applied sciences

MDPI

Article

Improving Sentence Retrieval Using Sequence Similarity

Ivan Boban [1,*], Alen Doko [2] and Sven Gotovac [3]

[1] Faculty of Mechanical Engineering, Computing and Electrical Engineering, University of Mostar, 88000 Mostar, Bosnia and Herzegovina

[2] Institute for Software Technology, German Aerospace Center, 28199 Bremen, Germany; alen.doko@dlr.de

[3] Faculty of Electrical Engineering, Mechanical Engineering and Naval Architecture, University of Split, 21000 Split, Croatia; sven.gotovac@fesb.hr

* Correspondence: ivan.boban@hotmail.com or ivan.boban@student.fsre.ba; Tel.: +387-63-484-395

Received: 1 June 2020; Accepted: 19 June 2020; Published: 23 June 2020

Abstract: Sentence retrieval is an information retrieval technique that aims to find sentences corresponding to an information need. It is used for tasks like question answering (QA) or novelty detection. Since it is similar to document retrieval but with a smaller unit of retrieval, methods for document retrieval are also used for sentence retrieval like term frequency—inverse document frequency (TF-IDF), BM25, and language modeling-based methods. The effect of partial matching of words to sentence retrieval is an issue that has not been analyzed. We think that there is a substantial potential for the improvement of sentence retrieval methods if we consider this approach. We adapted TF-ISF, BM25, and language modeling-based methods to test the partial matching of terms through combining sentence retrieval with sequence similarity, which allows matching of words that are similar but not identical. All tests were conducted using data from the novelty tracks of the Text Retrieval Conference (TREC). The scope of this paper was to find out if such approach is generally beneficial to sentence retrieval. However, we did not examine in depth how partial matching helps or hinders the finding of relevant sentences.

Keywords: sentence retrieval; TF−ISF; BM25; language modeling; partial match; sequence similarity

1. Introduction

Information retrieval involves finding material (e.g., documents) of an unstructured nature (e.g., text), that satisfies an information need from within large collections [1]. Information retrieval systems generally consist of an index of documents and a query provided by the user [2]. Information retrieval systems should rank documents by their relevance after processing the documents. When the information retrieval system receives the query from the user, a system aims to provide documents from within the collection that are relevant to an arbitrary user information need [1].

Sentence retrieval is similar to document retrieval but with smaller unit of retrieval [3]. Sentence retrieval is defined as the task of acquiring relevant sentences as a response to a query, question, or reference sentence [2]. It can be used in various ways to simplify the end user task of finding the right information from document collections [4]. One of the first and most successful methods for sentence retrieval is the term frequency—inverse sentence frequency (TF-ISF) method, which is an adaptation of the term frequency—inverse document frequency (TF-IDF) method to sentence retrieval [5,3]. Also, BM25 and language modeling-based methods are used for sentence retrieval where the sentence is the unit of retrieval [6].

In this paper, we thoroughly tested the effect of partial matching of terms to sentence retrieval. Text matching was the basis for each natural language processing task [7,8].

We tested the TF-ISF, BM25, and language modeling-based method with sequence similarity presented as the partial matching of words.

For the testing and evaluation of new methods, data of the Text Retrieval Conference (TREC) novelty tracks [9–11] were used as a standard test collection for the sentence retrieval methods.

Many different information retrieval methods are used for sentence retrieval. These methods are always document retrieval methods which are adapted for sentences. Contrary to document retrieval, when implementing sentence retrieval, no processing is implemented that allows the non-exact or partial matching of words. We think that taking the partial matching of words into account has a great potential to improve sentence retrieval, especially when taking into account how little information a sentence contains and that every clue about the relevance of the sentence can be precious. The remainder of this paper is organized as follows. Previous work is shown in Section 2. The research objectives are presented in Section 3. New methods, experiments, and results are presented in Sections 4 and 5, respectively, and the conclusion is given in Section 6.

