Annals of Emerging Technologies in Computing (AETiC)
Vol. 7, No. 1, 2023

Research Article

Similarity Detection of Time-Sensitive Online News Articles Based on RSS Feeds and Contextual Data

Mohammad Daoud

American University of Madaba, Jordan
m.daoud@aum.edu.jo

Received: 15th June 2022; Accepted: 19th September 2022; Published: 1st January 2023

Abstract: This article tackles the problem of finding similarity between web time-sensitive news articles, which can be a challenge. This challenge was approached with a novel methodology that uses supervised learning algorithms with carefully selected features (Semantic, Lexical and Temporal features (content and contextual features)). The proposed approach considers not only the textual content, which is a well-studied approach that may yield misleading results, but also the context, community engagement, and community-deduced importance of that news article. This paper details the major procedures of title pair pre-processing, analysis of lexical units, feature engineering, and similarity measures. Thousands of web articles are being published every second, and therefore, it is essential to determine the similarity of these articles efficiently without wasting time on unnecessary text processing of the bodies. Hence, the proposed approach focuses on short contents (titles) and context. The conducted experiment showed high precision and accuracy on a Really Simple Syndication (RSS) dataset of 8000 Arabic news article pairs collected automatically from 10 different news sources. The proposed approach achieved an accuracy of 0.81. Contextual features increased the accuracy and the precision. The proposed algorithm achieved a 0.89 correlation with the evaluations of two human judges based on Pearson's Correlation Coefficient. The results outperform the state-of-the-art systems on Arabic news articles.

Keywords: ***Arabic NLP; News aggregators; Recommendation systems; Semantic similarity; Web personalization***

1. Introduction

Constantly, time-sensitive news updates are being published on the web [1]. When using intelligent web feeds (news feeds), it is desirable to link new updates (time-sensitive news articles) with other updates. This will help personalize the web services [2] and improve the relevance and quality of the news feed. Besides, It is an essential necessity for many important modern applications to find a similarity between textual online components (including online news articles) such as text searching [3], news aggregators [4-5], content platforms and news recommendation systems [4-6]. That was the focus of many research projects, where similarity was tackled as a text distance or classification, giving more importance to the textual content of articles or web documents in general [7-8]. Detecting similarity between various linguistic utterances is a crucial computational linguistic task [9] whether the similarity is measured between characters, lexical units, expressions, phrases, paragraphs or documents. Many research projects show a significant improvement in performance when accurate similarity detection is used, such as Search Engines and information retrieval (IR) [10], recommendation systems [11-12], machine translation (MT) [13], Chatbots [14], text clustering and classification [15], sentiment analysis and opinion mining [16-17], etc.

Text similarity was studied by researchers from various points of view. Some methods suggest that two textual items (utterances) are similar if they have a common subsequence of characters (string) and words. For example, algorithms such as Jaccard similarity [11] and cosine similarity are very effective and

Mohammad Daoud, "Similarity Detection of Time-Sensitive Online News Articles Based on RSS Feeds and Contextual Data", *Annals of Emerging Technologies in Computing (AETiC)*, Print ISSN: 2516-0281, Online ISSN: 2516-029X, pp. 75-88, Vol. 7, No. 1, 1st January 2023, Published by International Association for Educators and Researchers (IAER), DOI: 10.33166/AETiC.2023.01.006, Available: http://aetic.theiaer.org/archive/v7/v7n1/p6.html.

efficient in measuring similarity between utterances based on the common strings between them. On the other hand, semantic similarity targets discovering logical similarities between textual items even if the lexical similarity does not exist [18].

In this paper, the focus is on the similarity between Arabic web news articles, which is not only a text similarity task. Web articles have contextual aspects that can be used. Besides, relying on text only can be misleading. For example, two authors may write about the same news story but use distant textual elements, so the two articles will be considered as dissimilar. Meanwhile, a story of a Christmas event in 2016 (or some other periodical events) might be textually similar to a Christmas story in 2018, but in reality, there is no point in grouping them together. In this article, content similarity and contextual similarity of web news articles were considered. Content similarity is a measure based on the textual content, whether it is lexical or semantic. Content similarity was applied on titles and keywords rather than the full article. Context similarity relies on metadata and the impact of the article, particularly the publication date, importance of the article, and social media impact.

Measuring similarity between Arabic utterances can be a challenging task. Many researchers consider Arabic as a pi-language (poorly informatized language) [19-20] and maintaining high quality lexical and semantic data from its corpus is difficult [21]. Similarity experiments in resourceful languages achieve better results in general [22]. Temporal and contextual features will be beneficial, especially without the needed lexical and semantic resources. In this paper, the focus is on seeking a hybrid approach that utilizes supervised learning on Semantic, Lexical and Temporal features. The idea is to use social media interaction and engagement with a web article as a feature that can determine its similarity to other articles.

This paper is organized as follows: the next section discusses related work. After that, in section three, the main approach to article similarity detection is introduced. Section four presents the approach to data acquisition and preprocessing. And then section five shows the experiment and its results, while section six evaluates and assesses the proposed method. And finally, some conclusions, future work, and possible applications are shown.

2. Related work

Relating news articles were studied as a text classification problem, a document clustering problem, or a content similarity problem. Time-sensitive data may introduce new (unseen) topics, classes, community interest, and terminology (lexical units); therefore, this will be a challenge for classifications and clustering. The content similarity of news articles relies on content overlapping. Many IR approaches investigate textual and multimedia overlapping. A promising line of research focuses on the layout, visualization, and structure of articles to deduce similarity [23], but this requires huge computational capabilities and resources. It is desired to find similarity without heavy processing of the article's content and structure. Thus, the proposed approach relies on finding similarity based on RSS feeds that do not show the final visualization and the article's content. Therefore, it is considering short metadata such as titles, publication dates, and times.

Many researchers approach the problem of article similarity as a text similarity problem. Textual similarity [24] depends on the string representation of phrases. And consequently, two documents are similar if they have similar strings.

