algorithms

MDPI

Article

A Novel Hybrid Genetic-Whale Optimization Model for Ontology Learning from Arabic Text

Rania M. Ghoniem [1,2,*], Nawal Alhelwa [3] and Khaled Shaalan [4]

[1] Department of Computer, Mansoura University, Mansoura 35516, Egypt
[2] Department of Information Technology, College of Computer and Information Sciences, Princess Nourah Bint Abdulrahman University, Riyadh 84428, Saudi Arabia
[3] Department of Arabic, College of Arts, Princess Nourah Bint Abdulrahman University, Riyadh 84428, Saudi Arabia
[4] Faculty of Engineering & IT, The British University in Dubai, Dubai 345015, UAE
* Correspondence: RMGhoniem@pnu.edu.sa

Received: 18 July 2019; Accepted: 11 August 2019; Published: 29 August 2019

Abstract: Ontologies are used to model knowledge in several domains of interest, such as the biomedical domain. Conceptualization is the basic task for ontology building. Concepts are identified, and then they are linked through their semantic relationships. Recently, ontologies have constituted a crucial part of modern semantic webs because they can convert a web of documents into a web of things. Although ontology learning generally occupies a large space in computer science, Arabic ontology learning, in particular, is underdeveloped due to the Arabic language's nature as well as the profundity required in this domain. The previously published research on Arabic ontology learning from text falls into three categories: developing manually hand-crafted rules, using ordinary supervised/unsupervised machine learning algorithms, or a hybrid of these two approaches. The model proposed in this work contributes to Arabic ontology learning in two ways. First, a text mining algorithm is proposed for extracting concepts and their semantic relations from text documents. The algorithm calculates the concept frequency weights using the term frequency weights. Then, it calculates the weights of concept similarity using the information of the ontology structure, involving (1) the concept's path distance, (2) the concept's distribution layer, and (3) the mutual parent concept's distribution layer. Then, feature mapping is performed by assigning the concepts' similarities to the concept features. Second, a hybrid genetic-whale optimization algorithm was proposed to optimize ontology learning from Arabic text. The operator of the G-WOA is a hybrid operator integrating GA's mutation, crossover, and selection processes with the WOA's processes (encircling prey, attacking of bubble-net, and searching for prey) to fulfill the balance between both exploitation and exploration, and to find the solutions that exhibit the highest fitness. For evaluating the performance of the ontology learning approach, extensive comparisons are conducted using different Arabic corpora and bio-inspired optimization algorithms. Furthermore, two publicly available non-Arabic corpora are used to compare the efficiency of the proposed approach with those of other languages. The results reveal that the proposed genetic-whale optimization algorithm outperforms the other compared algorithms across all the Arabic corpora in terms of precision, recall, and F-score measures. Moreover, the proposed approach outperforms the state-of-the-art methods of ontology learning from Arabic and non-Arabic texts in terms of these three measures.

Keywords: text mining; ontology learning; hybrid models; genetic algorithms; whale optimization algorithm

1. Introduction

In recent times, the internet has become people's principle source of information. A huge quantity of web pages and databases is accessed every day. The instant growth in the quantity of information accessed via the Internet has caused difficulty and frustration for those trying to find a particular piece of information. Likewise, the various kinds of information resources that exist on the Internet constitute an enormous quantity of information in the form of web pages, e-libraries, blogs, e-mails, e-documents, and news articles, all containing huge amounts of data [1]. Such information is unstructured or semi-structured, which means that the knowledge discovery process is challenging. To deal with this challenge, the semantic web was invented as an extension of the ordinary web [2].

Ontology is a method for extending web syntactic interoperability to semantic interoperability. Ontologies are exploited to represent huge data in such a way that allows machines to interpret its meaning, allowing it to be reused and shared [3]. They are formal and explicit specifications of concepts and relations [4] and play a crucial role in improving natural language processing (NLP) task performance, such as information extraction and information retrieval. Ontologies are usually restricted to a particular domain of interest. The preliminary identification of ontology is expressed as Characterization of Conceptualization. The ontology learning from texts is "The acquisition of a domain model from textual corpus" [5].

Building ontologies can be accomplished manually, automatically, or in a semi-automatic way. However, the manual building of ontologies has the drawbacks of being time-consuming, expensive, and error-prone [6]. Furthermore, it demands the cooperation of ontology engineers and domain experts. In order to avoid these shortcomings, ontology learning has evolved to automate or semi-automate the construction of ontologies. Ontology learning includes knowledge extraction through two principle tasks: concepts extraction (which constitute the ontology) and extracting the semantic relations that link them [7–9].

Despite the Arabic language's importance as the sixth most spoken language in the world [2] and the tremendous growth of Arabic content via the web in recent years, it has been given little attention in the ontology learning field [10–12]. Several contributions are available on domain ontologies in English [13–15] and other languages. However, Arabic is not commonly considered by specialists in this field. Furthermore, the automatic extraction of semantic relationships from Arabic corpora has not been extensively investigated in comparison to other languages such as English. The majority of attempts to construct Arabic ontology is still implemented manually [2,16]. Manually developing conceptual ontologies is not only a time-consuming but also a labor-intensive job. Furthermore, extra challenges are encountered when extracting knowledge from Arabic texts due to the nature of the Arabic language, the words' semantic vagueness, and the lack of tools and resources which support Arabic. Consequently, the Arabic language suffers from a lack of ontologies and applications in the semantic web [17,18].

