Article

Semantic Ontology-Based Approach to Enhance Arabic Text Classification

Ahmad Hawalah

College of Computer Science and Engineering, Taibah University, Madina 42317, Saudi Arabia;
ahawalah@taibahu.edu.sa

Received: 16 August 2019; Accepted: 12 November 2019; Published: 25 November 2019

Abstract: Text classification is a process of classifying textual contents to a set of predefined classes and categories. As enormous numbers of documents and contextual contents are introduced every day on the Internet, it becomes essential to use text classification techniques for different purposes such as enhancing search retrieval and recommendation systems. A lot of work has been done to study different aspects of English text classification techniques. However, little attention has been devoted to study Arabic text classification due to the difficulty of processing Arabic language. Consequently, in this paper, we propose an enhanced Arabic topic-discovery architecture (EATA) that can use ontology to provide an effective Arabic topic classification mechanism. We have introduced a semantic enhancement model to improve Arabic text classification and the topic discovery technique by utilizing the rich semantic information in Arabic ontology. We rely in this study on the vector space model (term frequency-inverse document frequency (TF-IDF)) as well as the cosine similarity approach to classify new Arabic textual documents.

Keywords: Arabic text classification; Arabic topic discovery; ontology; semantic reasoning

1. Introduction

Nowadays, text classification has become a vital technique to classify unclassified contents to pre-defined classes. Such a technique can help in finding interesting information and can enhance decision making techniques. This technique is widely used in a wide range of applications such as search engines [1–3], web mining [4], recommender systems [5–8], and sentiment analysis [9–13]. Due to the increase of Arabic content on the Internet, there is a need of an effective Arabic text classifier by considering the rich morphology, syntax and semantics of the Arabic language.

Over the last few years, many researchers have introduced different methods for effective Arabic text classification such as k-nearest neighbor (kNN), artificial neural networks (ANN), naive Bayesian classifier (NB), and support vector machine (SVM) [14–19]. However, there is little work done in analyzing Arabic text in the literature. Hence, we propose enhanced Arabic topic-discovery architecture (EATA), which can provide an improved Arabic topic discovery mechanism for documents. We have used an Arabic ontology, which has been introduced by HAWALAH [20], to propose an Arabic multi-disciplinary ontology from multiple resources. This ontology consists of a number of topics, which are linked with sub-topics. We have introduced a semantic model by using a vector space model (term frequency-inverse document frequency (TF-IDF)) and the cosine similarity approach to improve Arabic classification and topic discovery techniques. Finally, we have evaluated the performance of EATA for Arabic text classification by using common information retrieval evaluation techniques.

The rest of this paper is structured as follows. Section 2 discusses related work in the fields of topic discovery and text classification. Section 3 presents our approach of EATA. Section 4 presents the

evaluation of our proposed framework and lists the experimental results. Finally, the paper ends with conclusions and future work.

2. Previous Work

Topic discovery, text categorization, or text classification can be defined as the process of labeling unstructured data (e.g., in webpages or documents) by using pre-defined categories in order to define such data and organize them [21–26]. A lot of work has been done to study the different aspects of text classification techniques and their usage in the knowledge discovery. Similarly, many studies have been conducted on comparing different text classification algorithms for Arabic textual contents.

Mamoun et al. [27] have compared the performance of different classification algorithms when applying to Arabic texts. In this work, the authors overcome the lack of Arabic corpuses in the literature by forming a small corpus that consists of 10 categories and 1000 documents. This corpus is used in this paper to compare three classifiers such as support vector machine, naive Bayesian, and k-nearest neighbor. The authors claimed that the SVM classifier has outperformed the other classifiers, and provides better results. However, we can notice that one limitation of this work is that the number of categories that are used in the experiment is less, and there is no relationship between them.

Hussien et al. [28] have compared the performance of three classifiers: SMO, naive Bayesian, and a decision tree classifier by applying them on Arabic texts. A dataset of 2363 documents, which is categorized into 6 categories, has been used to evaluate this work. The authors have shown that applying pre-processing techniques over the Arabic texts can improve the accuracy of all the classifiers. However, the evaluation is applied on a small dataset that consists of six unrelated categories.