Some approaches deal with the text as a sequence of characters (String), such as Longest Common Subsequence [25], Jaro [26], Damerau - Levenshtein [27] and Needleman – Wunsch [28]. And other approaches consider the text as words governed by syntaxes such as Block Distance, Cosine similarity [29], Dice's coefficient [30], Euclidean distance (L2), and Jaccard similarity [31]. Text similarity can be effective and easy to implement, but word ambiguity is challenging for this approach. On the other hand, semantic similarity may resolve ambiguity [32], but requires significant resources. Semantic similarity can be used to solve word ambiguity [32]. It tries to correlate different lexical units (and sometimes other utterances) based on their logical (meaning) similarity rather than their character or lexical similarity. It is very common for semantic similarity algorithms to rely on a large textual corpus to deduce additional information about the words and phrases. For instance, finding similar words based on their frequent collocation. The following algorithms and methods are considered as text corpus semantic similarity algorithms: Hyperspace Analogue to Language (HAL) [33], Latent Semantic Analysis (LSA) [34], Generalized Latent Semantic Analysis (GLSA) [35], Explicit Semantic Analysis (ESA) [36], Pointwise Mutual Information - Information Retrieval (PMI-IR) [37], Second-order co-occurrence point wise mutual information (SCO-PMI) [38], Normalized Google Distance (NGD) [39] and Extracting DIStributionally similar words using COoccurrences (DISCO) [40]. For these algorithms to work accurately and effectively, a vast and clean textual corpus must be used, and they deduce similarity based on textual collocations. Normalized Google Distance (NGD) can be considered as an exception to that last condition, because it utilizes the huge capabilities of Google's indexed data to our advantage. Therefore, it can deduce similarity without building a local data repository.

Usually, a semantic network such as Wordnet [41] can be attached to the semantic similarity engine to increase its accuracy and coverage. In fact, many researchers are using Wordnet comprehensively to measure the semantic distances between words and phrases, which can be considered an autonomous semantic similarity measure. This can be useful for resourceful languages such as English (English Wordnet has 155 327 words organized in 175 979 synsets). Still, it is not as effective for The Arabic language where the synsets count is significantly lower than the number of English synsets.

Some researchers consider other aspects, such as URL linking structures, visual location (location on a web page), metadata and users. But those methods are media and applications dependent.

Text and semantic similarity can be used for the article's title similarity, but they are not enough. Similar articles may describe the exact same event using completely different texts.

The proposed method in this paper represents a hybrid approach that utilizes content (string) similarity of the titles and contextual similarity, which utilizes the engagement with the community and the importance of the content.

3. Article Comparison

3.1. Motivation and Research Objectives

Finding similar Arabic news articles is very important for news aggregators, search engines, recommendation systems, and many other web applications. Most current approaches invest heavily in lexical and semantic similarities between the news textual bodies. These approaches have three issues: 1) they ignore the contextual elements of a news article, 2) they require heavy processing of entire textual bodies. 3) Textual similarity (lexical or semantic) between news bodies might not reflect relevance or interest by the users.

This article introduces an approach that tries to improve similarity detection between Arabic news articles by employing a novel approach that provides powerful lexical and semantic embedding for every title and considers the metadata of news articles simultaneously.

This paper introduces a novel method to determine the similarity between two Arabic web articles. The proposed algorithm focuses on Content Similarity (lexical and semantic similarity) and Contextual Similarity. This section details the main approach, starting with the main algorithm, pre-processing, and feature engineering.

The proposed algorithm relies on Really Simple Syndication (RSS) data. This is a common data format for sharing new web articles. RSS data rarely contains an entire article body, but it contains metadata, titles, and links to the web article, etc.

The following subsections detail the extraction of content features and context features.

3.2. Content Similarity

This subsection describes the extraction process for content similarity features. Note that not all RSS feeds have a summary or full text. Therefore, only titles are considered in the string-based similarity.

To compare between two articles, a list of lexical and semantic features for the titles of every couple is generated. Next subsection will introduce the lexical similarity features followed by the semantic similarity.

3.2.1. String-based similarity (lexical similarity)

Assume that we have two articles, A1 and A2. We want to find the textual similarity between those two articles through their titles (T1 and T2). We consider several forms of each title, as follows:

$$T1 = \{original1, norm1, bow1, ner1, pos1\} \tag{1}$$

$$T2 = \{original2, norm2, bow2, ner2, pos2\} \tag{2}$$

Where original1 and original1 are the original text of the titles, norm1 and norm2 are the normalized forms of the titles (text after standard normalization), bow1 and bow2 are the bag of words of the normalized titles, ner1 and ner2 are the named entity in the original titles, pos1 and pos2 are the part of speech of the original titles. The reason for creating these forms is to find distances between each corresponding form. It should be noted that for the NLP pipeline in creating these forms, including named entity recognition (NER) and part of speech (POS) analysis, the Farasa Arabic Java Library [42] was used.

Accordingly, various similarities are measured between the above couples. Forming the similarity features as follows:

1. Similarity between the normalized T1 and T2 according to the Longest Common Subsequence algorithm (effective in measuring common lexical units in the titles). LCS can be obtained using the following equation:

$$LCS\ (i,j) = \begin{cases} \emptyset & if\ i = 0\ or\ j = 0 \\ LCS(i-1, j-1) + 1 & if\ norm1i = norm2j \\ max\{\ LCS(i, j-1), LCS\ (i-1, j) & if\ norm1i \neq norm2j \end{cases} \tag{3}$$

If n is the size of norm1 and m is the size of norm2, then LCS (n, m) is the number of the longest common subsequence between norm1 and norm2.

2. Distance between the normalized T1 and T2 according to Cosine similarity. As obtained by the below equation.

$$Cosine\ (norm1, norm2) = \frac{\overrightarrow{norm1}.\overrightarrow{norm2}}{\left\|\overrightarrow{norm1}\right\| \left\|\overrightarrow{norm2}\right\|} = \frac{\sum_1^n norm1_i norm2_i}{\sqrt{\sum_1^n norm1_i^2}\sqrt{\sum_1^n norm2_i^2}} \tag{4}$$

Where $\overrightarrow{nomr1}.\overrightarrow{norm\ 2}$ is the dot product if the two vectors and n is the length of the larger vector (larger title, in terms of the number of words).

3. Distance between the normalized T1 and T2 according to the Jaccard similarity algorithm. As obtained by the below equation.

$$Jaccard\ (norm1, norm2) = \frac{|nomr1 \cap norm2|}{|nomr1 \cup nomr2|} \tag{5}$$

4. Distance between the normalized T1 and T2 according to their Euclidean distance. As obtained by the below equation:

$$Eucl\ (norm1, norm2) = \sqrt{\sum_{i=1}^n (nomr1_i - nomr2_i)^2} \tag{6}$$

5. Distance between the named entity of T1 and T2 according to Jaccard similarity. As obtained by the equation below.

$$Jaccard\ (ner1, ner2) = \frac{|ner1 \cap ner2|}{|ner1 \cup ner2|} \tag{7}$$

6. Distance between the named entity of T1 and T2 according to Cosine similarity. As obtained by the equation below.

$$Cosine\ (ner1, ner2) = \frac{\overrightarrow{ner1}.\overrightarrow{ner2}}{\left\|\overrightarrow{ner1}\right\| \left\|\overrightarrow{ner2}\right\|} = \frac{\sum_1^n ner1_i ner2_i}{\sqrt{\sum_1^n ner1_i^2}\sqrt{\sum_1^n ner2_i^2}} \tag{8}$$

7. Distance between the part of speech output for T1 and T2 according to Jaccard similarity.

$$Jaccard\ (pos1, pos2) = \frac{|pos1 \cap pos2|}{|pos1 \cup pos2|} \tag{9}$$

8. Distance between the part of speech output for T1 and T2 according to Cosine similarity. As obtained by the equation below.

$$Cosine\ (pos1, pos2) = \frac{\overrightarrow{pos1}.\overrightarrow{pos2}}{\left\|\overrightarrow{pos1}\right\| \left\|\overrightarrow{pos2}\right\|} = \frac{\sum_1^n pos1_i pos2_i}{\sqrt{\sum_1^n pos1_i^2}\sqrt{\sum_1^n pos2_i^2}} \tag{10}$$

These features are normalized to represent the textual and lexical similarity (or distance) between couples of articles.