2. Previous Research

Sentence retrieval is similar to document retrieval, and sentence retrieval methods are usually simple adaptations of document retrieval methods where sentences are treated as documents [6,3]. The sentence retrieval task consists of finding relevant sentences from a document base which has been given a query [6].

Generally, when it comes to information retrieval, the TF-IDF method is still very much present today. For example, the authors of [12] presented a text document search system on distributed high-performance information systems, where initial document weighting was performed using the TF-IDF method. The weighting of text by the TF-IDF method, or the assignment of weight to each linguistic concept in comments from social networks, has also been described by the authors of [13].

The authors of [14] presented two different techniques (BM25 and TF-IDF) to extract the keywords from data collection using Twitter. TF-IDF has also been also used for novelty detection in news events [15]. TF-IDF is widely used for text pre-processing and feature engineering [13]. There have also been attempts to outperform TF-IDF. For example, the authors of [16] presented a Phrase-Based document similarity, which effectively outperformed term-based TF-IDF. The authors of [17] proposed a refined TF-IDF method, called TA TF-IDF, for calculating the weights of hot terms. The vector-space model is one of the most commonly used models for documents and sentence retrieval [2]. The authors of [18] proposed a method of analyzing patent texts based on the vector space model, with the features of patent texts being excluded. Using the proposed algorithm, the authors of [19] calculated the distance between document and topics, and then each document was represented as a vector.

When it comes to sentence retrieval, TF-ISF (the sentence retrieval variation of the TF-IDF method) is one of the first and most widely used methods for sentence retrieval [5]. TF–ISF has been shown to outperform other methods, like BM25-based methods and methods based on language modeling [5,20]. The sentences are represented as a collection of unique words with the weights of words that appear in the selected sentence [21]. The second method popularly used with sentence retrieval is query probability [2,6]. There have been multiple attempts to improve the standard TF-ISF method, which include analyzing the context in the form of a document and the previous, following, or current sentences [6]. The authors of [22] analyzed the effectiveness of contextual information for answer sentence selection.

In our research, we adapted tree standard retrieval models, TF-ISF, BM25 and the language modeling-based method, to improve sentence retrieval using the partial matching of words.

3. Research Objective

The aim of the research was to determine whether and to what extent the partial matching of words (terms between the query q and the sentence s) influences the performance of methods for

sentence retrieval. The influence of partial matching of words were presented by experimental results on three ranking models: TF-ISF, BM25 and the language modeling-based method. Sequence similarity is a technique which allows matching of terms that are similar but not identical. By testing sequence similarity, we intend to add some weight to the question of whether or not it is generally profitable to use partial matching of terms for sentence retrieval.

In large, our research hypothesis was that the partial matching of words improves of sentence retrieval methods.

4. Partial Match of Terms Using Sequence Similarity

The bulk of sentence retrieval methods proposed in the literature are adaptations of standard retrieval models, such as TF-IDF, BM25, the language modeling-based method, where the sentence is the unit of retrieval [6].

In this work, we showed sentence retrieval using sequence similarity and presented the experimental results on three ranking models: TF-ISF (based on vector space model), BM25, and the language modeling-based methods. All three models were adapted in such a way to allow us to test the partial matching of terms.

4.1. TF-ISF Method with Sentence Retrieval

One of the first and most successful methods for sentence retrieval is the TF-ISF defined as [5,4,23]:

$$\text{TF-ISF}(s,q) = \sum_{t \in q} \log\,(tf_{t,q} + 1) \log\,(tf_{t,s} + 1)\, \log\,(\frac{n+1}{0.5 + \ sf_t}) \tag{1}$$

The TF-ISF method assess relevance of sentence s with regard to query q, where

- sf_t is the number of sentences in which the term t appears,
- n is the number of sentences in the collection,
- $tf_{t,q}$ is number of appearances of the term t in a query q, and
- $tf_{t,s}$ is number of appearances of the term t in a sentence s.

