 2020), Presumm(Liu and Lapata, 2019), MatchSum(Zhong et al., 2020), PanGu(Zeng et al., 2021), etc. However, the lack of large-scale and high-quality Chinese datasets for model training leads to the superiority of these models cannot being fully demonstrated, which greatly limits the further development of Chinese automatic summarization. In a sense, pre-training models, which have become popular in recent years, are designed to overcome the lack of specific datasets for specific NLP tasks(Xu et al., 2021). Numerous NLP tasks were unified into one type of task, the existing datasets were used to train the model, and then the pre-training model was transferred to specific tasks by fine-tuning. The fine-tuning process also requires a large number of task-specific datasets. In the automatic summarization task, the annotated data consists of the summary and source text. The summary serves as the label of the data. On the premise of the same summary quality, the length of the source text can improve the quality of the trained model to some extent. Therefore, the longer the text, the harder it is to get a good summary.

Current dataset work has made some progress, but there are the following problems. Insufficient amount of data in the dataset. Data cleaning algorithms are too simple to get high-quality data. The effect of the models can not be fully demonstrated with these data.

To tackle these problems, we checked Chinese news websites of domestic and abroad media. More than 200K Chinese long news <article,summary> pair were crawled. Afterward, the crawled data was cleaned with ***HSS*** and a high-quality dataset was obtained.

The contributions are as follows: (1)***NEWSFARM*** is the largest highest quality Chinese corpus for long news summarization, up to now. To a certain extent, it makes up for the lack of Chinese datasets in the field of automatic summarization. (2)We design a comprehensive data filtering algorithm ***HSS*** based on hidden topics, semantic similarity, and syntactic similarity, which can help improve the quality of datasets. (3)The whole dataset is divided into

three parts: training, verification, and test set. We compare the static metrics of common datasets, the superiority of ***NEWSFARM*** is demonstrated with these detailed metrics. (4)We designed many experiments based on the baseline models and calculated all kinds of metrics' scores. The results of the evaluation proved the utility and challenges of this dataset.

2 Related work

2.1 Common datesets

According to the advances in research, the English summarization datasets are superior to the Chinese in both quality and quantity.

English summarization datasets such as short summaries Gigaword(Napoles et al., 2012), Newsroom(Grusky et al., 2018). Long text CNN/DM(Hermann et al., 2015; Nallapati et al., 2017; See et al., 2017b), multi-document DUC2004(Harman and Over, 2004), TAC2011(Giannakopoulos et al., 2011), MultiNews(Fabbri et al., 2019), WCEP(Ghalandari et al., 2020), dialogue summarization corpus(Gliwa et al., 2019), patent documents(Sharma et al., 2019), scientifific papers(Cohan et al., 2018; Yasunaga et al., 2019), bills(Kornilova and Eidelman, 2019), Crowdsourcing(Falke and Gurevych, 2017), besides there are XSUM(Narayan et al., 2018), MDSWriter(Meyer et al., 2016), TALKSUMM(Lev et al., 2019), etc.

Chinese summarization datasets have the large-scale Chinese short text summarization dataset LCSTS(Hu et al., 2015), the Chinese long text extractive summarization dataset CLES(Chen et al., 2021), the sports game summarization dataset SPORTSSUM(Huang et al., 2020), the long Chinese summarization of police inquiry record dataset LCSPIRT(Xi et al., 2020), and a Chinese e-commerce product summarization dataset(Yuan et al., 2020).

Minority language summarization dataset only have INDOSUM(Kurniawan and Louvan, 2018).

The neural Cross-Lingual summarization dataset have NCLS(Zhu et al., 2019).

2.2 Construction method and research

At present, there are mainly four methods to construct text summarization datasets:

(1)Find the appropriate text source, directly crawl these records and clean it up. The CNN/DM(Hermann et al., 2015) directly collect 93K articles from the CNN and 220K articles from the Daily Mail websites with summaries. The Gigaword(Napoles et al., 2012) corpus collect 9.5 million news articles from the New York Times. The BigPatent(Sharma et al., 2019) through BigQuery to obtain 1.3 million US patent documents and abstract summaries of human writing. The Billsum(Kornilova and Eidelman, 2019) comes from two sources: the US bill and the California bill. The INDOSUMU(Kurniawan and Louvan, 2018) use a dataset provided by Shortir, an Indonesian news aggregator and summarizer company. Multi-news(Fabbri et al., 2019) is composed of news articles and artificial summaries of those articles from Newser.com. Newsroom(Grusky et al., 2018) uses social media and search engine metadata to collect short news and summaries. XSUM(Narayan et al., 2018) is built by collecting BBC articles and accompanying one-sentence summaries. CLES(Chen et al., 2021) extract the <article,summary> pairs from the Chinese Sina Weibo. Besides, using existing datasets of conversation documents and create similar datasets by linguists(Cohan et al., 2018), using the 1000 most cited papers from the American Civil Liberties Union Anatology Network(AAN)(Radev et al., 2009), and their citation information to create dataset(Yasunaga et al., 2019).

(2) Find some data sources, select some of them as seeds, and then get more data through some processing. The LCSTS(Hu et al., 2015) collects 50 very popular organization users as seeds, capturing the aggregator followed by these seed users, and using manually written rules to filter them, then use selected users and text crawlers to capture their micro-blogs.

(3)Crowdsourcing method, using the internet in the form of questions to obtain the dataset. An improved crowdsourcing approach is used to build the dataset(Falke and Gurevych, 2017).

(4)Some special ways. Following the NLP and ML conferences, 1,716 video interviews from the ACLU, ACLU, EMNLP, Sigdal, and ICML were analyzed, the videos were downloaded, and voice data were extracted to construct dataset(Lev et al., 2019). The MDSwriter system proposed by the writer to construct the dataset(Meyer et al., 2016).

3 Dataset

NEWSFARM construction processes for this paper is as follows:

Step1: Target selection.
Step2: Crawl data.
Step3: Data filtering.
Step4: Forming the final dataset.
The specific processes is shown in Figure 1.

3.1 Data collection

To build a high-quality dataset, we must choose some high-quality data source that contains artificial summaries, which should cover news from various fields. We checked news websites of domestic and abroad media. After extensive screening, we select some websites that meet the requirements, such as the United Nations News Network, China Daily website, etc. Designed a crawler program(Zhang et al., 2018), which can filter out the noise in the page, such as advertisements, pictures, etc, to automatically extract effective information from the websites. After simple processing, the preliminary dataset is obtained.

3.2 Filter algorithm

The preliminary dataset is filtered in two aspects: format and content.

3.2.1 Format filter

We define short summaries as those with less than 45 words and short news as those with less than 600 words.

Step1: Error format filtering. Traverse all the collected records. If a summary or news body is missing from the record pair, delete this record.

Step2: Short summaries filtering. Traverse all the collected summaries records. If the summary words are less than 45, delete the record pair.

Step3: Short news filtering. Traverse all the collected news body records. If the news words are less than 600, delete the record pair.

After format filtering, the average length of summary in our dataset is 87 words and the average length of an article is 1048 words.

3.2.2 Content filter

To further improve the quality of the dataset, we need to strictly detect the matching degree between the summary and the article, and filter out those records that do not match between the summary and article.

There are many ways to calculate text similarity, some are based on the theme of the texts, some are based on the semantics of the texts, and some are based on the structure of the texts. In the task of text summarization, the summary is a short text and the original is a long text, which brings great difficulty to the calculation of text similarity. To calculate text similarity from any single point of topics, semantics, and structure will miss information.

On this issue, we propose a comprehensive algorithm ***HSS*** to calculate the text similarity of different lengths based on the hidden text topic(Gong et al., 2018) and the short text similarity with semantic and syntactic information(Yang et al., 2021). The hidden text topic ignores the impact of word ambiguity and the semantic information contained in the structure of the text. The short text similarity with semantic and syntactic information(Yang et al., 2021) complements the former approach. We compare the ***HSS*** score with the threshold value and filter out the records below the threshold. Here we only describe the idea of this method. Please refer to Appendix A for specific algorithms.

(1)The hidden text topic:

Step1: The preprocessing module tokenizes articles and removes stop words and prepositions.

Step2: Topic generation from articles, the word vectors in a article are $W = \{w_1, w_2, ..., w_n\}$, and the hidden topic vectors of the article are $H = \{h_1, h_2, ..., h_n\}$. We define the reconstructed word vector $\widetilde{w}_i$ for the word w_i as the optimal linear approximation given by topic vectors. The goal is to find the optimal H^* so as to minimize the reconstruction error E for the whole article. The detailed introduction is in Appendix A.