In summary, only a few studies have considered automatic ontology learning from Arabic text [4,9,12,19–23]. These works fall into one of the following three categories: handcrafted rule-based methods [12,20,21], machine learning methods [9,19,20,22], and hybrid rule-based/machine learning methods [4,23]. The studies that have introduced rule-based approaches for ontology learning are based on extracting the semantic relationships between Arabic concepts or Arabic named entities, and utilize the same technique, which can identify linguistic patterns from a given corpus. These patterns are then converted to rules and transducers. The drawbacks of the rule-based methods include being time-consuming and having the requirement to fully cover all rules which may represent any kind of relationship. The works that have proposed machine-learning approaches for ontology learning are based on conventional classification algorithms that categorize Arabic relations into corresponding types, but do not provide any solutions to overcome the drawbacks of these classification algorithms, such as their low performance when analyzing large textual datasets and high-dimension data. Some works have attempted to overcome the shortcomings of the two previous methods by integrating inference rules and machine learning algorithms into hybrid approaches. Although this hybridization

has somewhat optimized the overall performance, more advanced hybrid approaches to optimize Arabic ontology learning are still required.

By comparison to the other languages, several studies have been conducted for learning ontology from English text [24–26], which has achieved the largest number of contributions among the other languages. Some of these studies presented rule-based approaches [24], and the others proposed machine-learning-based approaches [25,26]. In [24], the authors presented a rule-based approach for learning the English ontology in which the inductive logic programming was used to obtain ontology mapping. This method described the ontology in OWL format and then interpreted it into first-order logic. Thereafter, it generated generalized logical rules depending on background knowledge, just as mappings do. In [25], an exemplar-based algorithm was introduced to link the text to semantically similar classes in an ontology built for the domain of chronic pain medicine. In [26], a machine learning approach based upon a neural network was presented to learn ontology through the encoder–decoder configuration. Accordingly, the natural language definitions were translated into Description Logics formulae through syntactic transformation. These methods of building ontologies are domain-specific. Therefore, they are not applicable with the Arabic language and do not support the Arabic texts.

Recently, the hybrid approaches of different bio-inspired optimization algorithms [27–29] demonstrated competitive performances in different applications of computer science, where two or more algorithms of the following are used as hybrid to optimize the problem in the domain of interest: the genetic algorithm (GA) [30–32], social spider optimization [33,34], ant colony optimization (ACO) [35,36], and whale optimization algorithm (WOA) [37,38]. These methods have several merits, including having a small parameter set, simple frameworks, and capability to avoid the shortcoming of being trapped in the local optima. Thus, they are suitable for several real applications and have the robustness to solve many problems of global optimization without the need to change the original algorithm structure.

In between these algorithms, the WOA was introduced in [39] for solving the global optimization problem through emulating the humpback whales behavior. These humpback whales are well known of a hunting method, namely, bubble-net feeding [39]. This behavior operates in three phases, including coral loop, lobtail, and capture loop [39]. The extra information on this behavior can be found in [40]. In comparison to the other bio-inspired optimization algorithms, such as Particle Swarm Optimization (PSO), the WOA algorithm has a good exploration capability of the search space [37]. However, it suffers from poor exploitation and the probability to be trapped into local optima.

In addition, GA is another heuristic algorithm for combinatorial optimization [31]. In comparison to the other similar algorithms like Tabu Search (TS) [41,42] and simulated annealing (SA) [43], we can find that all of them are applied for several combinatorial optimization problems. Furthermore, they also have different properties. First, a great computational cost is required by GA to find the optimal solution. Secondly, the best solution quality provided by the GA is superior to the SA and is comparable to the TS. Moreover, the domain-specific knowledge can be incorporated by the GA in all combinatorial or optimization phases to dictate the strategy of search, in contrary to TS and SA, which lack this feature. Therefore, based on the proven superiority of the GA and WOA in many applications [30–32,37,38] and to overcome the drawbacks of the ordinary WOA, this work further demonstrates the robustness of the proposed hybrid genetic-whale optimization algorithm (G-WOA) to optimize ontology learning from Arabic texts, in which the GA algorithm is used to optimize the exploitation capability of the ordinary WOA algorithm and solve its premature convergence issue by combining the genetic operations of GA into the WOA.

This paper contributes to the state-of-the-art of Arabic ontology learning through the following:

- Firstly, a text mining algorithm is proposed particularly for extracting the concepts and their semantic relations from the Arabic documents. The extracted set of concepts with the semantic relations constitutes the structure of the ontology. In this regard, the algorithm operates on the Arabic documents by calculating the concept frequency weights depending on the term frequency weights. Thereafter, it calculates the weights of concept similarity using the information-driven

from the ontology structure involving the concept's path distance, the concept's distribution layer, and the mutual parent concept's distribution layer. Eventually, it performs the mapping of features by assigning the concept similarity to the concept features. Unlike the ordinary text mining algorithms [9,10], this property is crucial because merging the concept frequency weights with the concept similarity weights supports the detection of Arabic semantic information and optimizes the ontology learning.