3.2.2. Semantic Similarity (Normalized Google Distance)

Many options can be used for semantic similarity. It was reported that Normalized Google Distance NGD is very accurate for semantic disambiguation with little to no context, which is convenient for our short titles and keywords [43]. What is powerful about NGD is that it utilizes the prevailing size of Google data to your advantage, even when you do not have a relevant corpus, which is perfect in our problem. As a result, NGD was chosen for semantic similarity. The Normalized Google Distance (NGD) is a semantic similarity measure that can be determined from the number of hits / results returned by the Google search engine for a particular group of search terms. Semantically related words (lexical units) will have close measures of Normalized Google Distance, while words with dissimilar meanings tend to be farther apart.

To be precise, NGD between two terms can be calculated as follows:

$$NGD(x, y) = \frac{max\{log\, f(x), log\, f(y)\} - log\, f(x, y)}{log\, G - min\{log\, f(x), log\, f(ry)\}} \tag{11}$$

Where:

1. x and y are two search terms
2. f (x) is the number of results returned by Google Search Engine for the term x.
3. f (y) is the number of results returned by Google Search Engine for the term y.
4. f (x , y) is the number of results returned when we search Google for x and y together.
5. While G is the total number of pages indexed by Google.

NGD (x , y) will be close to 0 if the terms x and y are related. NGD was used for Arabic news titles because it is practically convenient, computationally efficient and does not require a corpus (not like most of the other semantic similarity algorithms).

Algorithm 1. Shows the steps towards finding NGD similarity.

```
Algorithm 1
NGDSim ( T1 , T2 )
//Start of Algorithm 1
   normt1 = NormalizeTitle ( T1 )
   normt2 = NormalizeTitle ( T2 )
   fx = callgooglesearch ( normt1 )
   fy = callgooglesearch ( normt2  )
   fxy = callgooglesearch ( normt1 + normt2 )
   G = callgooglesearch ( " the " )
   sim = ( max ( log  fx , log fy ) – log fxy ) / log G  – min ( log fx , log fy ) )
 return sim
//end of Algorithm 1
```

Algorithm 1 receives a couple of Arabic titles and returns their NGD similarity. Algorithm 1 estimates the total number of pages indexed by Google using the return number of hits when the keyword "the" is searched. The function *callgooglesearch* (x) returns the number of search results of x.

It should be noted that NGDSim does not use the inclusive double quotations " " on the normalized T1 and T2, which means that fx will equal the number of search results returned by all the terms in normt1 according to their original order in T1.

3.3. Contextual similarity

As mentioned earlier, content alone should not determine similarity, even if a semantic disambiguation mechanism is deployed. Because context is very important in the domain of news articles. Two articles can have high content similarity, but in reality, they should not be grouped together simply because they have different context (temporal, geographical, demographical, etc.).

The following contextual similarity features were introduced to eliminate content bias::

1. Time distance (publication date distance).
2. Community importance (importance-based retweets).
3. Community interest.

3.3.1. Time distance

Time distance is identified as the difference between the publication date of article A1 and the publication date of article A2. The distance could be measured in any date/time unit. For technical convenience, time distance is measured using the number of days. Equation 12 calculates the time distance:

$$TD1 = A1\ (\,pub\,) - A2\ (\,pub\,) \tag{12}$$

Where:

1. TD1 is the new time difference feature
2. A1 (pub) is the publication date of the first article.
3. A2 (pub) is the publication date of the second article.

However, more weight should be given to titles published within a close period. Therefore, equation 13 is presented as follows:

$$TD2 = e^{TD1} \tag{13}$$

Where TD2 is a second time distance that gives close time intervals more importance. And TD1 does not provide any bias towards close time distances. We keep TD1 and TD2 as useful features for the experiment.

3.3.2. Importance

It is anticipated that similar web articles would have comparable community importance; therefore, there is a need for the following simple feature, which measures the difference in the importance of two web articles in terms of the number of retweets:

$$I = \frac{RT1 - RT2}{Max\ (\,RT1\ ,RT2\,)} \tag{14}$$

Where:

1. I is the relative importance feature.
2. RT1 is the number of retweets of the first article.
3. RT2 is the number of retweets of the second article.

Similar articles draw similar attention from the social network community. Therefore, they have a similar number of retweets.

It should be noted that the URL of the article in the RSS feed is being used to find the number of retweets.

3.3.3. Community interest

Similar web articles attract similar audiences. Therefore, the proposed approach measures the similarity between two articles through the similarity in the profiles of the Twitter users who interact with the articles.

Algorithm 2. Measuring community interest

```
Algorithm 2
CI ( A1 ,  A2 )
// Algorithm 2 starts
  P is the list of the retrieved profiles that retweeted A1
  R is the list of the retrieved profiles that retweeted A2
  Foreach profile in P
    Find the last 20 tweets and add them into prt
  Foreach profile in R
    Find the last 20 tweets and add them into rrt
  Community Interest = normalized intersection of prt and rrt
  Return Community interest
//Algorithm 2 ends
```

As illustrated in Algorithm 2 to measure the distance in community interest between A1 and A2, I retrieve all the retweets of A1. Assuming that:

$$p = \{\,p1\ ,\ p2\ ,\ p3\ \ldots\ pn\,\} \tag{15}$$

Where p is the list of Twitter profiles of the people who retweeted Article 1. And assuming that:

$$r\ =\ \{\,r1\ ,\ r2,\ r3\ \ldots\ rn\,\} \tag{16}$$

Where r is the list of Twitter profiles of the people who retweeted Article 2. Now, for each profile in p and r, we retrieve the last 20 retweets. Producing:

$$prt =\{\,prt1\ , prt2\ ,\ prt3 \ldots\ prtn * 20\,\} \tag{17}$$

Which is the set of all retweets from users who retweeted Article 1. And:

$$rrt = \{ rrt1 ,\ rrt2 ,\ rrt3 \ldots\ rrtm * 20 \} \tag{18}$$

Which is the set of all retweets from users who retweeted Article 2.