Step3: Topic mapping to summary. We have extracted K topic vectors $\{h_k^*\}_k^{k=1}$from the article matrix W. Suppose the vectors of the words in a summary are $S = \{s_1, s_2, ..., s_m\}$.

Let $\widetilde{s}_j^k$ be the reconstruction of the summary word s_j given by one topic h_k^*. The $r(h_k^*, s_j)$ is the relevance between a topic vector h_k^* and summary word s_j. It is defined as the cosine similarity between $\widetilde{s}_j^k$ and s_j

Therefore, the $topic\_sim = r(W, S)$ as the relevance between the article W and the summary S, and r(W,S) is the sum of topic-summary relevance weighted by the importance of the topic. The detailed introduction is in Appendix A.

Step4: The higher $r(W, S)$(topic_sim) is, the better the summary matches the article.

(2)The text similarity with semantic and syntactic information:

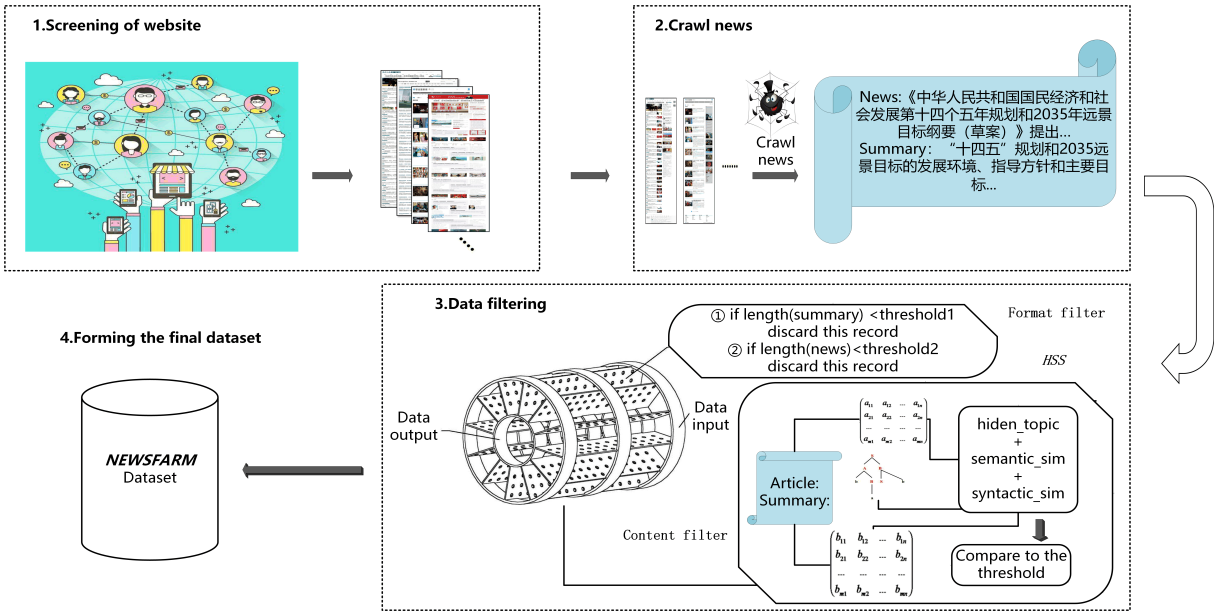

Figure 1: ***NEWSFARM*** construction processes

Measuring the text similarity based on semantic.

Definition 1(Term). A term t is a word or a multiword expression(MWE).

Definition 2(Instance). An instance e is a concrete object.

Definition 3(Concept). A concept c is defined as a general and abstract description of a set of instances.

a.Semantic similarity between terms:

The term similarity calculation can be roughly divided into the corpus based method and knowledge-based method. Both types of methods have shortcomings, so we combine the two methods to get a better solution.

First, in the corpus based method, we use semantic composition to obtain semantic vectors of terms. Given a term t and the semantic vector of each word in t, the semantic vector of t can be calculated by $V_t = \sum_{k=1}^{K} V_{w_k} * SIF(w_k)$. Where K stands for the number of words, SIF is the smooth inverse frequency based on the attention mechanism. The similarity of terms is computed using the two semantic vectors $R_c(t_1, t_2) = (V_{t_1} * V_{t_2})/(|V_{t_1}| * |V_{t_2}|)$ The detailed introduction is in Appendix A.

Second, the similarity of terms based on knowledge is obtained by Probase(Wu et al., 2012) and calculated based on the two concept vectors. $R_k(t_1, t_2) = (I_{t_1} * I_{t_2})/(|I_{t_1}| * |I_{t_2}|)$. The large-scale knowledge base Probase(Wu et al., 2012), which is a probabilistic semantic network that contains millions of concepts and instances. The I_{t_1} and I_{t_2} is the concept vector. The detailed introduction is in Appendix A.

Finally, a linear method is adopted to fuse R_c and R_k, $R = \alpha * R_k + (1 - \alpha) * R_c$. Where α is a tuning parameter. Since R_c plays a subordinate role in term similarity, α should be a value greater than 0.5. The detailed introduction is in Appendix A.

b.Semantic similarity of texts:

Step1: Text segmentation. We split the text into set of terms.

Step2: Part of speech judgment. For the terms set after segmentation, we need to determine the POS of each term in the current context. Stanford CoreNLP is used to determine the POS of each term. We use the method in(Li et al., 2017) to further distinguish the type of terms(concept or instance). The detailed introduction is in Appendix A.

Step3: Conceptualization of term. With the help of Probase(Wu et al., 2012), we can easily obtain concepts of instances. However, in natural language, instances are often ambiguous, it has at least two completely unrelated concepts. For ambiguous terms, we need to select appropriate contextual terms to eliminate ambiguity. Based on the context selection and assigned weights, the concept with the maximum score is considered the meaning of the target word in the current context. The detailed introduction is in Appendix A.

Step4: Semantic vector of texts. After the above

three parts, we have constructed the semantic vector. Using the semantic vector of the article and summary, the similarity score of the <article,summary> pair is obtained by using the similarity calculation formula(semantic_sim). The detailed introduction is in Appendix A.

c.Syntactic similarity of texts:

The above semantic similarity calculation method of texts is simple and effective, but it ignores the impact of syntactic information. We compute the syntactic similarity of texts(syntactic_sim) based on a constituency parse tree(CPT). Here, we use term as the minimum semantic unit to construct the CPT of texts. The detailed introduction is in Appendix A.

d.Overall text similarity based on the semantic and syntactic information:

A linear method is adopted to fuse the semantic and syntactic information. $sim(article, summary) = \varphi * semantic\_sim + (1 - \varphi) * syntactic\_sim$, where φ is a tuning parameter. Since $semantic\_sim$ plays a subordinate role in text similarity, φ should be a value greater than 0.5. The detailed introduction is in Appendix A.

(3)Overall<article,summary>pair similarity

The linear method is adopted to fuse the topic, the semantic and syntactic information. $sim(article, summary) = (1 - \theta) * [\varphi * semantic\_sim + (1 - \varphi) * syntactic\_sim] + \theta * hiden\_topic$, where θ is a tuning parameter. Since $hiden\_topic$ plays a subordinate role in text similarity, θ should be a value greater than 0.5. The detailed introduction is in Appendix A.

We compare the sim(article,summary) with the threshold value, and filter out the records below the threshold. Afterward, high-quality data was obtained for our dataset.

3.3 Build the dataset

After data collection and data filtering, we collect these records and finally construct the ***NEWSFARM***. An example of the dataset is shown in Appendix B, Figure 1.

4 Data analysis

4.1 Data statistics

The ***NEWSFARM*** contains a total of 200K pieces of data, each including a summary and a news story. We count the size of the corpus, the size of training, validation, and test set, the average document (source), and summary (target) length (in terms of words and sentences). We compare it with the common datasets, as detailed in Table 2.

According to the content of 2.1, the Chinese summarization datasets have CLES(Chen et al., 2021), LCSTS(Hu et al., 2015), SPORTSSUM(Huang et al., 2020), LCSPIRT(Xi et al., 2020), and a Chinese e-commerce product summarization dataset(Yuan et al., 2020). The LCSTS is a short text summarization dataset and is not comparable to our dataset. Other Chinese summarization datasets are quite different from ours in both quality and quantity, so we chose the best of them to compare. By comparing with CLES(Chen et al., 2021), it can be seen that the scale of our dataset is larger than CLES(Chen et al., 2021), and it has advantages in the number of sentences. Compared with CNN/DM(Hermann et al., 2015; Nallapati et al., 2017; See et al., 2017b), our dataset has obvious advantages in terms of document length and summary length. In addition, our dataset covers a wider range of fields than CLES(Chen et al., 2021), including the world's politics, economy, culture, tourism, and other aspects. These metrics fully demonstrate the superiority of our dataset.