Wahbeh et al. [29] have conducted a comparison of three text classifiers on Arabic text such as support vector machine, naive Bayesian, and Decision Tree Classifier. They have suggested a pre-processing approach, which includes removing stop words and normalization in order to reduce the noise of Arabic texts. In this experiment, they have used a dataset of four categories and a total of 1000 documents. Moreover, they have used two evaluation measurements such as k-fold cross validation and a percentage split of testing and learning datasets. The results have showed that the naive Bayesian classifier has outperformed the other algorithms. One limitation of this work is that the evaluated dataset is very small for evaluating the algorithms.

Al-Thwaib et al. [30] have conducted a comparison study of two classifiers, such as support vector machine (SVM) and k-nearest neighbor (kNN), when applying to Arabic texts. About 800 documents have been classified under four categories. In contradiction to [29], this paper has demonstrated that the SVM algorithm has outperformed the kNN algorithm in all of the two proposed experiments such as the percentage split method, and the k-fold cross validation method.

Al-diabat and Mofleh [31] have evaluated different rule based classification algorithms for Arabic text corpuses such as one rule, rule induction and decision trees. In this experimental study, they have found that customized hybrid approach has performed better than the traditional algorithms.

The authors of [32–34] utilized naive Bayesian to perform supervised sentiment analysis for Arabic languages [32]. A multi-algorithm was used, with naive Bayesian and a bag of words model. The algorithms were applied to a dataset collected from e-commerce websites. The result showed the accuracy for naive Bayesian is 93.87% [33] by applying the hyper algorithm with naive Bayesian and with unigram and bigram features for the algorithms applied to a Twitter dataset. The result showed the accuracy is up to 90% [34]; using naive Bayesian, with precision, recall and f-measure metrics, the result showed the accuracy of naive Bayesian lexicon up to 90%.

The authors of [35,36] applied naive Bayesian in their experiments with a dataset collected from social networks including Twitter and Facebook [35] and addressed the issue related to the lack of

lexicons for analysis and testing in the Arabic sentiment analysis context. A relatively large dataset of modern standard Arabic (MSA) comments and reviews has been collected. The result for naive Bayesian performance accuracy is up to 85%. Moreover, the supervised approach was used by [36]; the study used hyper-languages Iraqi, Egyptian, and Lebanese dialect Arabic with 10-fold cross-validation. The result showed an Accuracy of 87.9%.

There are several studies on the pre-processing techniques of Arabic documents to enhance the classification performance. Larkey et al. [37] have proposed a novel stemming approach to improve the information retrieval for Arabic documents using the TREC Arabic dataset. El-Khair and Ibrahim Abu [38] have observed that the classification effectiveness and accuracy of Arabic documents is enhanced by eliminating the stop words in these documents.

Reviewing the previous research, we noted that most of the proposed researchers used traditional classification techniques, but little attention has been made to find the effectiveness of more advanced techniques, such as ontologies and semantic web, to improve Arabic text classification.

3. Enhanced Arabic Topic-Discovery Architecture (EATA)

In 2018, we published a paper on introducing a framework for constructing an Arabic multi-disciplinary ontology from various resources. This framework had two main phases, building an Arabic ontology and enhancing the Arabic ontology [20]. In this paper, we have introduced a third phase, in which we provide a novel way to utilize the prepared ontology and provide an effective mechanism for Arabic topic discovery. Figure 1 represents the complete EATA including the relationship with the previous phase.

Figure 1. An Enhanced Arabic Topic-discovery Architecture (EATA).

Phase 3: Semantic Topic Discovery

In this phase, we aim at proposing a semantic enhancement model to improve the Arabic classification and topic discovery technique by utilizing the rich semantic information in the Arabic ontology. We rely in this phase on the vector space model (VSM) as well as the cosine similarity approach to classify new Arabic text in the Arabic ontology [39,40].

According to Salton et al. [41], classifying textual contents by using VSM and ranking them based on the similarity weights is effective and simple. Due to this reason, many researchers have used this technique for different applications. For example, Challam et al. [42] have suggested VSM to provide users with more precise search results based on their pre-collected profiles. Castells et al. [43] have proposed an ontology-based model to enhance the search mechanism by using VSM and a ranking algorithm. Mamoun et al. [27] have observed that VSM performs better than the other classifiers such as k-nearest neighbor (kNN) and naive Bayesian (NB) in classifying Arabic documents. Odeh et al. [44] have proposed an approach that uses the VSM to categorize Arabic textual contents.