4.2. BM25 Model with Sentence Retrieval

The BM25 model uses document ranking, and this model can also be used for sentence retrieval. The ranking function of the BM25 method used to sentence retrieval is defined as [6]:

$$\text{BM25}(s,q) = \sum_{t \in q} log \frac{N - sf(t) + 0.5}{sf(t) + 0.5} \cdot \frac{(k_1 + 1)c(t,s)}{k1\left[(1-b) + b\frac{|s|}{avsl}\right] + c(t,s)} \cdot \frac{(k_3 + 1)c(t,q)}{k_3 + c(t,q)} \tag{2}$$

where

- N is the number of sentences in the collection,
- $sf(t)$ is the number of sentences in which the term t appears,
- $c(t,s)$ is the number of appearances of the term t in a sentence s,
- $c(t,q)$ is the number of appearances of the term t in a query q,
- $|s|$ is the sentence length,
- $avsl$ is the average sentence length, and
- k_1, k_3, and b are the adjustment parameters.

4.3. Sentence Retrieval with Language Model (LM)

The language modeling-based method (language mode) for document retrieval can be applied analogously to sentence retrieval. The probability of a query q given the sentence s can then be estimated using the standard LM approach [6]:

$$\mathrm{LM}(q,s) = \prod_{t \in q} P(t|s)^{c(t,q)} \tag{3}$$

$$P(t|s) = \frac{c(t,s)}{|s|} \tag{4}$$

where

- $c(t,s)$ is number of appearances of the term t in a sentence s, and
- $|s|$ is the sentence length.

One of the most commonly used methods when it comes to sentence retrieval is Dirichlet smoothing, which, when applied to Method (3), gives:

$$\mathrm{LM}(q,s) = \prod_{t \in q} \frac{c(t,s) + \mu P(t)}{|s| + \mu} \tag{5}$$

where

- $c(t,s)$ is the number of appearances of the term t in a sentence s,
- $|s|$ is the sentence length,
- μ is the parameter that control the amount of smoothing, and
- $P(t)$ can be calculated using the maximum likelihood estimator of the term in a large collection: $p(t,C)$ (where C is the collection) [6].

In the previous literature, previously presented ranking functions were combined with data pre-processing and stop word removal. Removing stop words is generally considered to be useful, since stop words do not contain any information [24].

There are several methods for removing stop words, which have been presented by the authors of [25]. Some papers [26] have also proposed time efficient methods. The method we used in this paper is based on a previously compiled list of words. The performance of functions was tested in its basic form with stop word removal.

4.4. Sentence Retrieval Using Sequence Similarity

When it comes to the effect of sequence similarity on sentence retrieval, we made use of the contextual similarity functions presented by the authors of [27]. This procedure enabled us to match of terms that were not identical but similar. Through analyzing the equation for the assessment of the common context by the authors of [27], we concluded that the same analogy could be used to find out if a certain term from query q and a certain term from sentence s share a common subsequence.

In Reference [27], the formula $\delta(N,M)$ determines the appearance of subsequence N in a sequence M. We can define the formula $\delta(q,s)$ analogous to the formula from Reference [27], with query q (instead of subsequence in [27]) and sentence s (instead of sequence M in [27]) as follows:

$$\delta(q,s) = \sum_{i=1}^{|q|} \sum_{j=i}^{|q|} \left| q_{ij} \cap s \right| \tag{6}$$

where

- q_{ij} presents subsequence of sequence s.

If sequence s does not contain the subsequence q_{ij}, there is no need to check for $qi(j+1)$ [27]. When the sequence q is a subsequence of s, the $\delta(q,s) = 1$.

Furthermore, we have to extend Formula (6) to include the normalization parameter T_σ that is given in work [27]. This solves the measurement problem that appears when larger sequences that have more subsequences are more similar than the sequences with shorter lengths, which is not correct [27]. In other words, T_σ is the coefficient which represents the total score that can be achieved

under the assumption that the first sequence is a proper subsequence of the second, or N ⊆ M [27]. After including the normalization parameter T_σ, we obtained:

$$\delta(q,s) = \frac{1}{T_\sigma} \sum_{i=1}^{|q|} \sum_{j=i}^{|q|} \left| q_{ij} \cap s \right| \tag{7}$$

where

- T_σ is the normalization parameter.