4.2 Bound

LEAD-3:A common automatic summarization strategy of online publications is to copy the first sentence, first paragraph, or first K words of the text and treat these as the summary. According to the prior work(Mihalcea and Tarau, 2004), we use the LEAD-3 baseline, in which the first three sentences of the text are returned as the summary. This part makes LEAD-3 can be competitive with some state-of-art systems.

ORACLE:Given an article text $A =< a_1, a_1, ...a_n >$ consisting of a sequence of tokens a_i and the corresponding article summary $S =< s_1, s_1, ...s_m >$ consisting of tokens s_i, the set of extractive fragments F(A,S)(Grusky et al., 2018)is the set of shared sequences of tokens in A and S. We identify these extractive fragments of an article-summary pair using a greedy process. Oracle represents best possible performance of an ideal extractive system.

We selected three different metrics(Lin and Hovy, 2003), namely ROUGE-1 which measures the overlap of unigrams, ROUGE-2 which measures the overlap of bi-grams, and ROUGE-L

Datasets	#docs(total/train/val/test)	avg.document length		avg.summary length	
		words	sentences	words	sentences
NEWSFARM	**206,480/185,125/18,123/21,232**	1,048.00	**39.29**	86.90	**3.05**
CLES	103,893/95,000/3,839/5,000	1,584	36.00	106.00	3.00
LCSTS	2,412,163/2,400,391/10,666/1,106	108.80	10.13	19	1.00
CNN/DM	312,085/287,227/13,368/11,490	687.09	31.66	48.49	3.73

Table 2: Comparison of summarization datasets with respect to overall corpus size, size of training, validation, and test set, average document and summary length.(Statistical length is divided into sentence level and word level)

which measures the longest common subsequence. Use the two baseline models introduced above to compare the scores corresponding to the three metrics on different datasets. If summaries generated by the above model achieve a high ROUGE score, it means that the dataset has a low level of abstraction. The comparison results of ***NEWSFARM*** and other datasets are shown in Table 3.

The LCSTS(Hu et al., 2015) is a short text summarization dataset and is not comparable to our dataset. By comparison with CLES(Chen et al., 2021), it can be seen that all metrics of our dataset in n-gram are higher than that of CLES. All the rouge scores in LEAD-3 and ORACLE are lower than CLES. These data demonstrate that the quality of our dataset is superior to the CLES, especially in terms of abstraction. Compared to the English summarization dataset CNN/DM(Hermann et al., 2015; Nallapati et al., 2017; See et al., 2017b), our dataset is slightly inadequate, but not by much.

5 Experiment

In this part, we prove the quality of the dataset from three aspects:

(1)Automatic evaluation. Demonstrate the high quality of our dataset by demonstrating the quality of the models trained by our dataset.

(2)Human evaluation. Demonstrate the high quality of our dataset by showing the scores of the human evaluation metrics.

(3)Experiment with out-of-domain data. Demonstrate the high quality of our dataset by demonstrating the actual effects of the model tested with additional data.

First, we select twelve existing baseline models which are frequently used, obtain the ROUGE scores of these models on CNN/DM(Hermann et al., 2015; Nallapati et al., 2017; See et al., 2017b) and ***NEWSFARM***. After that, we evaluate the quality of the dataset by analyzing the ROUGE scores.

Second, we select five human evaluation metrics, such as fluency, coherence, consistency, informativeness, and novelty to evaluate our dataset by questionnaire.

Third, we design a set of experiments to compare the effects of training the same model with different datasets when using additional data for testing.

5.1 Baseline

Twelve existing automatic text summarization models of different categories were selected to evaluate the datasets.

Among them, the extractive model include LEAD-3, TextRank(Mihalcea and Tarau, 2004), MatchSum(Zhong et al., 2020) and BertSumExt(Liu and Lapata, 2019).

The LEAD-3, which extracts several sentences in front of the text paragraphs as the summary, and the TextRank(Mihalcea and Tarau, 2004) based on the PageRank algorithm, which extracts several sentences with the highest score as the summary. The BertSumExt(Liu and Lapata, 2019), which is based on bert(Devlin et al., 2018) pretraining model and Oracle algorithm. MatchSum(Zhong et al., 2020), which based on bert(Devlin et al., 2018) pretraining model and BertSumExt(Liu and Lapata, 2019).

The abstractive model include seq2seq-att (Chopra et al., 2016; Nallapati et al., 2016a), pointer-gen(See et al., 2017a), Pointer-gen+cov(See et al., 2017a; Zeng et al., 2016), Transformer(Vaswani et al., 2017), BertSumAbs(Liu and Lapata, 2019) and BertSumExtAbs(Liu and Lapata, 2019).

Seq2seq-att:A novel recurrent neural network for the problem of abstractive sentence summarization.

Pointer-gen: Pointer-generator network(See et al., 2017a) is a hybrid between seq2seq and a

Datasets	% of novel n-grams in gold summary				LEAD-3			ORACLE		
	unigrams	bigrams	trigrams	4-grams	R-1	R-2	R-L	R-1	R-2	R-L
NEWSFARM	**18.85**	**19.59**	**30.51**	**42.42**	**45.44**	**30.03**	**38.44**	**59.79**	**53.11**	**55.52**
CLES	6.75	6.86	5.10	7.81	48.59	35.53	45.32	73.46	66.21	72.15
CNN/DM	16.89	54.06	72.28	80.33	40.34	17.70	36.57	55.12	30.55	51.24
NYTimes	22.64	55.59	71.93	80.16	31.85	15.86	23.75	52.08	31.59	46.72

Table 3: Corpus bias towards extractive methods in the ***NEWSFARM***, CLES, CNN/DM and NY Times datasets. We show the proportion of novel n-grams in gold summaries. We also report ROUGE scores for the LEAD-3 and the extractive oracle baseline. Results are computed on the test set.

pointer network(Vinyals et al., 2015), as it allows both copying words via pointing, and generating words from a fixed vocabulary.

Pointer-gen+cov: Coverage mechanism(Zeng et al., 2016) is added to Pointer-gen(See et al., 2017a).

Transformer: This model has an encoder-decoder structure. The encoder is composed of a stack of $N = 6$ identical layers. Each layer has two sub-layers. The first is a multi-head self-attention mechanism and the second is a simple, position-wise fully connected feed-forward network. The decoder is also composed of a stack of $N = 6$ identical layers. In addition to the two sub-layers in each encoder layer, the decoder inserts a third sub-layer, which performs multi-head attention over the output of the encoder stack.

BertSumAbs and BertSumExtAbs:A new fine-tuning schedule based on bert(Devlin et al., 2018) pre-training model, which adopts different optimizers for the encoder and the decoder as a means of alleviating the mismatch between the two (the former is pre-training while the latter is not). This two-staged fine-tuning approach can further boost the quality of the generated summaries.

5.2 Automatic Evaluation

Using the baseline model mentioned in 5.1, we train the model on ***NEWSFARM***, and CNN/DM respectively. The ROUGE score results obtained are shown in Table 4(extractive model) and Table 5(abstractive model), we can find that ***NEWSFARM***s scores, whether its ROUGE-1, ROUGE-2, or ROUGE-L are higher than CNN/DM in all abstractive baseline models. Especially, the Transformer(Vaswani et al., 2017) showed the greatest performance improvement. Compared to CNN/DM, ***NEWSFARM*** has produced better models, and this result is a testament to the quality of our dataset. In addition, our dataset also obtained better results on the extractive models. Due to the characteristics of TextRank(Mihalcea and Tarau, 2004) algorithm, its rouge score does not have strong proof significance, so it is only used as a comparison here.

The experimental results also showed that although the Chinese pre-training model has developed in recent years, there is still a certain gap between it and English pre-training. We transferred the original English pre-training baseline model to Chinese pre-training and adjusted the parameters repeatedly, but the results were even worse than some traditional models without pre-training. This indicates that Chinese pre-training still needs further development, but it does not affect us to use them as a baseline to prove the high quality of our dataset. The scores of our dataset on models requiring pre-training also exceeded those of CNN/DM, this is further evidence of the high quality of our dataset. Actual training effects on six abstractive baseline models are shown in Appendix B, Figure 2.

5.3 Human Evalution

There are five significant metrics in human evaluation.

(1)**Fluency**: The summary is written smoothly and there are no grammar mistakes.

(2)**Coherence**: Each sentence in the summary needs to be connected organically and meaningfully.

(3)**Consistency**:The facts stated in the summary should be consistent with the source text.