Therefore, to measure the similarity of Article 1 and Article 2, the intersection between rrt and prt is measured as follows:

$$CI = \frac{prt \cap rrt}{max\ (| \ prt\ | , | \ rrt\ |)} \tag{19}$$

4. Data acquisition and preprocessing

4.1. Data Compilation

The following RSS feeds provided by various Arabic online news agencies were considered. Table 1 shows the RSS feeds that were used for data collection.

Table 1. List of RSS feeds used for the experiment

Main website	Number of RSS feeds
http://arabic.cnn.com	7 feeds
http://aawsat.com	31 feeds
http://www.bbc.com/arabic	10 feeds
https://www.albawaba.com	11 feeds
https://www.youm7.com	17 feeds
https://www.alhurra.com	44 feeds
http://www.rumonline.net	69 feeds
http://alrai.com	41 feeds
https://news.un.org	19 feeds
https://www.arabstoday.net	25 feeds

The feeds were automatically invoked to retrieve the data in XML formats.

Figure 1 shows the process of data collection.

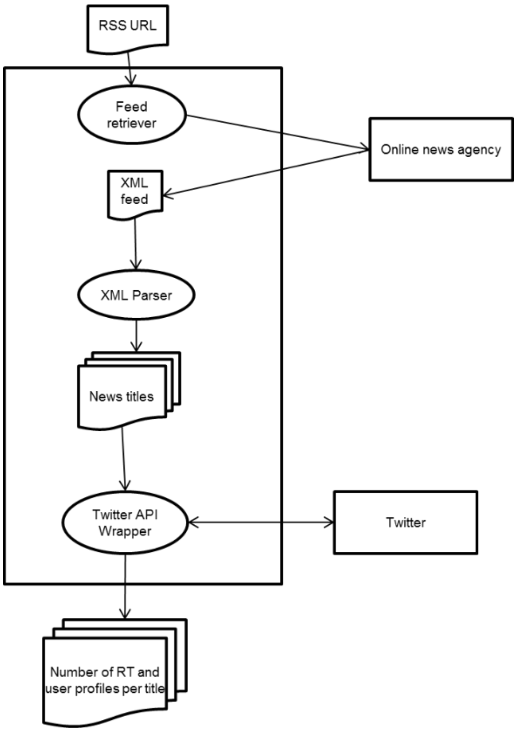

Figure 1. The process of data collection

As shown in figure 1, the feed retriever takes the RSS URLs as an input to retrieve the available XML documents from the news agencies. After that, an XML parser retrieves the raw data from these files, including their titles, dates, etc.

Once the raw data is available, a Twitter API wrapper finds the retweets for every extracted title. And for every title, the system probes the profiles that retweeted it. This is done to calculate the importance and the community interest discussed previously. And therefore, we have the full needed data for every news article, including its number of tweets and the profiles which retweeted it.

4.2. Data Description

The following is a sample XML document retrieved by an Arabic CNN RSS:

```
<item>
  <title>
فيروس كورونا . .  أسواق الأسهم بالصين تشهد أسوأ انخفاض منذ 2015
</title>
  <link>
http://arabic.cnn.com/business/video/2020/02/04/v84621-coronavirus-world-markets
</link>
 <description>
تراجعت الأسهم الصينية ، الاثنين ، وهو أول يوم تمكن فيه
 7.7 % في اليوم   المستثمرون من التفاعل مع انتشار فيروس كورونا منذ أكثر من أسبوع . و كان يومًا سيئًا جدًا في شنغهاي وشنزين . إذ انخفض مؤشر شنغهاي بنسبة
الأول من التداول في أعقاب عطلة طويلة للعام القمري الجديد - وهو أكبر انخفاض له منذ "الاثنين الأسود" في أغسطس/ آب 2015 عندما هزت الأسواق العالمية مخاوف من تباطؤ
النمو في الصين.
</description>
  <pubDate>
Tue, 04 Feb 2020 06:57:06 + 0000
</pubDate>
   <dc:creator>
Abelhaik
</dc:creator>
   <guid isPermaLink = " false ">
84621 at http://arabic.cnn.com
</guid>
   </item>
```

It is common for such XML files to contain a publication date, title, ID, source, URL and a description. For every title, the number of tweets and the list of retweeted profiles to be added to the XML file were generated. All used feeds have similar information. Every data item (article) in the dataset contains at least the following:

- Textual Title
- URL
- Date of Publication

In addition to that the system retrieves the following:

- Number of retweets based on the URL
- Links of profiles that retweeted the URL

4.3. Data Pre-processing

For the experiment, 6000 XML files were collected from the ten news resources mentioned previously. The dates of the articles span from January 2020 to January 2022. The size of the collection is larger than (or comparable to) Arabic and non-Arabic news article similarity experiments conducted based on labelled data [44-47].

The collected files (articles) were used to create a list of 8000 pairs of articles. Each pair is labelled with True or False. True means the couple is actually similar. False means the articles are not similar. Most of the pairs (5850 pairs) were False. While the rest of the pairs were True-labelled. Most of the true-labelled pairs were manually selected, because it is difficult to find organically similar articles in the randomly generated set of pairs.

The 8000 pairs were divided into a train data set and a test data set. For training, 5000 pairs (3500 of class False and 1500 of class True) were selected, which are 62.5% of the whole data set. And for testing, 3000 pairs were dedicated (2350 of class False and 650 of class True) which is 37.5% of the entire data set.

The 8000 couples were normalized and then used to generate the features described in section 3.

5. Experiment

The experiment compares the performances of supervised learning in three different settings:

1. The first setting relies on features related to simple lexical title text similarity between pairs; this should be the baseline of the experiment.

2. The second setting uses all the features in setting 1, in addition to that, the semantic similarity obtained by NGD).
3. The third setting adds the contextual similarity features (Time distance, Importance, Community interest) to the features of the first and second settings.

Among the various classification algorithms that were tested, the Random Forests classifier [48] has shown the best results in terms of precision, recall, and f-Measures. Note that comparison between classifiers is not the focus of this experiment, which is intended to validate the importance of content and contextual features. And therefore, Random Forest was used to compare the two settings mentioned above.

Table 2 shows the results reported from the Random Forests Classifier in the first setting using textual similarity of text only.

Table 2. Performance report (precision, recall and F-measure) with setting 1, baseline

Measure	Value	Derivations
Sensitivity (recall of True-class)	0.4585	TPR = TP / (TP + FN)
Specificity (recall of False-class)	0.8277	SPC = TN / (FP + TN)
Precision (of True-class)	0.4239	PPV = TP / (TP + FP)
Negative Predictive Value (Precision of Flase-class)	0.8468	NPV = TN / (TN + FN)
False Positive Rate	0.1723	FPR = FP / (FP + TN)
False Discovery Rate	0.5761	FDR = FP / (FP + TP)
False Negative Rate	0.5415	FNR = FN / (FN + TP)
Accuracy	0.7477	ACC = (TP + TN) / (P + N)
F1 Score	0.4405	F1 = 2TP / (2TP + FP + FN)
Matthews Correlation Coefficient	0.2783	TP*TN - FP*FN / sqrt((TP+FP)*(TP+FN)*(TN+FP)*(TN+FN))

Table 3 shows the result with the second setting, where the proposed semantic features were used with the textual features.