- Secondly, this is the first study to propose bio-inspired algorithms for optimization of Arabic ontology learning, in which a hybrid G-WOA algorithm is proposed in this context, to optimize the Arabic ontology learning from the raw text, by optimizing the exploration-exploitation trade-off. It can benefit from a priori knowledge (initial concept set obtained using the text mining algorithm) to create innovative solutions for the best concept/relation set that can constitute the ontology.
- Thirdly, investigating the comparable performance between the proposed G-WOA and five other bio-inspired optimization algorithms [32,39,44–46], when learning ontology from Arabic text, where its solutions are also compared to those obtained by the other algorithms, across different Arabic corpora. To the best of our knowledge, the proposed and compared bio-inspired algorithms have not been investigated in Arabic or non-Arabic ontology learning yet.
- Fourthly, the proposed ontology learning approach is applicable with the other languages, where it can be applied to extract the optimal ontology structure from the non-Arabic texts.

2. Literature Review

Due to the rapid surge of textual data in recent years, several studies have concentrated on how to create taxonomy from labeled data [47–50]. In this context, there were many attempts to deal with multi-label learning/classification problems. In [47], the authors concentrated on how to learn classifiers the balanced label through label representation, using a proposed algorithm, namely, Parabel. This algorithm could learn the balanced and deep trees. The trees learned using this algorithm were prone to prediction performance degradation because of forceful aggregation for labels of head and tail into longer decision paths and generic partitions. In [48], the authors introduced a shallow tree algorithm, namely Bonsai, which can deal with the label space diversity and scales to a large number of labels. The Bonsai algorithm was able to treat with diversity in the process of partitioning by allowing a larger fan-out at every node.

In [49,50], the authors used the hierarchical and flat classification strategies with the large-scale taxonomies, relying on error generalization bounds for the multiclass hierarchical classifiers. The main goal of some of these works was the large-scale classification of data into a large number of classes, while the others concentrated on how to learn the classifier the given trees. In contrary to these works, the main goal of this paper was to introduce an approach for extracting the optimal structure that constitutes the ontology from the raw textual data by employing the text mining and bio-inspired optimization techniques.

2.1. Literature Review on Arabic Text Mining

Although several works have been devoted to text mining from English and Latin languages [51,52], little attention has been paid to mining the Arabic texts. This is mainly because of the Arabic structural complexity and the presence of several Arabic dialects. Table 1 presents state-of-the-art information on Arabic text mining [53–59]. The majority of works in this context have concentrated on using the Vector Space Model [57], Latent Semantic Indexing [56], and Term Frequency (TF)/Inverse Document Frequency (TF/IDF) [54,55]. However, these algorithms still suffer from two shortcomings: the dimension curse and the semantic information lack. Therefore, in this study, we proposed a specific text mining algorithm that begins with the conceptualization stage to extract the initial concept set constituting the ontology and captures their semantic information.

Table 1. A state-of-the-art on Arabic text mining.

Reference	Year of Publication	Arabic Text Mining Algorithm	Corpus	Accuracy
[53]	2014	Cosine Coefficient, Jacaard Coefficient, and Dice Coefficient	Saudi Newspapers (SNP)	Cosine coefficient outperformed Jaccard and Dice coefficients with 0.917, 0.979, and 0.947 for Precision, Recall, and *F*-measure, respectively.
[54]	2014	Term frequency (TF), and Term Frequency/Inverse Document Frequency (TF/IDF) for feature extraction, Semi-Automatic Categorization (SAC), and Automatic Categorization (AC) for feature selection.	News books: Arabic Dataset for Theme Classification (subsets 1 & 2)	Global recognition score is used to measure the ratio of correctly-classified documents: employing TF/IDF (95%), and TF (88%)
[55]	2015	TF/IDF, Chi Square for selecting feature, besides a local class-based policy for feature reduction	Al-Jazeera News	Recall of 0.967%, and *F*-measure of 0.959
[56]	2017	Latent Semantic Indexing	Alqabas newspaper in Kuwait	82.50%
[57]	2018	Vector Space Model (VSM)	Set by Alqabas newspaper, in Kuwait	84.4%
[58]	2018	VSM	Alqabas newspaper	90.29%
[59]	2019	Removing the stop words existed in the collected tweets, extracting the keywords and sorting them into one of the corresponding categories: classified words or unclassified words. Then, applying named entity recognition as well as data analysis rules on the classified words to generate final report. The lexical features along with Twitter-specific features were employed in classification.	A private database of collected Arabic tweets	96.49%

2.2. Literature Review on Arabic Ontology Learning

Ontology learning from text is a very important area in computer science. Published works on ontology learning from Arabic texts are still rare. As previously mentioned, the contributions of the state-of-the-art Arabic ontology learning from texts can be distinguished into one of the following categories. The works under these categories were examined in the following section and using Table 2.