(4)**Informativeness**:The summary captures key points from the source text.

(5)**Novelty**:Use as few sentences, phrases, and words as possible from the source text in the summary.

In this part, the corresponding score was obtained by questionnaire. The full score of each

	NEWSFARM			CNN/DM		
models	Rouge-1	Rouge-2	Rouge-L	Rouge-1	Rouge-2	Rouge-L
LEAD3	45.44	30.03	38.44	40.42	17.62	36.67
oracle	59.79	53.11	55.52	52.59	17.62	36.67
TextRank	43.08	26.86	33.35	35.23	13.90	31.48
BertSumExt	47.74	35.62	43.24	43.23	20.24	39.63
MatchSum+bert	46.56	32.77	46.46	44.22	20.62	40.38
MatchSum+roberta	44.98	31.62	44.89	44.41	20.86	40.55

Table 4: ROUGE scores of extractive models on ***NEWSFARM***, and non-anonymized version of the CNN/DM.

	NEWSFARM			CNN/DM		
models	Rouge-1	Rouge-2	Rouge-L	Rouge-1	Rouge-2	Rouge-L
Seq2seq-att	**56.35**	**47.06**	**54.85**	31.33	11.81	28.83
Pointer-gen	**58.70**	**47.84**	**54.58**	36.44	15.66	33.42
Pointer-gen+cov	**56.37**	**44.44**	**51.49**	39.53	17.28	36.38
Transformer	**61.08**	**49.94**	**56.30**	40.05	17.72	36.67
BertSumAbs	**51.77**	**38.35**	**44.97**	41.72	19.39	38.76
BertSumExtAbs	**52.50**	**39.24**	**45.90**	42.13	19.60	39.18

Table 5: ROUGE scores of abstractive models on ***NEWSFARM***, and non-anonymized version of the CNN/DM.

	OURS	CLES	CNN/DM
Fluency	**4.62**	4.36	4.56
Coherence	**4.23**	3.87	3.90
Consistency	**4.36**	3.89	3.72
Informativeness	**4.26**	3.86	3.96
Novelty	**3.60**	3.56	3.58

Table 6: Human evaluation on ***NEWSFARM***, CLES, and CNN/DM respectively.

metric is 5 points. The scoring standards and samples of human evaluation metrics are shown in Appendix B, Figure 3. Different groups of people were selected and the mean value was obtained according to the statistical results of the questionnaire. The average score of the questionnaire survey is shown in Table 5. Comparison results of different metrics of different datasets, it can be found that ***NEWSFARM*** achieved high scores on all five metrics, which exceeded other datasets. These scores fully demonstrate the high quality of our dataset.

5.4 Results on Out-of-domain Data

To further demonstrate the superiority of the model trained by our dataset, we designed some experiments to test the effects when using additional data(some data which is not in ***NEWSFARM***). The results of the test are shown in Appendix B, Figure 4.

The experiments show that the model trained by our dataset has a good test effect on Out-of-domain data, and a relatively ideal summary has been obtained.

Through the above three aspects of experiments, we can find that each experiment has achieved positive results, which fully proves the high quality of our dataset from three aspects: the quality of the model trained by datasets, the score of the human evaluation indicators, and the actual effect of the model trained by ***NEWSFARM***.

6 Conclusion and future work

NEWSFARM is the largest highest quality Chinese long news summarization dataset at present, which contains long news and corresponding summaries in various fields. The ***HSS*** algorithms help improve the quality of our dataset. Moreover, data analysis shows the scale of our dataset, and the experiments fully demonstrate the quality of our dataset. We hope that ***NEWSFARM*** can not only accelerate the development of automatic text summarization but also promote the formation of a higher-quality summarization system to facilitate our lives.

In the future, more and larger datasets of various types need to be proposed to support larger and larger models.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Kai Chen, Guanyu Fu, Qingcai Chen, and Baotian Hu. 2021. A large-scale chinese long-text extractive summarization corpus. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7828–7832. IEEE.

Sumit Chopra, Michael Auli, and Alexander M Rush. 2016. Abstractive sentence summarization with attentive recurrent neural networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 93–98.

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. 2018. A discourse-aware attention model for abstractive summarization of long documents. *arXiv preprint arXiv:1804.05685*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Alexander R Fabbri, Irene Li, Tianwei She, Suyi Li, and Dragomir R Radev. 2019. Multi-news: A large-scale multi-document summarization dataset and abstractive hierarchical model. *arXiv preprint arXiv:1906.01749*.

Tobias Falke and Iryna Gurevych. 2017. Bringing structure into summaries: Crowdsourcing a benchmark corpus of concept maps. *arXiv preprint arXiv:1704.04452*.

Demian Gholipour Ghalandari, Chris Hokamp, Nghia The Pham, John Glover, and Georgiana Ifrim. 2020. A large-scale multi-document summarization dataset from the wikipedia current events portal. *arXiv preprint arXiv:2005.10070*.

George Giannakopoulos, Mahmoud El-Haj, Benoit Favre, Marina Litvak, Josef Steinberger, and Vasudeva Varma. 2011. Tac 2011 multiling pilot overview.

Bogdan Gliwa, Iwona Mochol, Maciej Biesek, and Aleksander Wawer. 2019. Samsum corpus: A human-annotated dialogue dataset for abstractive summarization. *arXiv preprint arXiv:1911.12237*.

Hongyu Gong, Tarek Sakakini, Suma Bhat, and JinJun Xiong. 2018. Document similarity for texts of varying lengths via hidden topics. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2341–2351, Melbourne, Australia. Association for Computational Linguistics.

Max Grusky, Mor Naaman, and Yoav Artzi. 2018. Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies. *arXiv preprint arXiv:1804.11283*.

Donna Harman and Paul Over. 2004. The effects of human variation in duc summarization evaluation. In *Text Summarization Branches Out*, pages 10–17.

Karl Moritz Hermann, Tomáš Kočiskỳ, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. *arXiv preprint arXiv:1506.03340*.

Baotian Hu, Qingcai Chen, and Fangze Zhu. 2015. Lcsts: A large scale chinese short text summarization dataset. *arXiv preprint arXiv:1506.05865*.

Kuan-Hao Huang, Chen Li, and Kai-Wei Chang. 2020. Generating sports news from live commentary: A chinese dataset for sports game summarization. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 609–615.

Anastassia Kornilova and Vlad Eidelman. 2019. Billsum: a corpus for automatic summarization of us legislation. *arXiv preprint arXiv:1910.00523*.

Kemal Kurniawan and Samuel Louvan. 2018. Indosum: A new benchmark dataset for indonesian text summarization. In *2018 International Conference on Asian Language Processing (IALP)*, pages 215–220. IEEE.

Guy Lev, Michal Shmueli-Scheuer, Jonathan Herzig, Achiya Jerbi, and David Konopnicki. 2019. TalkSumm: A dataset and scalable annotation method for scientific paper summarization based on conference talks. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2125–2131, Florence, Italy. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Peipei Li, Lu He, Haiyan Wang, Xuegang Hu, Yuhong Zhang, Lei Li, and Xindong Wu. 2017. Learning from short text streams with topic drifts. *IEEE transactions on cybernetics*, 48(9):2697–2711.

Yongjun Li, Zhen Zhang, You Peng, Hongzhi Yin, and Quanqing Xu. 2018. Matching user accounts based on user generated content across social networks. *Future Generation Computer Systems*, 83:104–115.

Chin-Yew Lin and Eduard Hovy. 2003. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 150–157.

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. *arXiv preprint arXiv:1908.08345*.

Christian M Meyer, Darina Benikova, Margot Mieskes, and Iryna Gurevych. 2016. Mdswriter: Annotation tool for creating high-quality multi-document summarization corpora. In *Proceedings of ACL-2016 System Demonstrations*, pages 97–102.

Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing*, pages 404–411.

Alessandro Moschitti. 2006. Efficient convolution kernels for dependency and constituent syntactic trees. In *European Conference on Machine Learning*, pages 318–329. Springer.

Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2017. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Çağlar Gulçehre, and Bing Xiang. 2016a. Abstractive text summarization using sequence-to-sequence RNNs and beyond. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 280–290, Berlin, Germany. Association for Computational Linguistics.

Ramesh Nallapati, Bowen Zhou, Caglar Gulcehre, Bing Xiang, et al. 2016b. Abstractive text summarization using sequence-to-sequence rnns and beyond. *arXiv preprint arXiv:1602.06023*.

Courtney Napoles, Matthew R Gormley, and Benjamin Van Durme. 2012. Annotated gigaword. In *Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction (AKBC-WEKEX)*, pages 95–100.