Table 3. Performance report (precision, recall and F-measure) with setting 2

Measure	Value	Derivations
Sensitivity (recall of True-class)	0.4785	TPR = TP / (TP + FN)
Specificity (recall of False-class)	0.8519	SPC = TN / (FP + TN)
Precision (of True-class)	0.4719	PPV = TP / (TP + FP)
Negative Predictive Value (Precision of Flase-class)	0.8552	NPV = TN / (TN + FN)
False Positive Rate	0.1481	FPR = FP / (FP + TN)
False Discovery Rate	0.5281	FDR = FP / (FP + TP)
False Negative Rate	0.5215	FNR = FN / (FN + TP)
Accuracy	0.771	ACC = (TP + TN) / (P + N)
F1 Score	0.4752	F1 = 2TP / (2TP + FP + FN)
Matthews Correlation Coefficient	0.3287	TP*TN - FP*FN / sqrt((TP+FP)*(TP+FN)*(TN+FP)*(TN+FN))

Table 4 shows setting number 3, where contextual features were used.

Table 4. Performance report (precision, recall and F-measure) with setting 2

Measure	Value	Derivations
Sensitivity (recall of True-class)	0.5246	TPR = TP / (TP + FN)
Specificity (recall of False-class)	0.8915	SPC = TN / (FP + TN)
Precision (of True-class)	0.5721	PPV = TP / (TP + FP)
Negative Predictive Value (Precision of Flase-class)	0.8715	NPV = TN / (TN + FN)
False Positive Rate	0.1085	FPR = FP / (FP + TN)
False Discovery Rate	0.4279	FDR = FP / (FP + TP)
False Negative Rate	0.4754	FNR = FN / (FN + TP)
Accuracy	0.812	ACC = (TP + TN) / (P + N)
F1 Score	0.5474	F1 = 2TP / (2TP + FP + FN)
Matthews Correlation Coefficient	0.4296	TP*TN - FP*FN / sqrt((TP+FP)*(TP+FN)*(TN+FP)*(TN+FN))

It is noted from the tables above that there is a significant increase in performance in the second and third setting. Especially for the True class precision.

For further investigation into the quality of the proposed similar couples, a human-based evaluation was conducted, where two judges evaluated randomly selected couples that had not been part of the training or the testing. The focus is on the similar proposed articles. Therefore, 650 true labeled couples and 450 false labeled couples were used to be evaluated by the two judges. Each couple is given a score of

1-10 from each judge. Where 1 indicates the absence of similarity and higher scores indicate higher similarity.

Kappa Coefficient was used to measure the agreement between the judges. The inter-agreement score was 0.9634, which indicates a substantial agreement between the human evaluators.

The human-based evaluation showed a significant correlation between the labels predicted by the proposed approach and the average score provided by the judges. According to the Pearson correlation coefficient (r), the correlation score between the human evaluations and the proposed systems predictions is 0.8837. This indicates that there is a significant large positive relationship between human evaluation and the predictions. This correlation establishes confidence in the test.

The proposed algorithm achieved an accuracy of 0.91 based on human scores on the predicted labels. Considering that an average score from 1 to 5 corresponds to a False label, and an average score from 6 to 10 corresponds to a True label.

6. Evaluation and Assessment

The proposed approach can detect article similarity with high overall accuracy. The proposed algorithms for finding NGD and contextual features have led to increased accuracy, f-measure and precision. This was achieved without using a lexical or semantic dictionary.

Community interest seems to directly improve the performance of the True labeled pairs. In fact, setting 3 led to a 10% improvement in True precision from setting 2 and 15% improvement from setting 1. This validates the intuition that similar articles would attract similar audiences. However, importance seems to have a slight positive impact on the performance, because importance can be inherited from the article's source, not by the content of the articles themselves.

The performance of the proposed algorithm in terms of precision and F-measures is comparable with similar reported tasks on resourceful languages. Table 5 shows the results of similar Arabic and non-Arabic experiments. It should be noted that a sentence-based Arabic news similarity experiment with the same goals as this one could not be found.

Table 5. Comparison with the state-of-the-art systems for Arabic and news similarity

Reference	Type	Language	Result
Singh and Singh [45]	Bilingual news article	Hindi and English	Best Accuracy 0.81
Meddeb et al. [49]	Documents	Arabic	Root Mean Squared Error (RMSE) is 0.8
Xu et al. [44]	Multilingual news similarity	Multilingual, including Arabic	Pearson's correlation coefficient for Arabic 0.79
Nagoudi et al, [50]	Arabic-English Sentences	Arabic and English	Pearson's correlation coefficient for Arabic 0.77

The proposed approach shows superiority despite the fact that it uses the title and the context only, while most of the other approaches rely on corpus full text analysis.

This means that news articles should not be exclusively considered as a textual document, but rather it is a web entity that is composed of other important features.

Contextual based similarity can complement textual similarity. And therefore, many applications could utilize that. Especially when there is a shortage of lexical resources.

Investigating further social networks and lexical resources may increase performance, which is part of the future work.

7. Conclusion

This article describes a novel method for detecting similarities between Arabic news articles. The proposed algorithm showed effectiveness according to the conducted experiment, even though the proposed approach has limited dependency on a lexical resource. String based similarity and lexical based similarity can be used as bases for the proposed algorithm. Still, they have limited capabilities, and thus the proposed similarity measures presented in this paper have improved accuracy and precision. The results obtained by the experiment were comparable to similar experiments in the English language, which is significant considering that English is a resource rich language if compared to Arabic. The result will be improved further with the help of a carefully constructed multi-domain Arabic lexicon. And this is part of our future work.

References

[1] Deepika Varshney and Dinesh Kumar Vishwakarma, "Hoax news-inspector: a real-time prediction of fake news using content resemblance over web search results for authenticating the credibility of news articles", *Journal of Ambient Intelligence and Humanized Computing*, Print ISSN: 1868-5137, Online ISSN: 1868-5145, Vol. 12, No. 9 , pp. 8961-8974, September 2021, Published by Springer Nature, DOI: 10.1007/s12652-020-02698-1, Available: https://link.springer.com/article/10.1007/s12652-020-02698-1.