The rule-based approaches [12,20,21,60–62] rely on patterns comprising all the possibly-correlated linguistic sequences commonly executed in a form of finite-state transducers or even regular expressions. Despite those methods being beneficial for a limited domain, besides their better analysis quality, they cannot act in a good way, in particular, the creation of the manually hand-crafted patterns is so laborious with regard to effort and time. Hence, through the applications of such approaches, it is difficult to manipulate enormous amounts of data.

For automating the relations extraction, some studies [9,19,22,63,64] have used machine learning algorithms involving (1) unsupervised, (2) semi-supervised, and (3) supervised learning. For the unsupervised methods, the popular approach takes clusters from the patterns of the same relationship and then generalizes them. However, the semantic representations of relational patterns, in addition to the scalability to big data, make these methods face a challenge in reference to the reliability of the obtained patterns [55]. Although these algorithms can manipulate large quantities of data, the conversion of the output relations to ontologies represents a labor-intensive task.

Table 2. A state-of-the-art on Arabic ontology learning.

Methodology	Works	Year of Publication	Contribution
Rule-based approaches	[60,61]	2010 & 2012	Extracting a set of linguistic patterns from text then rewriting it into finite state transducers
	[12]	2016	The authors developed a model of pattern recognizer that targets to signal the existence of cause–effect information in sentences from non-specific domain texts. The model incorporated 700 linguistic patterns to distinguish the sentence parts representing the cause, besides to these representing the effect. To construct patterns, various sets of the syntactic features were considered through analyzing the untagged corpus.
	[62]	2017	The authors introduced a rule-based system namely, ASRextractor, to extract and annotate semantic relations relating Arabic named entities. The semantic relation extraction was based upon an annotated corpus of Arabic Wikipedia. The corpus incorporated 18 types of semantic relations like synonymy and origin.
	[20]	2018	A statistical parsing method was adopted to estimate the key-phrase/keyword from the Arabic dataset. The extracted dataset was converted to an OWL ontology format. Then, the mapping rules were used to link the components of ontology to corresponding keywords.
	[21]	2018	A set of rules/conjunctive patterns were defined for extracting the semantic relations of the Quranic Arabic according to a deep study for Arabic grammar, POS tagging, as well as the morphology features appears in the corpus of Quranic Arabic.
Machine learning-based approaches	[63]	2009	With the objective of semantic relation extraction, the authors amalgamated two supervised methods, to be specific, the basic Decision Tree as well as Decision Lists-based Algorithm. They estimated Three semantic relations (i.e., location, social and role) among named entities (NEs).
	[22]	2009	On the basis of the dependency graph producing by syntactic analysis, the authors adopted a learning pattern algorithm, denoted $(LP)^2$ for generating annotation rules.
	[19]	2013	A rule mining approach has been proposed to be applied on an Arabic corpus using lexical, numerical, and semantic features. After the learning features were extracted from the annotated instances, a set of rules were generated automatically by three learning algorithms, namely, Apriori, decision tree algorithm C4.5, and Tertius.
	[9]	2017	A statistical algorithm was used to extract the simple and complex terms, namely, "the repeated segments algorithm". For selecting segments that have sufficient weight, the authors used the Weighting Term Frequency–Inverse Document Frequency algorithm (WTF-IDF). Further, a learning approach was proposed based upon the analysis of examples for learning extraction markers to detect new pairs of relations.
	[64]	2018	Genetic algorithm (GA) was proposed to minimize the computation time needed to search out the informative and appropriate Arabic text features needed for classification. The SVM was used as machine learning algorithm that evaluates the accuracy of the Arabic named entities recognition.
Hybrid approaches	[65]	2013	Three methodologies were encompassed: kernel method, co-occurrence, and later rule-based. These methods were utilized for extracting simple and complicated relations regard the biomedical domain. For mapping the data into a feature space of high-dimensionality, Kernel-based algorithms have been used.
	[23]	2014	The authors proposed a hybrid rule-based/machine learning approach and a manual technique for extracting semantic relations between pairs on named entities.
	[4]	2017	A rules patterns set was defined from compound concepts for inducing of general relations. It utilized a gamification mechanism to specify relations based on prepositions semantics. The Formal Concept Analysis/Relational Concept Analysis approaches were employed for modeling the hierarchical as well as transversal relations of concepts.

To encounter the drawbacks of the unsupervised approaches, the studies investigated the semi-supervised methods or bootstrapping techniques that need seeding-points sets rather than training sets. The seeds are linguistic patterns or even relation-instances which are applied in an iterative way for acquisition of more basic elements until all objective relations are found. The shortcoming of the bootstrapping approaches deeply relies on the chosen initial seeds, which might reflect precisely the information of the corpus. On the other side, the extraction caliber is low. The supervised techniques [63] are the last category under the machine learning-based approaches, which depends on a completely labeled-corpus. Thus, extracting the relations is regarded as a matter of classification, according to the supervised techniques. Amongst them, we mention conditional random fields, support vector machine (SVM) [64], decision tree [19], in addition to Maximum Entropy (MaxEnt). These algorithms give a low performance in case of the high-dimensional corpora.