Shashi Narayan, Shay B Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. *arXiv preprint arXiv:1808.08745*.

Dragomir R. Radev, Pradeep Muthukrishnan, and Vahed Qazvinian. 2009. The ACL Anthology network. In *Proceedings of the 2009 Workshop on Text and Citation Analysis for Scholarly Digital Libraries (NLPIR4DL)*, pages 54–61, Suntec City, Singapore. Association for Computational Linguistics.

Alexander M Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. *arXiv preprint arXiv:1509.00685*.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017a. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Abigail See, Peter J Liu, and Christopher D Manning. 2017b. Get to the point: Summarization with pointer-generator networks. *arXiv preprint arXiv:1704.04368*.

Eva Sharma, Chen Li, and Lu Wang. 2019. Bigpatent: A large-scale dataset for abstractive and coherent summarization. *arXiv preprint arXiv:1906.03741*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762*.

Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. 2015. Pointer networks. *arXiv preprint arXiv:1506.03134*.

Wentao Wu, Hongsong Li, Haixun Wang, and Kenny Q Zhu. 2012. Probase: A probabilistic taxonomy for text understanding. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*, pages 481–492.

Xuefeng Xi, Zhou Pi, and Guodong Zhou. 2020. Global encoding for long chinese text summarization. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 19(6):1–17.

Han Xu, Zhang Zhengyan, Ding Ning, Gu Yuxian, Liu Xiao, Huo Yuqi, Qiu Jiezhong, Zhang Liang, Han Wentao, Huang Minlie, et al. 2021. Pre-trained models: Past, present and future. *arXiv preprint arXiv:2106.07139*.

Jiaqi Yang, Yongjun Li, Congjie Gao, and Yinyin Zhang. 2021. Measuring the short text similarity based on semantic and syntactic information. *Future Generation Computer Systems*, 114:169–180.

Michihiro Yasunaga, Jungo Kasai, Rui Zhang, Alexander R Fabbri, Irene Li, Dan Friedman, and Dragomir R Radev. 2019. Scisummnet: A large annotated corpus and content-impact models for scientific paper summarization with citation networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7386–7393.

Peng Yuan, Haoran Li, Song Xu, Youzheng Wu, Xiaodong He, and Bowen Zhou. 2020. On the faithfulness for e-commerce product summarization. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5712–5717.

Wei Zeng, Xiaozhe Ren, Teng Su, Hui Wang, Yi Liao, Zhiwei Wang, Xin Jiang, ZhenZhang Yang, Kaisheng Wang, Xiaoda Zhang, et al. 2021. Pangu: Large-scale autoregressive pretrained chinese language models with auto-parallel computation. *arXiv preprint arXiv:2104.12369*.

Wenyuan Zeng, Wenjie Luo, Sanja Fidler, and Raquel Urtasun. 2016. Efficient summarization with read-again and copy mechanism. *arXiv preprint arXiv:1611.03382*.

Kaihang Zhang, Chuang Zhang, Xiaojun Chen, and Jianlong Tan. 2018. Automatic web news extraction based on ds theory considering content topics. In *International Conference on Computational Science*, pages 194–207. Springer.

Ming Zhong, Pengfei Liu, Yiran Chen, Danqing Wang, Xipeng Qiu, and Xuanjing Huang. 2020. Extractive summarization as text matching. *arXiv preprint arXiv:2004.08795*.

Junnan Zhu, Qian Wang, Yining Wang, Yu Zhou, Jiajun Zhang, Shaonan Wang, and Chengqing Zong. 2019. Ncls: Neural cross-lingual summarization. *arXiv preprint arXiv:1909.00156*.

A Appendix

This part is a detailed introduction of ***HSS*** algorithm. ***HSS*** algorithm consists of three parts:the hidden text topic(hiden_topic), semantic similarity(semantic_sim) and syntactic similarity(syntactic_sim).

The hidden text topic:

Step1: The preprocessing module tokenizes articles and removes stop words and prepositions.

Step2: Topic generation from articles, the word vectors in a article are $W = \{w_1, w_2, ..., w_n\}$, and the hidden topic vectors of the article are $H = \{h_1, h_2, ..., h_n\}$. We define the reconstructed word vector $\widetilde{w}_i$ for the word w_i as the optimal linear approximation given by topic vectors: $\widetilde{w}_i = H\widetilde{\alpha}_i$ where:

$$\widetilde{\alpha}_i = argmin||w_i - \widetilde{\alpha}_i||_2^2 \quad (1)$$

$$\alpha_i \in R^k$$

The reconstruction error E for the whole article is given by:

$$E = \sum_{i=1}^{n} ||w_i - \widetilde{w}_i||_2^2 \quad (2)$$

The goal is to find the optimal H^* so as to minimize the error E. $H \in R^{d*k}$.

$$H^* = argmin \sum_{i=1}^{n} min||w_i - H\alpha_i||_2^2 \quad (3)$$

With the orthonormal constraints, we simplify the form of the reconstructed vector $\widetilde{w}_i$ as:

$$\widetilde{w}_i = HH^T w_i \quad (4)$$

Define E_k as the reconstruction error when we only use topic vector h_k^* to reconstruct the article:

$$E_k = ||W - h_k^{*T} W||_2^2 \quad (5)$$

Now define i_k as the importance of topic h_k^* , which measures the topics ability to reconstruct the words in a article: $i_k = ||h_k^{*T} W||_2^2$. We normalize i_k as $\bar{i}_k$ so that the importance does not scale with the norm of the word matrix W, and so that the importances of the K topics sum to 1.

$$\bar{i}_k = \frac{i_k}{(\sum_{j=1}^{k} i_j)} \quad (6)$$

The number of topics K is a hyperparameter in model.

Step3: Topic mapping to summaries. We have extracted K topic vectors $\{h_k^*\}_k^{k=1}$from the text matrix W, whose importance is reflected by $\{\hat{i}_k^*\}_k^{k=1}$. This part, we measure the relevance of a article-summary pair. Suppose the vectors of the words in a summary are $S = \{s_1, s_2, ..., s_m\}$. Similar to the reconstruction of the article, the summary can also be reconstructed from the articles topic vectors as shown in Eq.(4). Let $\widetilde{s}_j^k$ be the reconstruction of the summary word s_j given by one topic h_k^*. $\widetilde{s}_j^k = h_k^* h_k^{*T} S_j$. The $r(h_k^*, s_j)$ is the relevance between a topic vector h_k^* and summary word s_j. It is defined as the cosine similarity between $\widetilde{s}_j^k$ and s_j:

$$r(h_k^*, s_j) = \frac{S_j^T \widetilde{s}_j^k}{||s_j||_2 * ||\widetilde{s}_j^k||_2} \quad (7)$$

Let $r(h_k^*, s_j)$ be the relevance between a topic vector and the summary, defined to be the average similarity between the topic vector and the summary words:

$$r(h_k^*, S) = \frac{1}{m} \sum_{m}^{j=1} r(h_k^*, s_j) \quad (8)$$

Therefore, the $r(W,S)$ as the relevance between the text W and the summary S, and $r(W,S)$ is the sum of topic-summary relevance weighted by the importance of the topic:

$$r(W,S) = \sum_{K}^{k=1} \overline{i_k} * r(h_k^*, S) \tag{9}$$

Step4: The higher $r(W,S)$ is, the better the summary matches the article.

Measuring the text similarity based on semantic

Definition 1(Term). A term t is a word or a multiword expression(MWE).

Definition 2(Instance). An instance e is a concrete object.

Definition 3(Concept). A concept c is defined as a general and abstract description of a set of instances.

*a.Semantic similarity between terms:*the term similarity calculation can be roughly divided into a knowledge-based method and a corpus based method. Both types of methods have shortcomings, so we combine the two methods to get a better solution.

First, the corpus based method. We use semantic composition to obtain semantic vectors of terms. Given a term t and the semantic vector of each word in t, the semantic vector of t can be calculated by Eq.(10).

$$V_t = \sum_{k=1}^{K} V_{w_k} * SIF(w_k) \tag{10}$$

Where K stands for the number of words, SIF is the smooth inverse frequency based on the attention mechanism. The similarity of terms is computed using the two semantic vectors.

$$R_c(t_1,t_2) = \frac{(V_{t_1} * V_{t_2})}{(|V_{t_1}| * |V_{t_2}|)} \tag{11}$$

Second, the similarity of terms based on knowledge is obtained by Probase and calculated based on the two concept vectors.

$$R_k(t_1,t_2) = \frac{(I_{t_1} * I_{t_2})}{(|I_{t_1}| * |I_{t_2}|)} \tag{12}$$

The large-scale knowledge base Probase(Wu et al., 2012), which is a probabilistic semantic network that contains millions of concepts and instances. The I_{t_1} and I_{t_2} is the concept vector.