[2] Mayura Kinikar and B. Saleena, "An intelligent personalized web user information retrieval using partial least squares and artificial neural networks", *Journal of Ambient Intelligence and Humanized Computing*, Print ISSN: 1868-5137, Online ISSN: 1868-5145, pp. 1–13, January 2022, Published by Springer Nature, DOI: 10.1007/s12652-021-03518-w, Available: https://link.springer.com/article/10.1007/s12652-021-03518-w.

[3] Abdelkader El Mahdaouy, Saïd Ouatik El Alaoui and Eric Gaussier, "Improving Arabic information retrieval using word embedding similarities", *International Journal of Speech Technology*, Electronic ISSN: 1572-8110, Print ISSN: 1381-2416, Vol. 21, No. 1, pp. 121–136, March 2018, Published by Springer Nature, DOI: 10.1007/s10772-018-9492-y, Available: https://link.springer.com/article/10.1007/s10772-018-9492-y.

[4] Haibo Liu, "A tag-based recommender system framework for social bookmarking websites", *International Journal of Web Based Communities*, Vol. 14, No. 3, pp. 303–322, 2018, Published by Inderscience, UK, DOI: 10.1504/IJWBC.2018.094916, Available: https://www.inderscienceonline.com/doi/abs/10.1504/IJWBC.2018.094916.

[5] Owen Noel Newton Fernando and Chan Wei Chang, "Twittener: An aggregated news platform", in *Proceedings of the International Conference on Cyberworlds, CW 2019*, October 2019, Kyoto, Japan, ISBN: 978172812297, pp. 378–381, DOI: 10.1109/CW.2019.00071, Available: https://ieeexplore.ieee.org/document/8919155.

[6] Janakiraman Bhavithra and A. Saradha, "Personalized web page recommendation using case-based clustering and weighted association rule mining", *Cluster Computing*, Vol. 22, pp. 6991–7002, May 2019, Published by Kluwer Academic Publishers, Netherlands, DOI: 10.1007/s10586-018-2053-y, Available: https://link.springer.com/article/10.1007/s10586-018-2053-y.

[7] Mauricio Pandolfi-González, Christian Quesada-López, Alexandra Martínez and Marcelo Jenkins, "Automatic Classification of Web News: A Systematic Mapping Study", in *Advances in Intelligent Systems and Computing*, ISBN: 9783030551865, Vol. 1251, pp. 558–574, DOI: 10.1007/978-3-030-55187-2_41, September 2021, Published by Springer Nature, Available: https://link.springer.com/chapter/10.1007/978-3-030-55187-2_41.

[8] Rakesh Dutta, Biswapati Jana and Mukta Majumder, "Semantic Similarity and Word-Net Based Web News Classification", in *Proceedings of the Intelligent Techniques and Applications in Science and Technology (ICIMSAT 2019)*, Siliguri, India, 2020, pp. 728–735, DOI: 10.1007/978-3-030-42363-6_85, Published by Springer Nature, Available: https://link.springer.com/chapter/10.1007/978-3-030-42363-6_85.

[9] M. K. Vijaymeena and K. Kavitha, "A survey on similarity measures in text mining", *Machine Learning and Applications: An International Journal*, Vol. 3, No. 2, pp. 19–28, 2016, Published by AIRCC Publishing Corporation, DOI: 10.5121/mlaij.2016.3103, Available: https://www.aircconline.com/mlaij/V3N1/3116mlaij03.pdf.

[10] Xin Ye, Hui Shen, Xiao Ma, Razvan Bunescu and Chang Liu, "From word embeddings to document similarities for improved information retrieval in software engineering", in *Proceedings of the International Conference on Software Engineering*, Texas, USA, 14-22 May 2016, ISBN: 9781450339001, pp. 404–415, DOI: 10.1145/2884781.2884862, Available: https://www.aircconline.com/mlaij/V3N1/3116mlaij03.pdf.

[11] Yang Wang, Lixin Han, Quiping Qian, Jianhua Xia and Jingxian Li, "Personalized Recommendation via Multi-dimensional Meta-paths Temporal Graph Probabilistic Spreading", *Information Processing & Management*, Vol. 59, No. 1, p. 102787, January 2022, Published by Elsevier, DOI: 10.1016/J.IPM.2021.102787, Available: https://www.sciencedirect.com/science/article/pii/S0306457321002661.

[12] David Robert Stöckli and Hamid Khobzi, "Recommendation systems and convergence of online reviews: The type of product network matters!", *Decision Support Systems*, Vol. 142, March 2021, Published by Elsevier, DOI: 10.1016/j.dss.2020.113475, Available: https://www.sciencedirect.com/science/article/pii/S016792362030230X.

[13] John Wieting, Taylor Berg-Kirkpatrick, Kevin Gimpel and Graham Neubig, "Beyond BLEU: Training Neural Machine Translation with Semantic Similarity", in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL 2019)*, September 2019, Florence, Italy, ISBN: 9781950737482, pp. 4344–4355, DOI: 10.18653/v1/P19-1427, Available: https://aclanthology.org/P19-1427.

[14] Michael Shumanov and Lester Johnson, "Making conversations with chatbots more personalized", *Computers in Human Behavior*, Vol. 117, p. 106627, April 2021, Published by Elsevier, DOI: 10.1016/j.chb.2020.106627, Available: https://www.sciencedirect.com/science/article/pii/S0747563220303745.

[15] Charu C. Aggarwal and Cheng Xiang Zhai, "A survey of text clustering algorithms", in *Mining Text Data*, Vol. 9781461432, pp. 77–128, 2012, Springer, DOI: 10.1007/978-1-4614-3223-4_4, ISBN: 9781461432234, Available: http://link.springer.com/10.1007/978-1-4614-3223-4_4.

[16] Kazuhiro Seki, Yusuke Ikuta and Yoichi Matsubayashi, "News-based business sentiment and its properties as an economic index", *Information Processing & Management*, Online ISSN: 1873-5371, Print ISSN: 0306-4573, Vol. 59,

No. 2, p. 102795, March 2022, Published by Elsevier, DOI: 10.1016/J.IPM.2021.102795, Available: https://www.sciencedirect.com/science/article/pii/S0306457321002739.

[17] Reza Amalia Priyantina and Riyanarto Sarno, "Sentiment analysis of hotel reviews using Latent Dirichlet Allocation, semantic similarity and LSTM", *International Journal of Intelligent Engineering and Systems*, Vol. 12, No. 4, pp. 142–155, 2019, DOI: 10.22266/IJIES2019.0831.14, Available: http://www.inass.org/2019/2019083114.pdf.

[18] Md Shajalal and Masaki Aono, "Semantic textual similarity between sentences using bilingual word semantics", *Progress in Artificial Intelligence*, Vol. 8, No. 2, pp. 263–272, March 2019, Published by Springer, DOI: 10.1007/S13748-019-00180-4, Available: https://link.springer.com/article/10.1007/s13748-019-00180-4.