On the other side, the researchers have successfully addressed some of the previously discussed challenges such as the long sentences of Arabic and the non-fixed location of semantic relations in sentences. Therefore, they have integrated the rule-based method with machine learning to get hybrid approaches [4,23,65]. These hybrid methods have demonstrated enhanced performance in comparison to the single rule-based or the machine learning-based approaches. Generally, recent literature demonstrates a huge interest in the hybrid artificial intelligence-based models to solve problems in several domains. In [27], a hybrid algorithm integrates the merits of GA, including the great global converging ratio together with ACO to introduce solutions for the supplier selection problems. In [28], a genetic-ant colony optimization model was proposed to overcome the word sense disambiguation that represents a serious natural language processing problem. Therefore, it is important to propose hybrid intelligent approaches to introduce numerous choices for unorthodox handling of Arabic ontology learning problem, which comprise vagueness, uncertainty, and high dimensionality of data.

In this context, these hybrid bio-inspired optimization algorithms can present innovative solutions to support the Arabic language. They can overcome the key shortcoming of existing methods for Arabic ontology learning as they can deal with the high-dimensional or sparse data that makes it hard to capture the relevant information, which helps to learn ontology via dimensionality reduction, depending on selecting only the optimal concepts and semantic relations that contribute to the ontology structure and ignoring the non-related ones. Therefore, this paper contributes to the state-of-the-art on Arabic ontology learning with a hybrid model based on GA and WOA. This model was experimented to ontology learning using a number of the publicly available Arabic and non-Arabic corpora.

3. Preliminaries

3.1. Genetic Algorithm

The GAs [30–32] are random-search algorithms that are inspired by natural genetic mechanism and biological natural selection, which belong to the computational intelligence algorithms. The GA emulates the reproduction, crossover, and mutation in the process of genetic mechanism and natural selection. In the GAs, the individual is the optimized solution of the problem, namely the chromosome or genetic string. The GA can be expressed as an eight tuple: $GA = \{C, Fitness, P, Pop_{Size}, L, \alpha, \beta, S\}$, where C is the encoding method for the individuals within population, $Fitness$ is a fitness function for evaluating individuals, P is the initial solution, Pop_{Size} is the population size, L, α and β indicate the operators of selection, crossover and mutation, respectively, and S defines the GA termination condition. A GA begins with the initial population of chromosomes or strings and then produces successive populations of chromosomes. The basic GA comprises the following three operations:

- Reproduction. The reproduction means keeping chromosomes without changes and transferring them to the next generation. Inputs and outputs of this procedure are the same chromosomes.
- Crossover. This process concatenates two chromosomes to produce a new two ones through switching genes. On this basis, the input for this step is two chromosomes, whereas the output is two different ones.
- Mutation. This process reverses randomly one gene value of a chromosome. Thus, the input chromosome is completely different from the output one.

When determining not to conduct crossover, the chromosomes of parents are duplicated to the off-spring without change. Evolution speed of genetic search is altered by varying the probability of crossover. Practically, the crossover value is close to 1. Contrarily, the mutation ratio is usually fairly small.

3.2. Whale Optimization Algorithm

The WOA was proposed in [39]. It is inspired by the humpback whales' behavior. In comparison to the other bio-inspired algorithms, the WOA improves the candidate solutions in each step of optimization. In this context, the emulation of bubble-nets was implemented using a spiral movement. This procedure imitates the helix-shaped movement of the actual humpback whales.

3.2.1. Encircling Prey

Assume that a whale $c(i)$ has a position which is updated through moving it simultaneously in a spiral around its prey c_{best}. Mathematically, this procedure is expressed as follows:

$$c(i + 1) = S.\, e^{hr}.\, cos(2\,\pi\, r)\;\; c_{best}(i) \tag{1}$$

where $S = \left| c_{best} - c(i) \right|$ refers to the distance between $c(i)$ and c_{best} at iteration i, $r \in [-1, 1]$ represents a random number, and h is a constant variable defining a logarithmic spiral shape. The positions of the wales are updated by the encircling behavior based upon $c_{best}(i)$ as follows:

$$S = \left| K.\, c_{best} - c(i) \right| \tag{2}$$

$$c(i + 1) = c_{best}(i) - A.\, S \tag{3}$$

K and A represent coefficient vectors and are defined using

$$K = 2m \tag{4}$$

$$A = 2o\, m - o \tag{5}$$

where m denotes a random vector and e is decreased linearly from 2 till 0 along iterations i, then the value of o is computed using

$$o = o - i\,\frac{o}{o_{max}} \tag{6}$$

3.2.2. Bubble-Net Attacking Method

For the bubble-net attacking, the whales are able to swim simultaneously around the prey over a spiral-shaped path and throughout a shrinking circle. Equation (7) defines this behavior:

$$c(i + 1) = \begin{cases} c_{best} - A.\, S & if\;\; m < 0.5 \\ S.\, e^{pr}.\, cos(2\,\pi\, r + c_{best}(i) & if\;\; m > 0.5 \end{cases} \tag{7}$$

where $m \in [0, 1]$ refers to the probability of choosing the mechanism of swimming on all the prey's sides (weather spiral model-based swimming or shrinking encircling-based swimming). Nevertheless, humpback whales search for prey in a random manner.