A linear method is adopted to fuse R_c and R_k,

$$R = \alpha * R_k + (1-\alpha) * R_c \tag{13}$$

Where α is a tuning parameter. Since R_c plays a subordinate role in term similarity, α should be a value greater than 0.5.

b.Semantic similarity between article and summary:

Step1: Text segmentation. Recently, the simplest and most effective method for segmentation is dictionary-based matching. It is a greedy algorithm to match the longest length, which only optimizes the local solution, not global optimization. We rely on a fact that the segmented terms should be semantically related.

First, all possible segmentations are generated recursively.

Second, terms that do not have segmentation ambiguity from all text segmentation cases are chosen. These terms are used as a reference to select the most related segmentation.

Third, the semantic similarity between the segment and each reference term is calculated, and the highest score is preserved.

Finally, the segment with the maximum score is selected as the best segmentation.

Step2: Part of speech judgment. For the terms set after segmentation, we need to determine the POS of each term in the current context. Stanford CoreNLP is used to determine the POS of each term. We use the method in(Li et al., 2017) to further distinguish the type of noun terms(concept or instance).

Step3: Conceptualization of term. With the help of Probase(Wu et al., 2012), we can easily obtain concepts of instances. However, in natural language, instances are often ambiguous('apple' can stand for both fruit and company), it has at least two completely unrelated concepts. For ambiguous terms, we need to select appropriate contextual terms to eliminate ambiguity.

All contextual terms are considered, and a priority is assigned to each informative contextual term by a variant sigmoid function.

$$weight(t_i) = 1.5 - \frac{1}{1+e^{-x}} \tag{14}$$

Where x represents the contextual distance, and the contextual distance refers to the number of terms between t_i and the target instance. Based on the above context selection and assigned weights,

the concept with the maximum score is considered the meaning of the target word in the current context({apple,pos}→{apple,pos,company}, the company is the concept information).

Step4: Semantic vector of texts. After the above three parts, we have constructed the semantic vector. Using the semantic vector of the article and summary, the similarity score of the<$article, summary$> pair is obtained by using the similarity calculation formula(semantic_sim).

First, a matrix S is constructed based on the number of terms in article of summary.

Second, the similarity of each term pair is computed based on the Eq.(12).

Finally, we compute the sum of all values in S and normalize it, obtaining the similarity score.

$$semantic\_sim = \frac{S(T_1, T_2)}{\sqrt{S(T_1, T_1) * S(T_2, T_2)}} \quad (15)$$

Where the T_1 and T_2 is terms set of the article and summary. Each term in the term set is represented as a triple (term, POS, concept). The semantic vector of S1 and S2 following(Li et al., 2018).

First, based on T_1 and T_2, a joint term set $T = T_1 \cup T_2$is formed and each entry of the semantic vector corresponds to a term in T.

Second, obtaining the value of the each entry based on term similarity.

-If t_i belongs to T_1, the value is set to 1.

-If t_i does not belong to T_1, we calculate the semantic similarity between ti and each term in T_1.

In essence, the attention mechanism is added in Eq.14.

Measuring the text similarity based on Syntactic

The above semantic similarity calculation method of texts is simple and effective, but it ignores the impact of syntactic information. We compute the syntactic similarity of texts based on a constituency parse tree(CPT).

step1: Construct the CPT with terms as the minimum semantic unit.

step2: Compute the similarity of each node pair based on rules.

$$PTK(T_1, T_2) = \sum_{n_1 \in N_{T_1}} \sum_{n_2 \in N_{T_2}} \triangle(n1, n2) \quad (16)$$

Where T_1 and T_2 are the CPTs of S_1 and S_2, NT_1 and NT_2 are the sets of nodes in T_1 and T_2, and $\triangle(n_1, n_2)$ refers to the number of common fragments rooted at the n_1 and n_2 nodes which is the core of the tree kernel.

We define $\triangle(n_1, n_2)$(Moschitti, 2006) as follows.

$\triangle(n_1, n_2) = similarity(n_1, n_2)$, when $n_1, n_2 \in leafnodes$.

$\triangle(n_1, n_2) = U(\lambda^2 + \sum_{p=1} lm \triangle p(cn_1, cn_2))$, when $n_1 == n_2 and n_1, n_2 \in non - leafnodes$

Otherwise,$\triangle(n_1, n_2) = 0$

Where u is the height of the tree and λ is the length of the child sequences, cn_1 and cn_2 are the ordered child sequences of n_1 and n_2, lm returns the minimum sequence length between cn_1 and cn_2, and $\triangle p$ evaluates the number of common subtrees rooted in the subsequence of exactly p children. $\triangle p$ is a recursive function.

$$\triangle p(cn_1, cn_2) = \triangle(a, b) * \sum_{i=1} |n_1| \sum_{r=1} |n_2| \lambda^{|n_1|+|n_2|-i-r} * \triangle(p-1) * (n_1[1:i], n_2[1:r]) \quad (17)$$

where $n_1[1:i]$ and $n_2[1:r]$ are the child subsequences from 1 to i and from 1 to r of n_1 and n_2. $\triangle(p-1)$ is recursively computed using Eq.(17)

step3: Sum all similarity values and normalize the sum to get the syntactic similarity(Moschitti, 2006)(syntactic_sim)

Overall text similarity:A linear method is adopted to fuse the hidden text topic approch and the text with semantic and syntactic information.

$$sim(summary, article) = \theta(\varphi * semantic\_sim + (1-\varphi) * syntactic\_sim) + (1-\theta) * hiden\_topic \quad (18)$$

Where θ is a tuning parameter. Since $hiden\_topic$ plays a subordinate role in text similarity, θ should be a value greater than 0.5.

The overall flow of the HSS is as follows:

step1: The preprocessing module tokenizes texts and removes stop words and prepositions

step2: Topic generation from texts and minimize the reconstruct error E.

step3: Topic mapping to summaries, and get the $r(W, S)$ as the relevance between the text W and the summary S.

step4: Obtaining the term set T_1(article) and T_2(summary) according to the text segmentation technique.

step5: Judging the POS of each term and get T_1{($term_1$,pos)...($term_n$,pos)} and T_2{($term_1$,pos)...($term_n$,pos)}.

step6: Conceptualizing each term and get T_1{($term_1$,pos,concept)...($term_n$,pos,concept)} and T_2{($term_1$,pos,concept)...($term_n$,pos,concept)}.

step7: Constructing the semantic matrix S of the two text, then using Eq.(15) to obtain the semantic similarity semantic_sim.

step8: Constructing the CPT of each text based on the term set obtained in step1. And then using the rules defined in Eq.(16) to obtain the syntactic similarity syntactic_sim.

step9: Using the rules defined in Eq.(18) to obtain the overall text similarity. The higher sim(article,summary) is, the better the summary matches the article. We compare the score with the threshold value, and filter out the records below the threshold.

B Appendix

Summary:联合国西亚经济社会委员会今天发布了2019-2020年度“阿拉伯地区经济和社会发展概览”（Survey of Economic and Social Developments in the Arab Region）报告。数据显示，阿拉伯地区2021年面临两种经济形势：在乐观的情况下，增长率预计为3.5%，在较不乐观的情况下，增长率预计将不超过2.8%

Summary: The United Nations Economic and Social Commission for Western Asia today released its "Survey of Economic and Social Developments in the Arab Region" for 2019-20. The data show that the Arab region faces two economic scenarios in 2021: in the optimistic case, growth is expected to be 3.5 percent, and in the less optimistic case, growth is expected to be no more than 2.8 percent.