The importance of a word depends on its frequency inside its sentence [28] or the number of times a particular word occurs in the sentence [29]. However, the exact matching of terms was used, and the partial matching where words are similar but not identical was not considered. The assumption is that instead of the total matching, the existing methods for sentence retrieval can be improved through partial matching between subsequences and sequences, or in other words, between the term from the query and the term from the sentence.

In Equations (1), (2), and (5), $tf_{t,s}$ and $c(t,s)$ represent the number of appearances of term t in the sentence s. Here, too, the exact matching of terms was used.

If instead of using the parameter $tf_{t,s}$ in the method TF-ISF(s,q), and $c(t,s)$ in method BM25(s,q) and LM(q,s), we define the parameters $tf_{t,s\,(partial)}$ and $c(t,s)_{(partial)}$ which are also defined using the method $\delta(q,s)$, we can define $tf_{t,s\,(partial)}$ and $c(t,s)_{(partial)}$ parameters as follows:

$$sim(t,s)_{partial} = \frac{1}{T_\sigma} \sum_{i=1}^{|t|} \sum_{j=i}^{|s|} \left| t_{ij} \cap s \right| \tag{8}$$

where

- t_{ij} presents subsequence of sequence s (term t from the query as a subsequence of term from the sentence s as a sequence.

To include the partial matching of terms, we took all defined ranking functions and replaced $tf_{t,s}$ and $c(t,s)$ with $sim(t,s)_{partial}$, and defined a new ranking functions for all three ranking models.

We further optimized all of the formulas to only consider terms that already appeared in both the query and sentence or $t \in q \cap s$. The assumption is that if we have a minimum of one match between the query and the sentence, it is more probable that additional matches in other terms between the query and the sentence could be found using the sequence similarity.

New ranking functions have been defined in their final form, which assesses the relevancy of the sentences regarding the query q, and considers the partial match of the term t from the query in relation to the terms from the sentence (Equations (9), (10), and (11)):

$$\text{TF-ISF}_{partial_{(t,s)}}(s,q) = \sum_{t \in q \cap s} \log\left(tf_{t,q} + 1\right) \log\left(sim(t,s)_{partial} + 1\right) \log\left(\frac{n+1}{0.5 + sf_t}\right) \tag{9}$$

$$\text{BM25}_{partial_{(t,s)}}(s,q) = \sum_{t \in q \cap s} log \frac{N - sf(t) + 0{,}5}{sf(t) + 0{,}5} \cdot \frac{(k_1 + 1)\, sim(t,s)_{partial}}{k1\left[(1-b) + b\frac{|s|}{avsl}\right] + sim(t,s)_{partial}} \cdot \frac{(k_3 + 1)c(t,q)}{k_3 + c(t,q)} \tag{10}$$

$$\text{LM}_{partial_{(t,s)}}(q,s) = \prod_{t \in q \cap s} \frac{sim(t,s)_{partial} + \mu P(t)}{|s| + \mu} \tag{11}$$

where

- $t \in q \cap s$ is the postulate that only terms that are in the query and in the sentence are considered. In this case, there was a minimum of at least one match of the terms from the query and from the sentence.

To repeat the point of the new ranking functions, it considers the total match of the query term and sentence term, as defined by the parameter $tf_{t,s}$ and $c(t,s)$ in the ranking functions TF-ISF (s,q), BM25(s,q), and LM(q,s).

The new ranking functions also consider additional appearances of terms from the query as the subsequence of terms from the sentence.

We denoted the new methods and their ranking function as shown in the Table 1.

Table 1. Overview of all sentence retrieval methods tested in this paper.