3.2.3. Searching for Prey

In reality, humpback whales swim randomly so that they search for prey. The positions of the whales are updated using a randomly chosen whale $c_{rand}(i)$ as given below:

$$S = \left| K.\, c_{rand}(i) - c(i) \right| \tag{8}$$

$$c(i + 1) = c_{rand}(i) - A.\, S. \tag{9}$$

Eventually, based upon the value of e (decreases from 2 till 0), K, A and the probability m, the position of every *ith* whale is updated. If $m > 0.5$, then go to Equation (1). Otherwise, go to either Equations (2) and (3) or Equations (8) and (9) depending on the value of $|K|$. This procedure is repeated until the stopping condition.

3.3. Arabic Ontology Learning

Ontology learning is one of the most important issues in Arabic language processing. In the literature, to construct the ontology of any conceptual domain, this is based on three dominant linguistic theories:

3.3.1. The Semantic Field Linguistic Theory

The semantic field linguistic theory [17], in which the word meaning is deemed within a specific perspective of the world, was presented by Jost Trier [5]. Accordingly, it is determined by its relationship to the words within the field/domain (conceptual area). It presumes that each word is constructed inside semantic fields based upon a primitive feature set. Moreover, the position of the word within the field determines its meaning, and the relations it creates with the remaining words in this field. Utilizing componential analysis, what is meant by a word is established in reference to some specified atomic components or decompositions representing the features that distinguish a considered word. Such features form the base for structuring a particular semantic domain. The individual word meaning can be identified as an integration of the representative features. Such formulae are indicated as componential definitions for the semantic units and denoting formalized dictionary definitions.

3.3.2. The SEMANTIC analysis Linguistic Theory

This is a strategy to extract and represent the meaning of word contextual usage by applying statistical methods to the textual corpus [66]. The main idea is to aggregate words into contexts within which a specified word is or does not belong. This depends on a set of constraints that decides the similarities of word meanings and sets words to each other.

3.3.3. The Semantic Relations Theory

Underlying semantic relations for Arabic text show a great deal of variety [67]. The three semantic relationships considered in the current work can be explained with the following examples of biomedical concepts from our corpus:

- Synonymy. This relationship type aims concepts that hold nearly similar meanings. For instance, the concepts شهيقinspiration and استنشاقinhalation are synonyms.
- Antonyms. This relationship aims concepts that demonstrate opposite meanings, i.e., antonyms, like خبيثmalignant, and حميد benign.
- Inclusion. This type of relation means that one entity-type comprises sub entity-types. For example, the concept صمام رئويpulmonary valve with the concept قلبheart, can indicate a part-to-whole or Is-a relationship. Figure 1 presents an example of some biomedical knowledge concepts available in our corpus which are linked with an Is-a relationship.

Figure 1. Representation of some biomedical concepts in our corpus which have an Is-a semantic relationship.

4. Proposed Model for Arabic Ontology Learning

This section introduces the proposed model for ontology learning from Arabic text. The proposed model integrates: (1) a proposed text mining algorithm for extracting the concepts and the semantic relations which they are linked with, from the text documents, and (2) a proposed hybrid genetic-whale optimization algorithm to select the optimal concept/relationship set that constitute the Arabic ontology.

4.1. Pre-Processing

Pre-processing of Arabic texts in the three datasets investigated in this study is performed in two steps:

- Eliminating stop-words. Words like pronouns and conjunctions are extremely common and if we remove these words from text we can focus on important concepts. Examples of stop words are: 'في' → *'in'*, 'هذا' → *this*, 'بين' → *'between'*, 'مع' → *'with'*, 'إلى' → *'to'*, 'أو' → *'or'*, 'و' → *'and'*, etc.
- Stemming. This task leaves out the primitive form of a word. Thus, words or terms that share identical root but differ in their surface-forms due to their affixes can be determined. Such a procedure encompasses eliminating two things: a prefix, like 'ال', at the start of words and as suffix such as 'ية'at the end of words. An instance of eliminating a prefix and a suffix is the input word ''السرطانية'cancerous' which is stemmed to 'سرطان' 'cancer'.

4.2. Proposed Text Mining Algorithm

The algorithm extracts concepts and their semantic relations that constitute the ontology from each document of Arabic text, in three steps: Term weighting, concept similarity weights, and feature mapping.

4.2.1. Term Weighting

The weight in text mining is a well-known statistical measure for evaluating how important a term (word) is for a textual document in a corpus. Thus, we assigned a weight to each term of a document. This procedure is called term weighting. Thereby, every document is expressed in a vector form relying on the terms encompassed inside. Formally speaking, the vector that characterizes the document will be in the following format:

$$doc_n \;=\; \left\{TW_1,\; TW_2,\; \ldots\ldots\; TW_a,\; \ldots\ldots\; TW_{|C|}\right\} \tag{10}$$

where TW_a refers to the weighting of the term that has the number m in the doc document of index n, C represents the term set, and $|C|$ denotes the cardinality of C.