News:联合国西亚经社会：乐观估计2021年阿拉伯地区经济增长达3.5%。报告指出，阿拉伯地区的实际增长情况将取决于阿拉伯国家应对新冠大流行的能力，疫情给该地区造成了大约1400亿美元的损失，导致今年经济增长收缩了3%。
报告警告说，尽管在两种情况下都有望实现正增长，但这不足以创造体面的就业机会。实际上，预计该地区的失业率将在2021年上升到12.5%。巴勒斯坦和利比亚的失业率将达到地区最高水平，分别为31%和22%；在约旦和突尼斯，这一数字将超过21%，在海湾合作委员会成员国（Gulf Cooperation Council）中，这一数字将达到5.8%左右。而且，今年该地区的出口额下降了50%，预计2021年将再增长10.4%。
报告主要作者穆罕默德·赫迪·比奇尔（Mohamed Hedi Bchir）表示，阿拉伯地区面临的危机已不仅限于经济领域，还包括重大的社会挑战。该地区还面临日益严重的贫困，到2021年，平均贫困率可能达到32%，影响1.16亿人。该地区正与不断上升的青年失业率以及持续存在的性别不平等现象作斗争——平均青年失业率达27%，性别不平等差距达40%，为全球最高。
报告称，中等收入国家有望成为该地区增长率最高的国家，在乐观的情况下达到5%，在较不乐观的情况下达到4.1%。海湾合作委员会成员国将实现2.3%至2.1%的增长率，最不发达国家的增长率将不超过0.5%或0.4%。
比奇尔进一步强调，该地区面临的挑战需要阿拉伯政府作出广泛的努力，以提供必要的社会安全网，特别是在收容难民和移民的社区中，由于经济衰退困扰着捐助国，人们越发担心这些社区生活条件进一步恶化。
今年的报告重点关注阿拉伯地区的债务水平，该地区债务水平在过去十年中翻了一番，不受冲突影响的阿拉伯国家的债务约为1.2万亿美元，约占阿拉伯中等收入国家GDP的80%。这种可怕的情况是由于该地区大多数国家仍然依靠借贷来为政府支出提供资金而造成，这对生产率和增长产生了负面影响。报告指出，这一局势显示出治理薄弱的问题，要求各国确定“如何”支出，而不是支出多少。报告还指出，如果当前债务状况持续下去，只会加深当前的社会经济危机，特别是在那些无法从面向低收入国家的20国集团暂停偿还债务倡议中受益的中等收入国家。通过这一倡议，低收入国家节省了约2.94亿美元。因此，报告要求将这一倡议的覆盖范围扩大到中等收入国家，这些国家的还本付息的偿债额已达到180亿美元，但前提是它们设定了财政赤字上限以确保债务可持续性。

News: Escwa: Optimistic forecast for Arab economic growth of 3.5% in 2021 Actual growth in the Arab region will depend on how well Arab countries respond to the COVID-19 pandemic, which has cost the region an estimated $140 billion and caused economic growth to contract by 3 percent this year, the report said.
While positive growth is expected in both cases, it is not enough to create decent jobs, the report warns. In fact, the region's unemployment rate is expected to rise to 12.5 percent by 2021. Palestinian and Libyan unemployment rates will be the highest in the region, at 31 and 22 per cent respectively. In Jordan and Tunisia, the figure will be more than 21 per cent, and in the Gulf Cooperation Council it will be about 5.8 per cent. Moreover, the region's exports fell 50 percent this year and are expected to grow another 10.4 percent in 2021.
The report's lead author, Mohamed Hedi Bchir, says the crisis facing the Arab region is no longer limited to the economic sphere, but also includes major social challenges. The region also faces increasing poverty, with an average poverty rate likely to reach 32 percent by 2021, affecting 116 million people. The region is struggling with rising youth unemployment, averaging 27 per cent, and persistent gender inequality, the highest in the world, at 40 per cent.Middle-income countries are expected to have the highest growth rates in the region, reaching 5 percent in an optimistic case and 4.1 per cent in a less optimistic case, the report said. GCC countries will achieve growth rates of 2.3 to 2.1 percent and the least developed countries will not exceed 0.5 or 0.4 percent.
Bichir further stressed that the challenges facing the region require extensive efforts by Arab governments to provide the necessary social safety nets, particularly in communities hosting refugees and migrants, where fears of further deterioration in living conditions are growing as economic recession afflicts donor countries.
This year's report focuses on debt levels in the Arab region, which have doubled in the past decade to about $1.2 trillion in conflict-free Arab countries, or about 80 percent of the GDP of middle-income Arab countries. This dire situation is due to the fact that most countries in the region still rely on borrowing to finance government spending, which has a negative impact on productivity and growth. The report says the situation shows weak governance and requires countries to decide "how" to spend, not how much.The report also notes that if the current debt situation continues, it will only deepen the current socio-economic crisis, especially in middle-income countries that will not benefit from the G20 debt standstill initiative for low-income countries. Low-income countries have saved about $294 million through this initiative. The report therefore calls for the initiative to be extended to middle-income countries, which have already reached $18 billion in debt service, but only if they set a ceiling on their fiscal deficits to ensure debt sustainability.

Figure 1: An example of the ***NEWSFARM***.

News:“ 沾链就涨” 成过去时去年以来仅1/3区块链概念股跑赢大盘多公司今年调减投资额.近年来,“ 区块链 ” 热潮席卷全球，不少上市公司也一头扎入其中。去年1月份，在区块链与加密数字货币价格位居高点时，多家上市公司宣布进军区块链，彼时市场曾出现一天内11只相关概念股涨停的盛况。但时隔1年有余 ，区块链概念股走势如何呢 ？《 证券日报》记者根据东方财富Choice数据统计，截至昨日收盘，80只区块链概念股中跑赢大盘的公司数量近28家，占比仅为35%左右。绝大多数股票已冲高回落。尽管区块链行业不断发展 ，但区块链技术大规模应用落地仍待时日。在去年A股市场区块链市场虚火旺盛之后，今年以来，已有不止一家A股公司已开始将目光从区块链上转移。概念股冲高 回落《 证券日报》记者根据东方财富Choice数据统计，在去年6月份时，纳入区块链概念股的A股上市公司有55家，而 截止到目前，数量已经达到80家。本报记者统计了从去年年初至昨日，区块链概念个股的涨跌幅情况，80家公司的区间 涨跌幅在- 72%-62%之间。其中，涨幅最高的是用友网络(62 . 43 %)，其次为恒生电子(45.31%) 。如果以大盘为基准进 行衡量的话，在该区间内，仅有28只区块链概念股的走势好于大盘，仅有10只股票收涨——这也意味着，投资者如果 “ 闭着眼”从概念股中选股的话，仅有35 %的几率跑赢大盘，仅有12.5%的几率飘红。
…
Since last year, only one-third of blockchain concept stocks have outperformed the market and many companies have reduced their.In recent years, the "blockchain" craze has swept the world, and many listed companies have jumped into it. In January last year, when the price of blockchain and cryptocurrency was at its peak, a number of listed companies announced their entry into blockchain. At that time, the market saw 11 related stocks rise by their daily limit in one day. But after more than a year, blockchain concept stock trend? Securities Daily reporter according to the eastern wealth Choice data statistics, as of yesterday's close, 80 blockchain concept stocks outperforming the market of nearly 28 companies, accounting for only about 35%. Most stocks have retreated from their gains. Despite the continuous development of the blockchain industry, the large-scale application of blockchain technology is still a long way off. After the boom in the a-share market blockchain market last year, since this year, more than one A-share company has begun to shift its focus from the blockchain. "Securities Daily" reporter according to the eastern wealth Choice data statistics, in June last year, the inclusion of block chain concept stocks of A shares listed companies 55, and up to now, the number has reached 80. Our reporter statistics from the beginning of last year to yesterday, the rise and fall of blockchain concept stocks, the range of 80 companies rose or fell between -72% to 62%. Among them, yonyou Network rose the most (62.43 percent), followed by Hang Seng Electronics (45.31 percent). Only 28 blockchain-based stocks outperformed the broader market, and only 10 stocks finished higher -- meaning that investors picking from blockchain-based stocks with a "closed eye" have a 35% chance of outperforming the market and a 12.5% chance of going red.
...
Gold summary:“沾链就涨”成过去时去年以来仅1/3区块链概念股跑赢大盘，多公司今年调减投资额 。近年来，“区块链”热潮席卷 全球，不少上市公司也一头扎入其中。而在今年1月初，上市公司远光软件公告称，调减对所投区块链产业基金的投资。
Since last year, only 1/3 blockchain concept stocks have outperformed the market, and many companies have reduced their investment this year. In recent years, the "blockchain" craze has swept the world, and many listed companies have jumped into it. In early January this year, listed company Yuanguang software announced that the reduction of investment in the blockchain industry fund.