In this project, VSM is used to convert all the Arabic ontology terms into vectors of weights using the term frequency-inverse document frequency (TF-IDF) technique using the following equations:

$$tf_{ij} = \frac{|t_i| \in d_j}{|t| \in d_j} \quad (1)$$

$$idf_i = \log\left(\frac{|d \in TS|}{|d : t \in d|}\right) \quad (2)$$

$$t_{weight} = (tf_{ij} * idf_i) \quad (3)$$

Once all the terms in the Arabic ontology are weighted, the cosine similarity approach is used to map any new Arabic text onto the Arabic ontology using Equation (4).

$$sim_{Cosine}(d_1, d_2) = \frac{\sum_{i=1}^n w_{i1} w_{i2}}{\sqrt{\sum_{i=1}^n w_{i1}^2} * \sqrt{\sum_{i=1}^n w_{i2}^2}} \quad (4)$$

Many researchers in this stage have attempted to improve the classification approach by taking advantage of the semantic information in ontologies. Ontology offers a conceptual representation of information, which includes semantic entities, relations, and characteristics of all the conceptual topics. In this project, Arabic ontology is used to provide rich semantic information to discover hidden semantic knowledge.

Various literature has been published in recent years to expose the rich semantic information in ontologies to improve the traditional VSM classifier by applying different techniques. One of these techniques is presented by Trajkova and Gauch [45], who have suggested to use the similarity of textual documents and categories c_i as well as super-categories c_i . Costa and Lima [46] have extracted rich semantic knowledge from the relations of ontology to improve the classification process. Du et al. [47] have improved the classification process by introducing a semantic similarity vector space model, which combines both the vector terms and semantic similarities between them.

However, most of the past studies do not consider noise while classifying documents. Similarly, same sub-categories with different meanings may appear under different super-categories such as 'education → software' and 'computing → software' in the ODP. In this paper, we propose a semantic clustering

mechanism (SCM) to improve the classification process and overcome such challenges. The SCM aims at extracting the semantic relations that are hidden between documents and ontological categories. We first compute the cosine similarities between the ontological categories and textual documents. Then, we formulate semantic clusters from the highest similar n results. Each cluster contains all the categories that are directly associated to each other. Algorithm 1 and Figure 2 present the working of SCM:

Algorithm 1: Semantic Clustering Mechanism

AO = the Arabic ontology
 $c_i.do$ = the document that contained the description of c_i .
 $CS.Compute(x, y)$ = is the cosine similarity method to compute the similarity between x and y .
 $SW.c_id$ = the semantic similarity weight between c_i and d .
 $TC-Array$ = is the top concept array that holds all the semantic similarity results between a document and all $c \in AO$.
Input: d = a textual document that needs to be classified to a $c \in AO$
Output: The top similar concept to d
 // Step1) Compute the cosine similarity between d and all $c_i \in AO$
 Initialize $TC-Array$;
foreach $c_i \in AO$ **do**
      $SW_{c_id}.Add(CS.Compute(d, c_i.do));$
      $TC-Array.Add(SW_{c_id}, c_i);$
 // Step2) Apply semantic clustering mechanism
 $TC-Array.Sort(asc);$
foreach $c_i \in TC-Array$ **do**
     Create-new-cluster(CL_k);
      $CL_k.Add(c_i);$
      $TC-Array.Remove(c_i);$
     foreach $c_l \in CL_k$ **do**
         foreach $c_j \in TC-Array$ **do**
             if (c_l, c_j) are linked directly **then**
                  $CL_k.Add(c_j);$
                  $TC-Array.Remove(c_j);$

Figure 2 shows an example of how the SCM works. Consider a document that can be classified as a software document, and can be mapped to three sub-categories related to the ontology in Figure 2 with different relatedness scores. These are 'biology software', 'computer software' and 'robotics software'. To implement the SCM, we created clusters based on all the directly-related categories (i.e., clusters 1–3 in Figure 3) using Algorithm 1.