To obtain a vector involves the terms of C, the TF-IDF is utilized as weighting. Assume that the term frequency TF_a expresses the occurrences number of T_a within the document, and the document frequency DF_a is the document number in which the given term T_a can be seen at least once. Thus, we can compute the inverse document frequency IDF_a, as illustrated in Equation (11) using DF_a [68]:

$$IDF_a \;=\; \log\left(\frac{|DOC|}{DF_a}\right) \tag{11}$$

where $|DOC|$ denotes the number of documents assigned as a training set, and $\mathrm{TW_a}$ is computed by Equation (12):

$$\mathrm{TW_a} \;=\; \mathrm{TF_a} \cdot \mathrm{IDF_a}. \tag{12}$$

Subsequently, the irrelevant and redundant features are eliminated from the text document, thus, we can represent the document set as a "document-term" matrix as follows:

$$\begin{bmatrix} T_1 \\ T_2 \\ . \\ . \\ T_a \end{bmatrix} = \begin{bmatrix} TW_{(1,1)} & TW_{(1,2)} & \ldots\ldots & TW_{(1,a)} \\ TW_{(2,1)} & TW_{(2,2)} & \ldots\ldots & TW_{(2,a)} \\ . & . & \ldots\ldots & . \\ . & . & \ldots\ldots & . \\ TW_{(a,1)} & TW_{(a,2)} & \ldots\ldots & TW_{(a,A)} \end{bmatrix}. \tag{13}$$

Depending on the resulting weights for feature frequency, the algorithm maps the document's terms to corresponding concepts. As illustrated in Algorithm 1, TW and CW are two matrices to the same document, and S_T and C_T indicate the sets of terms and concepts, respectively. The algorithm reveals that through mapping, the document's terms to correlative concepts, the document's vector of terms will be converted into a vector of concepts. Thus, the algorithm will replace the document set of Equation (13) by the *document-concept matrix* in Equation (14):

$$\begin{bmatrix} T_1 \\ T_2 \\ . \\ . \\ T_a \end{bmatrix} = \begin{bmatrix} CW_{(1,1)} & CW_{(1,2)} & \ldots\ldots & CW_{(1,a)} \\ CW_{(2,1)} & CW_{(2,2)} & \ldots\ldots & CW_{(2,a)} \\ . & . & \ldots\ldots & . \\ . & . & \ldots\ldots & . \\ CW_{(a,1)} & CW_{(a,2)} & \ldots\ldots & CW_{(a,A)} \end{bmatrix} \tag{14}$$

where $CW_{(l,\, m)}$ denotes the frequency weight for "concept 1" in document m, a represents the documents number, and A is the concepts number.

Algorithm 1: The Proposed Arabic Text Mining Algorithm

Input: A term weighting matrix TW of training set corresponding to term set $S_T = \{T_1, T_2, \ldots., T_a\}$
Output: Matrix of mapped features WS obtained by assigning concept similarity weights to the concepts

1. //Mapping terms to concepts
2. //The matrix CW is initially another copy of TW
3. Update a matrix CW with the resulting concept weighting set corresponding to concept set $C_T = \{C_1, C_2, \ldots\ldots, C_a\}$, as follows:
4. **While** $(S_T \neq \Phi)$
5. **While** $(C_T \neq \Phi)$
6. **For** $A = 1$ to $count\ (S_T)$
7. **For** $B = 1$ to $count\ (C_T)$
8. **IF** $Matching\ \ (T_A, C_B) = 1$ //The two elements are equal
9. $C_B \leftarrow mapping\ \ (T_A)$
10. $CW_{(A,B)} = CW_{(A,B)} + TW_{(A,B)}$
11. $Remove\ \ T_A\ from\ \ S_T$
12. **ELSE**
13. $Remove\ \ T_A\ from\ S_T$
14. **END IF**
15. **END FOR**
16. **END FOR**
17. END
18. END
19. //Calculation of semantic similarities between n concepts of CW
20. //Matrix of resulting weights of concept similarity
21. $[] \leftarrow S$
22. **For** $A = 1$ to $count\ (S_T)$
23. **For** $B = 1$ to $count\ (C_T)$
24. //Computation of semantic similarities between each two concepts in CW
25. $$WPD = \lambda \cdot \frac{M}{min(layer\ (CW_{(A,A)}), layer\ (CW_{(A,B)})} \text{ //} M \text{ is top layer number} \quad (15)$$
26.
27. $$Similarity(CW_{(A,A)}, CW_{(A,B)}) = \frac{layer(Nearest(CW_{(A,A)}, CW_{(A,B)}))}{WPD(CW_{(A,A)}, CW_{(A,B)}) \cdot M} \quad (16)$$
28. Append the similarity between $CW_{(A,A)}$ and $CW_{(A,B)}$ to S as:
29. $WS(A, B) = Similarity(CW_{(A,A)}, CW_{(A,B)})$
30. **End For**
31. **End For**
32. Assign resulting concept similarity weights to the concepts according to Equation (20).