Method	Ranking Function
TF-ISF	TF-ISF(s,q)
TF-ISF$_{part}$	TF-ISF$_{partial_{(t,s)}}(s,q)$
BM25	BM25(s,q)
BM25$_{part}$	BM25$_{partial_{(t,s)}}(s,q)$
LM	LM(q,s)
LM$_{part}$	LM$_{partial_{(t,s)}}(q,s)$

5. Experiments and Results

The experiment was conducted using data from the novelty tracks of the Text Retrieval Conference (TREC). There were three TREC novelty tracks in the years from 2002 to 2004: TREC 2002, TREC 2003, and TREC 2004 [9–11]. The task was novelty detection, which consists of two subtasks: Finding relevant sentences and finding novel sentences. Our experiment was entirely focused on sentence retrieval, which represents the first task of novelty detection. Three data collections were used, each consisting of 50 topics (queries) and 25 documents per topic, with multiple sentences (Table 2) [30].

Table 2. Description of dataset characteristics.

Name of the Collection	Number of Topics (Queries)	Number of Documents per Topic	Number of Sentences
TREC 2002	50	25	57,792
TREC 2003	50	25	39,820
TREC 2004	50	25	52,447

When it comes to TREC topics, it must be emphasized that each one has a *title*, a *description*, and *narrative*. These three parts represent three versions of the same query but with different lengths. The *title* is the shortest query and *narrative* the longest. In our tests, we used the shortest version called "*title*." Figure 1 depicts the *title* of topic N1 from TREC 2003.

```
<top>
<num>Number: N1
<title>partial birth abortion ban
<toptype>opinion

<desc>Description:
Find opinions about the proposed ban on partial birth abortions.

<narr>Narrative: Relevant information includes opinions on partial
birth abortion and whether or not it should be legal.  Opinions that
also cover abortion in general are relevant.  Opinions on the
implications of proposed bans on partial birth abortions and the
positions of the courts are also relevant.
```

Figure 1. Example of query from Text Retrieval Conference (TREC) 2003 novelty track.

Every query was executed on 25 documents. Each of the documents consisted of multiple sentences as described in Table 2. Each sentence was marked with a beginning and ending tag. Each

sentence had a number. Figure 2 shows an extract form document NYT19980629.0465 with sentence number 11.

```
<Sentence>
  <Name>NYT19980629.0465_11.txt</Name>
  <Content> Abortion-rights activists say the law is worded so vaguely that it
  could reasonably be interpreted to ban any number of abortion methods in even
  the first three months of a pregnancy, when a woman currently has an absolute
  right to seek an abortion.</Content>
</Sentence>
```

Figure 2. Example of the sentence part within the document from TREC 2003 novelty track.

Each TREC data collection also contains a list of relevant sentences. Figure 3 depicts an excerpt from the list of relevant sentences of TREC 2003.

The marked line in Figure 3 defines sentence 11 from the document NYT19980629.0465 as relevant to the topic (query) N1 ("partial birth abortion ban").

```
N1 NYT19980629.0465:8
N1 NYT19980629.0465:11
N1 NYT19980629.0465:12
N1 NYT19980629.0465:13
N1 NYT19980629.0465:14
N1 NYT19980629.0465:15
N1 NYT19980629.0465:18
N1 NYT19980629.0465:24
N1 NYT19980629.0465:33
N1 NYT19980629.0465:34
N1 NYT19980629.0465:36
N1 NYT19980629.0465:37
N1 NYT19980629.0465:42
N1 NYT19980629.0465:43
```

Figure 3. An excerpt in the dataset that contains a list of relevant sentences.

To test whether the new methods provide better sentence retrieval results than the existing methods, we compared the performances of the new methods in relation to the existing methods (baseline methods) using the following standard measures: P@10, the MAP, and the *R-precision* [1,6].

The precision at x or $P@x$ can be defined as:

$$P@x(q_j) = \frac{\text{number of relevant sentences within top } x \text{ retrieved}}{x} \quad (12)$$

The P@10 values shown in this paper refer to average P@10 for 50 queries [30].

R-precision can be defined as [1]:

$$R - precision = \frac{r}{|Rel|} \quad (13)$$

where

- $|Rel|$ is the number of relevant sentences to the query, and
- R is the number of relevant sentences in top $|Rel|$ sentences of the result.