[19] Mohammad Daoud, "Building Arabic polarizerd lexicon from rated online customer reviews", in *Proceedings of the International Conference on New Trends in Computing Sciences (ICTCS 2017)*, Amman, Jordan, 11-13 October 2017, ISBN: 9781538605271, pp. 241–246, DOI: 10.1109/ICTCS.2017.25, Published by IEEE, Available: https://ieeexplore.ieee.org/abstract/document/8250295/.

[20] Carlos Roberto Silveira, Marilde Terezinha Prado Santos and Marcela Xavier Ribeiro, "A flexible architecture for the pre-processing of solar satellite image time series data - The SETL architecture", *International Journal of Data Mining, Modelling and Management*, Vol. 11, No. 2, pp. 129–143, 2019, DOI: 10.1504/IJDMMM.2019.098968, Published by Inderscience, Available: http://www.inderscience.com/link.php?id=98968.

[21] Mohammad Daoud, "Novel approach towards Arabic question similarity detection", in *Proceedings of the 2017 International Conference on New Trends in Computing Sciences (ICTCS 2019),* Amman, Jordan, 09-11 October 2019, ISBN: 9781728128825, DOI: 10.1109/ICTCS.2019.8923102, Published by IEEE, Available: https://ieeexplore.ieee.org/document/8923102.

[22] Imane Guellil, Houda Saâdane, Faical Azouaou, Billel Gueni and Damien Nouvel, "Arabic natural language processing: An overview", *Journal of King Saud University - Computer and Information Sciences*, Vol. 33, pp. 497-507, June 2021, DOI: 10.1016/j.jksuci.2019.02.006, Published by Elsevier B.V., Available: https://www.sciencedirect.com/science/article/pii/S1319157818310553.

[23] Jingwei Li, Chong Zhang and Xiangzhan Yu, "Webpage visual feature extraction and similarity algorithm", *Proceedings of the 2020 International Conference on Cyberspace Innovation of Advanced Technologies (CIAT 2020)*, Guangzhou, China, 4-6 December 2020, ISBN: 9781450387828, pp. 80–85, DOI: 10.1145/3444370.3444552, Published by ACM, Available: https://dl.acm.org/doi/10.1145/3444370.3444552.

[24] Nguyen Huy Tien, Nguyen Minh Le, Yamasaki Tomohiro and I. Tatsuya, "Sentence modeling via multiple word embeddings and multi-level comparison for semantic textual similarity", *Information Processing & Management*, Vol. 56, No. 6, p. 102090, November 2019, Published by Elsevier, DOI: 10.1016/J.IPM.2019.102090, Available: https://www.sciencedirect.com/science/article/pii/S0306457319301335.

[25] Hikmat A. Abdeljaber, "Automatic Arabic Short Answers Scoring Using Longest Common Subsequence and Arabic WordNet", *IEEE Access*, Vol. 9, pp. 76433–76445, 2021, Published by IEEE, DOI: 10.1109/ACCESS.2021.3082408, Available: https://ieeexplore.ieee.org/document/9437188.

[26] Pilar Angeles and Adrian Espino-gamez, "Comparison of methods Hamming Distance, Jaro, and Monge-Elkan", in *Proceedings of the Seventh International Conference on Advances in Databases, Knowledge, and Data Applications (DBKDA 2015)*, 24 - 29 May 2015, Roma, Italy, ISBN: 9781612084084, pp. 63–69, Available: https://d-nb.info/1129261999/34#page=74.

[27] Chunchun Zhao and Sartaj Sahni, "String correction using the Damerau-Levenshtein distance", *BMC Bioinformatics*, Vol. 20, No. 11, pp. 1–28, June 2019, Published by BioMed Central, DOI: 10.1186/S12859-019-2819-0/FIGURES/24, Available: https://link.springer.com/articles/10.1186/s12859-019-2819-0.

[28] Yun Sup Lee, Yu Sin Kim and Roger Luis Uy, "Serial and parallel implementation of Needleman-Wunsch algorithm", *International Journal of Advances in Intelligent Informatics*, Vol. 6, No. 1, pp. 97–108, March 2020, Published by Universitas Ahmad Dahlan, Indonesia, DOI: 10.26555/IJAIN.V6I1.361, Available: http://ijain.org/index.php/IJAIN/article/view/361.

[29] Rada Mihalcea, Courtney Corley and Carlo Strapparava, "Corpus-based and knowledge-based measures of text semantic similarity", in *Proceedings of the 21st National Conference on Artificial Intelligence*, 16–20 July 2006, Boston, USA, ISBN: 1577352815, Vol. 1, pp. 775–780, DOI: 10.5555/1597538.1597662, Available: https://www.aaai.org/Papers/AAAI/2006/AAAI06-123.pdf.

[30] Nathaniel Oco, Leif Romeritch Syliongka, Rachel Edita Roxas and Joel Ilao, "Dice's coefficient on trigram profiles as metric for language similarity", in P*roceedings of the 2013 International Conference Oriental COCOSDA held jointly with 2013 Conference on Asian Spoken Language Research and Evaluation (O-COCOSDA/CASLRE)*, 25 – 27 November 2013, Gurgaon, India, ISBN: 978-1-4799-2378-6, pp. 1–4, Published by IEEE, DOI: 10.1109/ICSDA.2013.6709892, Available: http://ieeexplore.ieee.org/document/6709892/.

[31] Anna Huang, "Similarity measures for text document clustering", in *Proceedings of the New Zealand Computer Science Research Student Conference (NZCSRSC 2008)*, 14 – 17 April 2008, Christchurch, New Zealand, pp. 49–56, Available: https://www.yumpu.com/en/document/read/10658147/new-zealand-computer-science-research-student-conference.

[32] Ali Muttaleb Hasan, Noorhuzaimi Mohd Noor, Taha Hussein Rassem, Shahrul Azman Mohd Noah and Ahmed Muttaleb Hasan, "A Proposed Method Using the Semantic Similarity of WordNet 3.1 to Handle the Ambiguity to Apply in Social Media Text", in *Lecture Notes in Electrical Engineering*, ISBN: 9789811514647, Vol. 621, pp. 471–483, 2020, Published by Springer Nature, DOI: 10.1007/978-981-15-1465-4_47, Available: https://link.springer.com/chapter/10.1007/978-981-15-1465-4_47.

[33] Leif Azzopardi, Mark Girolami and Malcolm Crowe, "Probabilistic hyperspace analogue to language", in *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval - SIGIR '05*, 15 -19 August 2005, Salvador, Brazil, ISBN: 1595930345, pp. 575–576, DOI: 10.1145/1076034.1076135, Available: http://dl.acm.org/citation.cfm?doid=1076034.1076135.