Then, we assume that the larger cluster with more categories and higher average cosine similarity should have more relevance to a document. Therefore, to reflect this assumption, we first compute the average of clusters using: Equation (5).

$$ClusterAverage = \frac{\sum_{i=1}^n c_i \cdot SW \in CL_j}{|c \in CL_j|} \quad (5)$$

where CL_j is the cluster whose average c_i we need to compute, SW is the similarity weight of the c_i , and $|c \in CL_j|$ is the total number of $c \in CL_j$.

Next, we compute the strength of a cluster by dividing the total number of categories under a cluster by the total number of all interesting categories in all the clusters, which is represented by Equation (12). This gives us the importance of a larger cluster with more categories.

$$ClusterStrength = \frac{|c \in CL_j|}{|c \in CL|} \quad (6)$$

where $|c \in CL_j|$ is the total number of categories under one cluster CL_j divided by the total number of categories under all clusters. Next, we compute the weight of a cluster by using Equation (7).

$$ClusterWeight = ClusterAverage + ClusterStrenght \quad (7)$$

Figure 2. A sample of a hierarchical ontology.

We calculate the weight of a cluster for all the concepts in that cluster in order to select the top ontological concept by using Equation (8).

$$SW.c_id = SW.c_id + CL_j.ClusterWeight \quad (8)$$

where $SW.c_id$ is the similarity weight between c_i and d , $CL_j.ClusterWeight$ is the weight of the cluster CL_j including c_i .

Finally, once we spread the cluster weight to all the concepts in each cluster, the concept with the highest semantic similarity among all the clusters is selected to represent the input document.

4. Evaluation

In this section, we aim at evaluating the proposed semantic clustering mechanism (SCM) in comparison to three baseline classifiers: Support vector machine (SVM), naive Bayesian (NB) and decision tree (DT). In more detail, we applied the following steps:

4.1. Step 1: Dataset Preparation

As there is a lack of Arabic datasets in general, and Arabic ontological datasets in particular, we first create an Arabic ontological dataset using the following approaches:

(1) We have used the AODP ontology dataset that was proposed by Hawalah [20]. This dataset contained 12 domains with 107 categories that contained a total of 5021 documents mapped to these categories; Table 1 shows the 13 domains of AODP. Hawalah in [20] has also used many pre-processing techniques to clean and process the textual contents of the categories including: Lexical analysis and dimensionality reduction techniques. Hawalah [20] then proposed a mechanism to enrich the AODP by populating the sub-categories with related websites from a search engine.

Table 1. Number of domains and categories of AODP.

Domain	Number of Categories
Sport	12
News	7
Computer	11
ART	6
Business	16
Society	14
Health	13
Science	10
Entertainment	8
Education	4
Family	6

(2) We found that some of the sub-categories have limited representative textual contents that might affect evaluating the proposed semantic clustering mechanism (SCM), so we decided to populate the sub-categories with low representative textual contents with manually selected webpages. In order to accomplish this task, we invited 53 participants to populate the sub-categories that contain little or no contents with pre-labeled webpages. For this stage, each participant was asked to select two sub-categories from two different domains. For each sub-category, participants were asked to search and find 10 webpages related to the selected sub-categories. For each webpage, we implanted the pre-processing and statistic textual analysis using the proposed mechanisms in [25]. In more details, these mechanisms, which were used originally to create the AODP, are used to clean the noise of the collected webpages by first removing all the non-textual contents including menus, HTML codes, images, and videos. Then, a lexical analysis was conducted to remove the following: Non-Arabic characters, numbers, symbols, extra white lines and spaces, and Arabic diacritical markings. Then, the dimensionality reduction technique removed Arabic common stop words, and finally, we normalized Arabic terms using the Arabic stemmer that was proposed by [37] and cleaned the following prefixes and suffixes:

- Replacing $\tilde{ا}$, $\tilde{ل}$, and $\tilde{ل}$ with $ا$.
- Replacing $ي$, $ئ$, $ى$ with $ي$.
- Replacing $ة$ with $ه$.
- Removing the prefixes: $ف$, $ك$, $ب$, $ل$, $ال$, $بال$, $وال$, $كال$, $لل$, $فال$.
- Removing the suffixes: $ين$, $ان$, $ية$, $يه$, $ون$.

Table 2 shows the details of the prepared dataset that will be used for the evaluation:

Table 2. Number of domains and categories of AODP.