The *R-precision* values shown in this paper are (analogous to P@10) averages for 50 queries [30].

The Mean Average Precision and *R-precision* gave similar results and were used to test high recall. High recall means it is more important to find all of the relevant sentences, even if it means searching through many sentences including many that are nonrelevant. Meanwhile, P@10 is used for testing precision [30].

In terms of information retrieval, precision means it is more important to get only relevant sentences than finding all of the relevant sentence [30].

To compare the difference between methods, we used a two-tailed paired *t*-test with a significance level of $\alpha = 0.05$ [4]. The results of our tests are presented in tabular form.

Statistically significant differences in relation to the baseline methods are marked with a (*).

In each of the tested methods, we used stop word removal as pre-processing step.

Table 3 shows the results of our tests on two different versions of the $TF - ISF$ method using TRECs 2002, 2003, and 2004.

Table 3. Test results of methods TF-ISF and TF-ISF$_{part}$.

Data Collection	Measures	TF-ISF	TF-ISF$_{part}$
TREC 2002	P@10	0.304	0.32
	MAP	0.196	* 0.204
	R-prec.	0.245	0.250
TREC 2003	P@10	0.692	0.714
	MAP	0.576	* 0.591
	R-prec.	0.547	* 0.560
TREC 2004	P@10	0.434	0.468
	MAP	0.324	* 0.335
	R-prec.	0.336	* 0.355

* Statistically significant differences in relation to the baseline methods

Table 3 shows that the $TF - ISF_{part}$ methods provided better results and statistically significant differences in relation to the base $TF - ISF$ method when the MAP measures was used in all three collections, and R-prec. in TREC 2003 and TREC 2004 collection. Better results were not achieved when using the P@10 measures.

Table 4 shows the results of our tests on two different versions of the $BM25$ model using TRECs 2002, 2003, and 2004.

The parameters settings for the $BM25$ model were $k1 = 1.5$, $b = 0.75$, $k3 = 0$.

Table 4. Test results of methods BM25 and BM25$_{part}$ ($k1 = 1.5$, $b = 0.75$, $k3 = 0$).

Data Collection	Measures	BM	BM$_{part}$
TREC 2002	P@10	0.142	* 0.33
	MAP	0.105	* 0.209
	R-prec.	0.097	* 0.255
TREC 2003	P@10	0.628	* 0.75
	MAP	0.464	* 0.601
	R-prec.	0.4281	* 0.565
TREC 2004	P@10	0.366	* 0.472
	MAP	0.242	* 0.342
	R-prec.	0.236	* 0.363

* Statistically significant differences in relation to the baseline methods

Table 4 shows that the $BM25_{part}$ method (method using sequence similarity) provided better results and statistically significant differences in relation to the base method in all measures and collections.

Table 5 shows the results of our tests on two different version of the LM model using TREC collections from 2002, 2003, and 2004. The parameter settings for LM were $\mu = 100$.

Table 5. Test results of methods LM and LM$_{part}$ ($\mu = 100$).

Data Collection	Measures	LM	LM$_{part}$
TREC 2002	P@10	0.268	* 0.356
	MAP	0.170	* 0.207

	R-prec.	0.215	* 0.250
TREC 2003	P@10	0.71	0.7
	MAP	0.528	* 0.597
	R-prec.	0.501	* 0.567
TREC 2004	P@10	0.388	* 0.458
	MAP	0.287	* 0.334
	R-prec.	0.306	* 0.355

* Statistically significant differences in relation to the baseline methods

Table 5 shows that the LM_{part} method, as well the $\mathrm{BM25}_{part}$ method, provided better results and statistically significant differences in relation to the base method in all measures and collections, except for the P@10 measures for TREC 2003 collection.

6. Conclusions

In this paper, we thoroughly tested sentence retrieval with sequence similarity, which allowed us to match words that were similar but not identical. We tested sentence retrieval methods with the partial matching of terms using TREC data. We adapted TF-ISF, BM25 and the language modeling-based method to test the partial matching of terms using sequence similarity.