Seq2seq+att:1/3区块链概念股跑赢大盘，多公司今年调减投资额，近年来，“区块链”热潮席卷全球，不少上市公司也一头扎入其中。
1/3 blockchain concept stocks have outperformed the market, and many companies have reduced their investment this year. In recent years, the "blockchain" craze has swept the world, and many listed companies have jumped into it.
Pointer+gen:尽管区块链行业不断发展，但区块链技术大规模应用落地仍待时日。
Despite the continuous development of the blockchain industry, the large-scale application of blockchain technology is still a long way off.
Pointer+gen+cov:去年1月份，在区块链与加密数字货币价格位居高点时，多家上市公司宣布进军试链，彼时市场曾出现一天内11只相关概念股涨停的盛况。
In January last year, at the peak of the blockchain and cryptocurrency prices, a number of listed companies announced they were entering the trial chain, with the market seeing 11 rallies in one day and concept stocks trading of the grand situation.
Transformer:时去年以来仅1/3区块链概念股跑赢大盘多公司今年调减投资额近年来，“区块链”热潮席卷全球，不少上市公司也一头扎入其中。数据显示，2018年区块链整体环比下降2.3%。
In recent years, the "block chain" craze has swept the world, and many listed companies have plunged into it. The data showed that blockchain overall fell 2.3 percent month-on-month in 2018.
BertSumAbs:“沾链就涨”成过去时，去年以来仅1/3区块链概念股跑赢大盘多公司今年调减投资额近年来，” 区块链“ 热潮席卷全球，不少上市公司也一头扎入其中。去年1月份， 在区块链与加密数字货币价格位居高点时，多家上市公司也参与其中。
Many companies have reduced their investment this year. In recent years, the "blockchain" craze has swept the world, and many listed companies have plunged into it. In January last year, when blockchain and cryptocurrency prices were at their peak, a number of listed companies also got involved.
BertSumExtAbs:“沾链就涨”成过去时去年以来仅1/3区块链概念股跑赢大盘,多公司今年调减投资额。近年来，“区块链”热潮席卷全球，不少上市公司也一头扎入其中。
Only one-third of blockchain stocks have outperformed the market since last year, and many companies have reduced their investment this year. In recent years, the "blockchain" craze has swept the world, and many listed companies have also jumped into it.

Figure 2: Actual training effects on four generation baseline models using ***NEWSFARM***.

Fluency

5 points: The summary is written smoothly and there are no grammar mistakes.
保障和改善民生是经济社会发展的出发点和落脚点，也是全面建成小康社会的核心要义所在。会议提出，要重视解决好“一老一小”问题，加快建设养老服务体系，支持社会力量发展普惠托育服务。会议提出，要兜住基本生活底线，确保养老金按时足额发放，加快推进养老保险全国统筹 。
4 points: Only a few logical sequence errors but no grammar mistakes.
改善和保障民生是经济社会发展的落脚点和出发点，也是全面建成小康社会的核心要义所在。会议提出，要重视解决好“一老一小”问题，加快建设养老服务体系，支持社会力量发展普惠托育服务。会议提出，要兜住基本生活底线，确保养老金按时足额发放，加快推进养老保险全国统筹 。(You need “guarantee” to get “improve “.)
3 points: There are a few grammar mistakes.
是经济社会发展的出发点和落脚点，也是全面建成小康社会的核心要义所在。会议提出，要重视解决好“一老一小”问题，加快建设养老服务体系，支持社会力量发展普惠托育服务。会议提出，要兜住基本生活底线，确保养老金按时足额发放，加快推进养老保险全国统筹 。(eg:The lack of the subject)
2 points: Logical sequence errors exist in the summary and there are a few grammar mistakes.
是经济社会发展的落脚点和出发点，也是全面建成小康社会的核心要义所在。会议提出，要重视解决好“一老一小”问题，加快建设养老服务体系，支持社会力量发展普惠托育服务。会议提出，要兜住基本生活底线，确定养老金按时足额发放，加快推进养老保险全国统筹 。(You need “guarantee” to get “improve “. The lack of the subject. Miscollocation of words.)
1 points: Logical sequence errors exist in the summary and there are many grammar mistakes.
是经济社会发展的落脚点和出发点，也是全面建成小康社会的核心要义所在。会议提出，要重视解决好“一老一小”问题，加快成立养老服务体系，支持社会力量发展普惠托育服务。会议提出，要兜住基本生活底线，确定养老金按时足额发放，加快提高养老保险全国统筹 。(You need “guarantee” to get “improve “. The lack of the subject. Miscollocation of words.)

Coherence

5 points: Each sentence in the summary needs to be connected organically and meaningfully.
海南日报记者4月15日从博鳌亚洲论坛2021年年会海南省筹备工作和主题活动情况介绍新闻发布会上获悉，论坛年会召开期间，位于博鳌镇东屿岛的博鳌亚洲论坛主题公园和博鳌亚洲论坛成立20周年回顾展将对公众开放。博鳌亚洲论坛成立20周年回顾展主要内容有博鳌亚洲论坛宣言、组织架构、年会历程、每年主题及20年来海南发生的令人瞩目的变化。
4 points: There are a few problems with the internal connection of sentences.
海南日报记者4月15日从博鳌亚洲论坛2021年年会海南省筹备工作和主题活动情况介绍新闻发布会上获悉，改善和保障民生是经济社会发展的落脚点和出发点，也是全面建成小康社会的核心要义所在。博鳌亚洲论坛成立20周年回顾展主要内容有博鳌亚洲论坛宣言、组织架构、年会历程、每年主题及20年来海南发生的令人瞩目的变化。
3 points: There are a few problems with sentence to sentence connections.
海南日报记者4月15日从博鳌亚洲论坛2021年年会海南省筹备工作和主题活动情况介绍新闻发布会上获悉，论坛年会召开期间，位于博鳌镇东屿岛的博鳌亚洲论坛主题公园和博鳌亚洲论坛成立20周年回顾展将对公众开放。以智能制造统领产业转型升级，以智能制造来推动制造业高质量发展，致力于制造一个“智造之城”。
2 points: There are few problems with connections between sentences and connections within sentences.
海南日报记者4月15日从博鳌亚洲论坛2021年年会海南省筹备工作和主题活动情况介绍新闻发布会上获悉，改善和保障民生是经济社会发展的落脚点和出发点，也是全面建成小康社会的核心要义所在。以智能制造统领产业转型升级，以智能制造来推动制造业高质量发展，致力于制造一个“智造之城”。
1 points: There are many problems with connections between sentences and connections within sentences.
海南日报记者4月15日从博鳌亚洲论坛2021年年会海南省筹备工作和主题活动情况介绍新闻发布会上获悉，改善和保障民生是经济社会发展的落脚点和出发点，也是全面建成小康社会的核心要义所在。以智能制造统领产业转型升级，每年主题及20年来海南发生的令人瞩目的变化。

Consistency

5 points: The facts stated in the summary are consistent with the source text.
4 points: The facts stated in the summary and the original are basically the same, and are easily distinguishable.
3 points: The facts stated in the summary and the original are basically the same, and there are no ambiguity.
2 points: The facts stated in the summary and the original are basically the same, but there may be ambiguity.
1 points: The facts stated in the summary deviate completely from the source text.

Informativeness

5 points: The summary contains all the key points in the news.
4 points: The summary contains most of the key points in the news.
3 points: The summary misses some key points in the news, but it does not affect the understanding.
2 points: The summary misses many key points in the news and affect understanding.
1 points: The summary misses many key points in the news and is barely understandable.

Novelty

5 points: Only proper nouns in the summary are the same as in the source news.
4 points: Except for proper nouns, part of the content of some sentences in the summary comes from the source text.
3 points: Except for proper nouns, most of the content of some sentences in the summary comes from the source text, but the sentence is slightly different from the original text .
2 points: A few sentences in the summary come from the source text.
1 points: The summary is a complete sentence from the source text.

Figure 3: The scoring standards and samples of human evaluation metrics. Colored fonts indicate penalty points

News:5月3日是世界新闻自由日联合国秘书长今天敦促各国政府“尽其所能“来支持自由、独立和多样化的媒体，同时联合国人权高级专员巴切莱特强调新闻自由是“民主社会的基石”。古特雷斯秘书长在致辞中强调了可靠、经核实和可获取的信息的重要性。他表示，在新冠大流行期间，以及在包括气候紧急情况在内的其他危机中，记者和…
The UN secretary-general today urged governments to ″do everything in their power″ to support free, independent and diverse media, while THE UN High Commissioner for Human Rights, Michelle Bachelet, stressed that press freedom is a ″cornerstone of democratic societies″. In his address, Secretary-General Guterres stressed the importance of reliable, verified and accessible information. He said that during the COVID-19 pandemic and in other crises, including climate emergencies, journalists and...

Transformer trained on ***NEWSFARM***:
联合国秘书长古特雷斯今天敦促各国政府“尽其所能”来支持自由、独立和多样化的媒体，同时联合国人权高级专员巴切莱特强调新闻自由是”民主社会的基石”。
UN Secretary-General Antonio Guterres today urged governments to ″do everything in their power″ to support free, independent and diverse media, while UN High Commissioner for Human Rights Michelle Bachelet stressed that press freedom is a ″cornerstone of democratic societies″.

Figure 4: An example on the out-of-domain data of Transformer model trained on ***NEWSFARM***. The underlined words are creative enhancing part.