Dataset	Number of Documents
AODP	5021
Manual	1060

Finally, we divided the prepared dataset into two subsets: training dataset and testing dataset. We will use for this evaluation the 10-fold cross-validation approach, as for each stage of the cross-validation process, the first (n) set is used as test data, and the remaining set (n – 1) is used as training data.

4.2. Step 2: Evaluation Metrics

As different methods can be used to examine classification accuracy; we use in this paper the accuracy, precision, recall, and F1 measures as follows:

- Accuracy measure: is the number of correct classifications divided by the total number of classifications, and it can be calculated as follows:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (9)$$

- Precision measure: is the ratio of the accurate data among the retrieved data, and it can be calculated as follows:

$$Precision = \frac{TP}{TP + FP} \quad (10)$$

- Recall measure: is the ratio of relevant data among the retrieved data, and it can be calculated as follows:

$$Recall = \frac{TP}{TP + FN} \quad (11)$$

- F1 measure: is a combination of the precision and recall, and can be calculated as follows:

$$F1 = \frac{2(Precision * Recall)}{Precision + Recall} \quad (12)$$

where:

- TP: true positive.
- TN: true negative.
- FP: false positive.
- EN: false negative.

4.3. Step 2: Evaluation Process

We aim at examining the impact of the proposed SCM on the Arabic classification performance, as well as comparing the performance of the SCM with three baseline methods: Support vector machine (SVM), naive Bayesian (NB), and decision tree (DT). Furthermore, we have calculated the precision, recall, f-measure, and accuracy for each domain separately and compared all the results by calculating the averages of all the measures.

5. Evaluation Results

We have calculated the precision, recall, f-measure, and accuracy for each domain of the dataset separately using four main classifier methods: SVM, ND, DT, and SCM. Figures 3–6 show the comparison results of each domain when applying the DT, NB, SVM, and SCM, respectively.

Figure 3. The comparison results of the decision tree (DT) classifier for each domain.

Figure 4. The comparison results of the naive Bayesian (NB) classifier for each domain.

Figure 5. The comparison results of the semantic clustering mechanism (SCM) classifier for each domain.

Figure 6. The comparison results of the SCM classifier for each domain.

Table 3 summarizes the results of SCM compared to the other baseline classifiers (i.e., DT, NB, and SVM) in terms of four evaluation measures: Precision, recall, f-measure and accuracy. The SCM achieved the highest average accuracy for all domains (0.856), and achieved an improvement of 5.2%, 6.2%, and 13.3% over the SVM, NB, and DT classifiers, respectively. In contrast, the DT classifier achieved the lowest accuracy as well as the lowest results of all the other measures. We also notice that the SVM achieved the second best results on all the selected measures, and that might be because the SVM usually performs well with text classifications.

Table 3. The comparison results of the proposed SCM classifier and other baseline classifiers.

Classifier	Accuracy	Measures	Results
DT	0.763	Avg. Precision	0.712
		Avg. Recall	0.726
		Avg. F1	0.714
NB	0.814	Avg. Precision	0.792
		Avg. Recall	0.806
		Avg. F1	0.796
SVM	0.822	Avg. Precision	0.851
		Avg. Recall	0.856
		Avg. F1	0.853
SCM	0.865	C0C0C0Avg. Precision	0.893
		Avg. Recall	0.889
		Avg. F1	0.891

The overall results show that the proposed SCM classifier can effectively capture the semantic relations that might exist between related categories compared to traditional classifiers.

6. Conclusions and Future Work

Numerous studies have been conducted for topic discovery based on text classification algorithms. However, most of these studies are usually targeted at processing the English language not the Arabic language. Due to the complexity of the Arabic language, the same approaches are not very effective. Therefore, in this paper, we have presented the enhanced Arabic topic-discovery architecture (EATA), which provides a semantic ontology-based approach to enhance Arabic text classification. The lack of studies in the Information retrieval domain for the Arabic language has motivated us to present a novel approach in improving the traditional classification algorithms to cope with the challenges in processing the Arabic language. In this paper, we have proposed a semantic enhancement model to improve the Arabic classification and topic-discovery technique, which utilizes the rich semantic information in the Arabic ontology. A semantic clustering mechanism is suggested to capture the semantic relations between the potential topics that can be mapped onto the textual content. We set up a mathematical model to compute the importance of such clusters and reflect it on the encompassed topics. Finally, we have conducted an experiment to validate the effectiveness of the EATA. The results show that the classification performance of the proposed framework is much better than the traditional classification algorithms.