We found out that the partial matching of terms using sequence similarity can benefit sentence retrieval in all three tested collection. We showed the benefits of partial matching using sequence similarity through statistically significant better results.

The reason for the better position of the sentence when we used adapted methods using sequence similarity is the additional matching of terms between the query q and the sentence s.

We conclude that partial matching of words is beneficial when combined with sentence retrieval. However, we did not analyze whether some nonrelevant sentences were falsely high ranked. Therefore, future research will include a thorough analyses of the effect of the partial matching of words on sentence retrieval. Future research will also include experiments using pre-processing methods, such as stemming and lemmatization or some other technique.

Author Contributions: Conceptualization, I.B. and A.D.; methodology, I.B. and A.D. and S.G.; software, I.B. and A.D.; validation, A.D. and S.G.; formal analysis, I.B.; investigation, I.B.; resources, I.B. and A.D.; data curation, I.B. and A.D.; writing—original draft preparation, I.B.; writing—review and editing, I.B. and A.D.; visualization, I.B.; supervision, S.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Manning, C.D.; Raghavan, P.; Schutze, H. *Introduction to Information Retrieval*; Cambridge University Press: Cambridge, UK, 2008.
2. Murdock, V. G. Aspects of Sentence Retrieval. Ph.D. Thesis, University of Massachussetts, Amherst, MA, USA, 2006.
3. Doko, A.; Štula, M.; Seric, L. Using TF-ISF with Local Context to Generate an Owl Document Representation for Sentence Retrieval. *Comput. Sci. Eng. Int. J.* **2015**, *5*, 1–15.
4. Doko, A.; Štula, M.; Seric, L. Improved sentence retrieval using local context and sentence length. *Inf. Process. Manag.* **2013**, *49*, 1301–1312.
5. Allan, J.; Wade, C.; Bolivar, A. Retrieval and novelty detection at the sentence level. In Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval -SIGIR '03., Toronto, Canada, 28 July to 1 August 2003.
6. Fernández, R.T.; Losada, D.E.; Azzopardi, L. Extending the language modeling framework for sentence retrieval to include local context. *Inf. Retr.* **2010**, *14*, 355–389.
7. Agarwal, B.; Ramampiaro, H.; Langseth, H.; Ruocco, M. A deep network model for paraphrase detection in short text messages. *Inf. Process. Manag.* **2018**, *54*, 922–937.