[34] Suhyeon Kim, Haecheong Park and Junghye Lee, "Word2vec-based latent semantic analysis (W2V-LSA) for topic modeling: A study on blockchain technology trend analysis", *Expert Systems with Applications*, Vol. 152, p. 113401, 15 August 2020, Published by Elsevier, DOI: 10.1016/J.ESWA.2020.113401, Available: https://www.sciencedirect.com/science/article/pii/S0957417420302256.

[35] Himani Mittal and M. Syamala Devi, "Subjective Evaluation: A Comparison of Several Statistical Techniques", *Applied Artificial Intelligence*, Vol. 32, No. 1, pp. 85–95, January 2018, Published by Taylor and Francis, UK, DOI: 10.1080/08839514.2018.1451095, Available: https://www.tandfonline.com/doi/abs/10.1080/08839514.2018.1451095.

[36] Ofer Egozi, Shaul Markovitch and Evgeniy Gabrilovich, "Concept-Based Information Retrieval Using Explicit Semantic Analysis", *ACM Transactions on Information Systems*, Vol. 29, No. 2, pp. 1–34, April 2011, Published by ACM, DOI: 10.1145/1961209.1961211, Available: https://dl.acm.org/doi/10.1145/1961209.1961211.

[37] Rafeeq Ahmad, Tanvir Ahmad, B. L. Pal and Sunil Malviya, "Approaches for Semantic Relatedness Computation for Big Data", in *Preceedings of the 2nd International Conference on Advanced Computing and Software Engineering (ICACSE 2019)*, 8 February 2019, Sultanpur, India, DOI: 10.2139/SSRN.3349564, Available: https://ssrn.com/abstract=3349564.

[38] Md Aminul Islam and Diana Inkpen, "Second Order Co-occurrence PMI for determining the semantic similarity of words", in *Proceedings of the 5th International Conference on Language Resources and Evaluation, LREC 2006*, May 2006, Genoa, Italy, pp. 1033–1038, Accessed: May 31, 2019, Available: http://www.lrec-conf.org/proceedings/lrec2006/pdf/242_pdf.pdf.

[39] Hiteshwar Kumar Azad and Akshay Deepak, "Query expansion techniques for information retrieval: A survey", *Information Processing & Management*, Vol. 56, No. 5, pp. 1698–1735, September 2019, Published by Elsevier, DOI: 10.1016/J.IPM.2019.05.009, Available: https://www.sciencedirect.com/science/article/pii/S0306457318305466.

[40] Didik Dwi Prasetya, Aji Prasetya Wibawa and Tsukasa Hirashima, "The performance of text similarity algorithms", *International Journal of Advances in Intelligent Informatics (IJAIN)*, Vol. 4, No. 1, pp. 63–69, March 2018, Published by Universitas Ahmad Dahlan, Indonesia, DOI: 10.26555/IJAIN.V4I1.152, Available: http://ijain.org/index.php/IJAIN/article/view/152.

[41] George A. Miller, "WordNet: A Lexical Database for English", *Communications of the ACM*, Vol. 38, No. 11, pp. 39–41, November 1995, Published by ACM, DOI: 10.1145/219717.219748, Available: https://dl.acm.org/doi/10.1145/219717.219748.

[42] Ahmed Abdelali, Kareem Darwish, Nadir Durrani and Hamdy Mubarak, "Farasa: A Fast and Furious Segmenter for Arabic", in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT 2016): Human Language Technologies, Demonstrations Session*, pp. 11–16, June 2016, California, USA, DOI: 10.18653/V1/N16-3003, Available: https://aclanthology.org/N16-3003.

[43] Kai Hsiang Yang, Yu Li Lin and Chen Tao Chuang, "Using google distance for query expansion in expert finding", in *Proceedings of the 2014 9th International Conference on Digital Information Management (ICDIM 2014)*, 29 September 2014 - 01 October 2014, Bangkok, Thailand, ISBN: 9781479954209, pp. 104–109, Published by IEEE, DOI: 10.1109/ICDIM.2014.6991419, Available: https://ieeexplore.ieee.org/document/6991419.

[44] Zihang Xu, Ziqing Yang, Yiming Cui and Zhigang Chen, "HFL at SemEval-2022 Task 8: A Linguistics-inspired Regression Model with Data Augmentation for Multilingual News Similarity", in *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, July 2022, Seattle, USA, DOI: 10.18653/v1/2022.semeval-1.157, Available: https://aclanthology.org/2022.semeval-1.157.

[45] Ritika Singh and Satwinder Singh, "Text Similarity Measures in News Articles by Vector Space Model Using NLP", *Journal of The Institution of Engineers (India): Series B*, Vol. 102, No. 2, pp. 329–338, 7 November 2020, Published by Springer Nature, DOI: 10.1007/S40031-020-00501-5, Available: https://link.springer.com/article/10.1007/s40031-020-00501-5.

[46] Katarzyna Baraniak and Marcin Sydow, "News Articles Similarity for Automatic Media Bias Detection in Polish News Portals", in *Proceedings of the 2018 Federated Conference on Computer Science and Information Systems (FedCSIS 2018)*, September 2018, Poznan, Poland, ISBN: 9788394941970, pp. 21–24, DOI: 10.15439/2018F359, Available: https://www.scinapse.io/papers/2892687632.

[47] Shahinuzzaman Shawon, Mir Ummay Touhida, Md. Zakib Uddin Khan and Sabbir Ahmed, "Similarity of Trending News A Case Study of Bangladesh", *International Journal of Research Publications*, Vol. 73, No. 1, March

2021, DOI: 10.47119/IJRP100731320211831, Available: https://www.ijrp.org/paper-detail/1832.

[48] Leo Breiman, “Random forests”, *Machine learning*, Vol. 45, pp. 5–32, 2001, Published by Springer Nature, DOI: 10.1023/A:1010933404324, Available: https://link.springer.com/article/10.1023/A:1010933404324.

[49] Ons Meddeb, Mohsen Maraoui and Mounir Zrigui, “Arabic Text Documents Recommendation Using Joint Deep Representations Learning”, *Procedia Computer Science*, Vol. 192, pp. 812–821, 2021, Published by Elsevier, DOI: 10.1016/J.PROCS.2021.08.084, Available: https://www.sciencedirect.com/science/article/pii/S1877050921015726.

[50] El Moatez Billah Nagoudi, Jérémy Ferrero, Didier Schwab and Hadda Cherroun, “Word Embedding-Based Approaches for Measuring Semantic Similarity of Arabic-English Sentences”, *Communications in Computer and Information Science*, Vol. 782, pp. 19–33, 2018, Published by Springer Nature, DOI: 10.1007/978-3-319-73500-9_2, Available: https://link.springer.com/chapter/10.1007/978-3-319-73500-9_2.