In the future, we will attempt to increase the size of the Arabic ontology by including more domains and multi-level topics. In addition, we will examine the effect of different semantic relations in the Arabic ontology and use some of the semantic reasoning mechanisms.

Funding: This research received no external funding.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Wang, Q.; Li, J.J. Semantics Processing for Search Engines. In Proceedings of the International Conference on Pattern Recognition and Artificial Intelligence, Union, NJ, USA, 15–17 August 2018; pp. 124–126.
2. Nakamura, T.A.; Calais, P.H.; de Castro Reis, D.; Lemos, A.P. An anatomy for neural search engines. *Inf. Sci.* **2019**, *480*, 339–353. [[CrossRef](#)]
3. Dashtipour, K.; Hussain, A.; Gelbukh, A. Adaptation of sentiment analysis techniques to Persian language. In *International Conference on Computational Linguistics and Intelligent Text Processing*; Springer: Berlin, Germany, 2017; pp. 129–140.
4. Verma, K.; Srivastava, P.; Chakrabarti, P. Exploring Structure Oriented Feature Tag Weighting Algorithm for Web Documents Identification. In *International Conference on Soft Computing Systems*; Springer: Berlin, Germany, 2018; pp. 169–180.
5. Hawalah, A.; Fasli, M. Using user personalized ontological profile to infer semantic knowledge for personalized recommendation. In *International Conference on Electronic Commerce and Web Technologies*; Springer: Berlin, Germany, 2011; pp. 282–295.
6. Hawalah, A. Modelling Dynamic and Contextual User Profiles for Personalized Services. Ph.D. Thesis, University of Essex, Colchester, UK, 2012.
7. Hawalah, A.; Fasli, M. Utilizing contextual ontological user profiles for personalized recommendations. *Expert Syst. Appl.* **2014**, *41*, 4777–4797. [[CrossRef](#)]
8. Magara, M.B.; Ojo, S.O.; Zuva, T. A comparative analysis of text similarity measures and algorithms in research paper recommender systems. In Proceedings of the 2018 Conference on Information Communications Technology and Society (ICTAS), Durban, South Africa, 8–9 March 2018; pp. 1–5.
9. Dashtipour, K.; Hussain, A.; Zhou, Q.; Gelbukh, A.; Hawalah, A.Y.; Cambria, E. PerSent: A freely available Persian sentiment lexicon. In *International Conference on Brain Inspired Cognitive Systems*; Springer: Berlin, Germany, 2016; pp. 310–320.
10. Dashtipour, K.; Poria, S.; Hussain, A.; Cambria, E.; Hawalah, A.Y.; Gelbukh, A.; Zhou, Q. Multilingual sentiment analysis: State of the art and independent comparison of techniques. *Cogn. Comput.* **2016**, *8*, 757–771. [[CrossRef](#)] [[PubMed](#)]
11. Rosenthal, S.; Farra, N.; Nakov, P. SemEval-2017 task 4: Sentiment analysis in Twitter. In Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), Vancouver, DC, Canada, 3–4 August 2017; pp. 502–518.
12. Alqarafi, A.S.; Adeel, A.; Gogate, M.; Dashitpour, K.; Hussain, A.; Durrani, T. Toward's arabic multi-modal sentiment analysis. In *International Conference in Communications, Signal Processing, and Systems*; Springer: Berlin, Germany, 2017.
13. Ieracitano, C.; Adeel, A.; Gogate, M.; Dashtipour, K.; Morabito, F.C.; Larijani, H.; Raza, A.; Hussain, A. Statistical Analysis Driven Optimized Deep Learning System for Intrusion Detection. In *International Conference on Brain Inspired Cognitive Systems*; Springer: Berlin, Germany, 2018; pp. 759–769.
14. Al-Moslmi, T.; Albared, M.; Al-Shabi, A.; Omar, N.; Abdullah, S. Arabic senti-lexicon: Constructing publicly available language resources for Arabic sentiment analysis. *J. Inf. Sci.* **2018**, *44*, 345–362. [[CrossRef](#)]
15. Al-Radaideh, Q.A.; Al-Abrat, M.A. An Arabic text categorization approach using term weighting and multiple reducts. *Soft Comput.* **2018**, *23*, 5849–5863. [[CrossRef](#)]
16. Dashtipour, K.; Gogate, M.; Adeel, A.; Alqarafi, A.; Howard, N.; Hussain, A. Persian named entity recognition. In Proceedings of the 2017 IEEE 16th International Conference on Cognitive Informatics & Cognitive Computing (ICCI* CC), Oxford, UK, 26–28 July 2017; pp. 79–83.