8. Kenter, T.; de Rijke, M. Short Text Similarity with Word Embeddings. In Proceedings of the 24th ACM International on Conference on Information and Knowledge Management—CIKM '15., Melbourne, Australia, October 2015.
9. Harman, D. Overview of the TREC 2002 novelty track. In Proceedings of the Eleventh Text Retrieval Conference (TREC), Gaithersburg, Maryland, 19–22 November 2002.
10. Soboroff, I.; Harman, D. Overview of the TREC 2003 novelty track. In Proceedings of the Twelfth Text Retrieval Conference (TREC), Gaithersburg, Maryland, November 2003.
11. Soboroff, I. Overview of the TREC 2004 novelty track. In Proceedings of the Thirteenth Text Retrieval Conference (TREC), Gaithersburg, Maryland, 16-19 November 2004.
12. Chiranjeevi, H.; Manjula, K.S. An Text Document Retrieval System for University Support Service on a High Performance Distributed Information System. In Proceedings of the 2019 IEEE 4th International Conference on Cloud Computing and Big Data Analysis (ICCCBDA), Chengdu, China, 12–15 April 2019.
13. Yahav, I.; Shehory, O.; Schwartz, D.G. Comments Mining With TF-IDF: The Inherent Bias and Its Removal. *IEEE Trans. Knowl. Data Eng.* **2018**, *31*, 437–450.
14. Kadhim, A.I. Term Weighting for Feature Extraction on Twitter: A Comparison between BM25 and TF-IDF. In Proceedings of the 2019 International Conference on Advanced Science and Engineering (ICOASE), Zakho-Duhok, Iraq, 2–4 April 2019.
15. Fu, X.; Ch'Ng, E.; Aickelin, U.; Zhang, L. An Improved System for Sentence-level Novelty Detection in Textual Streams. *SSRN Electron. J.* **2015**, doi:10.2139/ssrn.2828008.
16. Niyigena, P.; Zuping, Z.; Khuhro, M.A.; Hanyurwimfura, D. Efficient Document Similarity Detection Using Weighted Phrase Indexing. *Int. J. Multimedia Ubiquitous Eng.* **2016**, *11*, 231–244.
17. Zhu, Z.; Liang, J.; Li, D.; Yu, H.; Liu, G. Hot Topic Detection Based on a Refined TF-IDF Algorithm. *IEEE Access* **2019**, *7*, 26996–27007.
18. Lei, L.; Qi, J.; Zheng, K. Patent Analytics Based on Feature Vector Space Model: A Case of IoT. *IEEE Access* **2019**, *7*, 45705–45715.
19. Xue, M. A Text Retrieval Algorithm Based on the Hybrid LDA and Word2Vec Model. In Proceedings of the 2019 International Conference on Intelligent Transportation, Big Data & Smart City (ICITBS), Changsha, China, 12–13 January 2019.
20. Losada, D.E.; Fernández, R.T. Highly frequent terms and sentence retrieval. In *International Symposium on String Processing and Information Retrieval*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 217–228.
21. Sharaff, A.; Shrawgi, H.; Arora, P.; Verma, A. Document Summarization by Agglomerative nested clustering approach. In Proceedings of the 2016 IEEE International Conference on Advances in Electronics, Communication and Computer Technology (ICAECCT), Pune, India, 2–3 December 2016.
22. Tan, C.; Wei, F.; Zhou, Q.; Yang, N.; Du, B.; Lv, W.; Zhou, M. Context-Aware Answer Sentence Selection with Hierarchical Gated Recurrent Neural Networks. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2018**, *26*, 540–549.
23. Losada, D.E. Statistical query expansion for sentence retrieval and its effects on weak and strong queries. *Inf. Retr.* **2010**, *13*, 485–506.
24. Srividhya, V.; Anitha, R. (2010). Evaluating preprocessing techniques in text categorization. *J. Comput. Sci. Appl.*, **2010**, *47*, 49-51.
25. Vijayarani, S.; Ilamathi, M.J.; & Nithya, M. Preprocessing techniques for text mining-an overview. *Int. J. Comput. Sci. Commun. Netw.* **2015**, *5*, 7–16.
26. Behera, S. Implementation of a Finite State Automaton to Recognize and Remove Stop Words in English Text on its Retrieval. In Proceedings of the 2018 2nd International Conference on Trends in Electronics and Informatics (ICOEI), Tirunelveli, India, 11–12 May 2018.
27. Karic, I.; Vejzovic, Z. Contextual Similarity: Quasilinear-Time Search and Comparison for Sequential Data. In Proceedings of the 2017 International Conference on Control, Artificial Intelligence, Robotics & Optimization (ICCAIRO), Prague, Czech Republic, 20–22 May 2017.
28. Singh, J.; Singh, G.; Singh, R.; Singh, P. Morphological evaluation and sentiment analysis of Punjabi text using deep learning classification. *J. King Saud Univ.—Comput. Inf. Sci.* **2018**, doi:10.1016/j.jksuci.2018.04.003.

29. Gupta, S.; Gupta, S.K. A Hybrid Approach to Single Document Extractive Summarization. *Int. J. Comput. Sci. Mob. Comput.* **2018**, *7*, 142–149.
30. Boban, I.; Doko, A.; Gotovac, S. Sentence Retrieval using Stemming and Lemmatization with Different Length of the Queries. *Adv. Sci. Technol. Eng. Syst. J.* **2020**, *5*, 349–354.