17. Gogate, M.; Adeel, A.; Hussain, A. Deep learning driven multimodal fusion for automated deception detection. In Proceedings of the 2017 IEEE Symposium Series on Computational Intelligence (SSCI), Honolulu, HI, USA, 27 November–1 December 2017; pp. 1–6.
18. Gogate, M.; Adeel, A.; Hussain, A. A novel brain-inspired compression-based optimised multimodal fusion for emotion recognition. In Proceedings of the 2017 IEEE Symposium Series on Computational Intelligence (SSCI), Honolulu, HI, USA, 27 November–1 December 2017; pp. 1–7.
19. Hussien, I.O.; Dashtipour, K.; Hussain, A. Comparison of Sentiment Analysis Approaches Using Modern Arabic and Sudanese Dialect. In *International Conference on Brain Inspired Cognitive Systems*; Springer: Berlin, Germany, 2018; pp. 615–624.
20. Hawalah, A. A framework for building an arabic multi-disciplinary ontology from multiple resources. *Cogn. Comput.* **2018**, *10*, 156–164. [[CrossRef](#)]
21. Wu, K.K.; Meng, H.; Yam, Y. Topic Discovery via Convex Polytopic Model: A Case Study with Small Corpora. In Proceedings of the 2018 9th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), Budapest, Hungary, 22–24 August 2018; pp. 000367–000372.
22. Xu, Y.; Xu, H.; Zhu, L.; Hao, H.; Deng, J.; Sun, X.; Bai, X. Topic Discovery for Streaming Short Texts with CTM. In Proceedings of the 2018 International Joint Conference on Neural Networks (IJCNN), Rio de Janeiro, Brazil, 8–13 July 2018; pp. 1–7.
23. Tellez, E.S.; Moctezuma, D.; Miranda-Jiménez, S.; Graff, M. An automated text categorization framework based on hyperparameter optimization. *Knowl.-Based Syst.* **2018**, *149*, 110–123. [[CrossRef](#)]
24. Tang, X.; Dai, Y.; Xiang, Y. Feature selection based on feature interactions with application to text categorization. *Expert Syst. Appl.* **2019**, *120*, 207–216. [[CrossRef](#)]
25. Dashtipour, K.; Gogate, M.; Adeel, A.; Ieracitano, C.; Larijani, H.; Hussain, A. Exploiting deep learning for persian sentiment analysis. In *International Conference on Brain Inspired Cognitive Systems*; Springer: Berlin, Germany, 2018; pp. 597–604.
26. Adeel, A.; Gogate, M.; Farooq, S.; Ieracitano, C.; Dashtipour, K.; Larijani, H.; Hussain, A. A survey on the role of wireless sensor networks and IoT in disaster management. In *Geological Disaster Monitoring Based on Sensor Networks*; Springer: Berlin, Germany, 2019; pp. 57–66.
27. Mamoun, R.; Ahmed, M.A. A Comparative Study on Different Types of Approaches to the Arabic text classification. *Expert Syst. Appl.* **2014**, *41*, 4777–4797.
28. Hussien, F.M.I.; Olayah, F.; AL-dwan, M.; Shamsan, A. Arabic Text Classification Using SMO, Nave Bayesian, J48 Algorithms. *Int. J. Res. Rev. Appl. Sci.* **2011**, *9*, 10.
29. Wahbeh, A.H.; Al-Kabi, M. Comparative Assessment of the Performance of Three WEKA Text Classifiers Applied to Arabic Text. *Abhath Al-Yarmouk Basic Sci. Eng.* **2012**, *21*, 15–28.
30. Al-Thwaib, E.; Al-Romimah, W. Support Vector Machine versus k-Nearest Neighbor for Arabic Text Classification. *Int. J. Sci.* **2014**, *3*, 1–5.
31. Al-diabat, M. Arabic Text Categorization Using Classification Rule Mining. *Appl. Math. Sci.* **2012**, *6*, 4033–4046.
32. Sghaier, M.A.; Zrigui, M. Sentiment Analysis for Arabic e-commerce websites. In Proceedings of the 2016 International Conference on Engineering & MIS (ICEMIS), Agadir, Morocco, 22–24 September 2016; pp. 1–7.
33. Alayba, A.M.; Palade, V.; England, M.; Iqbal, R. Arabic language sentiment analysis on health services. In Proceedings of the 2017 1st International Workshop on Arabic Script Analysis and Recognition (ASAR), Nancy, France, 3–5 April 2017; pp. 114–118.
34. Duwairi, R.M. Sentiment analysis for dialectical Arabic. In Proceedings of the 2015 6th International Conference on Information and Communication Systems (ICICS), Amman, Jordan, 7–9 April 2015; pp. 166–170.
35. Hathlian, N.F.B.; Hafezs, A.M. Sentiment-subjective analysis framework for arabic social media posts. In Proceedings of the 2016 4th Saudi International Conference on Information Technology (Big Data Analysis) (KACSTIT), Riyadh, Saudi Arabia, 6–9 November 2016; pp. 1–6.
36. Abdulkareem, M.; Tiun, S. Comparative Analysis of ML POS on Arabic Tweets. *J. Theor. Appl. Inf. Technol.* **2017**, *1–7*. [[CrossRef](#)]

37. Larkey, L.S.; Ballesteros, L.; Connell, M.E. Light Stemming for Arabic Information Retrieval. In *Arabic Computational Morphology*; Soudi, A., Bosch, A.V.D., Neumann, G., Eds.; Number 38 in Text, Speech and Language Technology; Springer: Dutch, The Netherlands, 2007; pp. 221–243. [\[CrossRef\]](#)
38. El-Khair, I.A. Effects of stop words elimination for Arabic information retrieval: A comparative study. *Int. J. Comput. Inf. Sci.* **2006**, *4*, 119–133.
39. Salton, G.; Wong, A.; Yang, C.S. A vector space model for automatic indexing. *Commun. ACM* **1975**, *18*, 613–620. [\[CrossRef\]](#)
40. Baeza-Yates, R.; Ribeiro-Neto, B. *Modern Information Retrieval: The Concepts and Technology behind Search*, 2nd ed.; Addison-Wesley Professional: Boston, MA, USA, 2011.
41. Salton, G.; McGill, M.J. *Introduction to Modern Information Retrieval*; McGraw-Hill, Inc.: New York, NY, USA, 1986.
42. Challam, V.; Gauch, S.; Chandramouli, A. Contextual search using ontology-based user profiles. In Proceedings of the Large Scale Semantic Access to Content (Text, Image, Video, and Sound), (RIAO '07), Pittsburgh, PA, USA, 30 May–1 June 2007; ACM: Paris, France, 2007; pp. 612–617.
43. Castells, P.; Fernandez, M.; Vallet, D. An Adaptation of the Vector-Space Model for Ontology-Based Information Retrieval. *IEEE Trans. Knowl. Data Eng.* **2007**, *19*, 261–272. [\[CrossRef\]](#)
44. Odeh, A.; Abu-Errub, A.; Shambour, Q.; Turab, N. Arabic Text Categorization Algorithm using Vector Evaluation Method. *arXiv* **2015**, arXiv:1501.01318.
45. Trajkova, J.; Gauch, S. Improving Ontology-Based User Profiles. In Proceedings of the Coupling Approaches, Coupling Media and Coupling Languages for Information Retrieval, Vacluse, France, 26–28 April 2004; pp. 380–389.
46. Costa, R.; Lima, C. Document Clustering Using an Ontology-Based Vector Space Model. *Int. J. Inf. Retr. Res.* **2015**, *5*, 39–60. [\[CrossRef\]](#)
47. Du, Y.; Liu, W.; Lv, X.; Peng, G. An improved focused crawler based on Semantic Similarity Vector Space Model. *Appl. Soft Comput.* **2015**, *36*, 392–407. [\[CrossRef\]](#)

© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).