4.2.2. Concept Similarity Weights

In this study, experts in the domain of Arabic language implemented the conceptual characterization of the Arabic ontology. The concepts and semantic relations of the ontology hierarchy were then built using the Protégé tool [69]. Considering the concept hierarchy structure of the biomedical information depicted in Figure 1, the concept similarities can be computed based on the distances between nodes. In this regard, computing distances between nodes has been introduced in several studies through different methods depending on the domain of application [70]. In the current

study, computing similarities among concepts that constitute the ontology structure encompasses three elements: (1) the path distance between concepts, (2) the concept's distribution layer, and (3) the mutual parent concept's distribution layer. For each concept node within the ontology, we can trace and obtain all its paths to the root concept node, then generate the routing table of the ontology.

Therefore, the concepts weighted path distance (WPD) is calculated by considering the following factors:

If the path distance (*PD*) that the concepts have is long, they will have less similarity, as in the following example, where *C* is the concept node of index *i* in the ontology structure.

$$\begin{aligned} &\text{IF } PD(C_4,\ C_{16}) < PD(C_1,\ C_{16}) \text{ THEN} \\ &\qquad Similarity(C_4,\ C_{16}) > Similarity(C_1,\ C_{16}) \\ &\text{END IF} \end{aligned}$$

Neglecting the path distance factor, the deeper the neighboring concepts localize at the distribution layer-level, the higher the similarity they have, as

$$Similarity(C_{16},\ C_4) > Similarity(C_4,\ C_1) > Similarity(C_1,\ C_5).$$

For concepts that have a mutual parent, the deeper they localize at the distribution layer, the higher the similarity they have, as an instance:

$$Similarity(C_{16},\ C_{15}) > Similarity(C_4,\ C_2).$$

Assuming two adjacent concepts q_A and q_B, we can compute the WPD of the concepts using Equation (15) of Algorithm 1, where $layer\ (q_A)$, and $layer\ (q_B)$ denote the distribution layer number for concepts q_A, and q_B, respectively. M represents the number of upper layer in the entire ontology hierarchy besides λ which is a scalar that is set through experimentation. For our work, it was assigned a value of 1.

Eventually, we estimated the similarity between the given concepts q_A and q_B using Equation (16) (see Algorithm 1) where $layer(Nearest(q_A,\ q_B))$ indicates the distribution layer number of the closest common concept of concepts q_A and q_B. After computing the concept similarities for all the concepts in the document's ontology hierarchy, we can construct a matrix of "concept-concept" as show below:

$$WS = \begin{bmatrix} Similarity(q_1,\ q_1) & Similarity(q_1,\ q_2) & \ldots\ldots\ldots & Similarity(q_1,\ q_n) \\ Similarity(q_2,\ q_1) & Similarity(q_2,\ q_2) & \ldots\ldots\ldots & Similarity(q_2,\ q_n) \\ . & . & \ldots\ldots\ldots & . \\ . & . & \ldots\ldots\ldots & . \\ Similarity(q_n,\ q_1) & Similarity(q_n,\ q_2) & \ldots\ldots\ldots & Similarity(q_n,\ q_N) \end{bmatrix} \tag{17}$$

$$= \begin{bmatrix} 1 & Similarity(q_1,\ q_2) & \ldots\ldots\ldots & Similarity(q_1,\ q_n) \\ Similarity(q_2,\ q_1) & . & \ldots\ldots\ldots & Similarity(q_2,\ q_n) \\ . & 1 & \ldots\ldots\ldots & . \\ . & . & \ldots\ldots\ldots & . \\ Similarity(q_n,\ q_1) & Similarity(q_n,\ q_2) & \ldots\ldots\ldots & 1 \end{bmatrix} \tag{18}$$

4.2.3. Feature Mapping (Assigning Similarity Weights to the Mapped Concepts)

As for the "*concept-concept*" matrix, the values within *WS* ought to be either above or equivalent to "0", where "0" denotes to non-similar concepts whereas "1" denotes to similar concepts, and *WS* is delineated as asymmetric-positive semi-definite. Hence, we can express *WS* as in Equation (19):

$$WS = E * DG * E^{-1} = E * \sqrt{DG} * \sqrt{DG} * E^{-1}. \quad (19)$$

where

$DG \rightarrow$ a diagonal matrix whose elements denote the non-negative eigenvalues of *WS*,
$E \rightarrow$ an orthogonal matrix whose columns point to the corresponding eigenvectors,
$\sqrt{DG} \rightarrow$ a diagonal matrix whose diagonal items are the square root for *DG* diagonal elements.

Eventually, the document set that is expressed as in Equation (14) will be rewritten as

$$\begin{bmatrix} \hat{T}_1 \\ \hat{T}_2 \\ . \\ . \\ \hat{T}_n \end{bmatrix} = \begin{bmatrix} T_1 * E\sqrt{DG} \\ T_2 * E\sqrt{DG} \\ . \\ . \\ T_n * E\sqrt{DG} \end{bmatrix} = \begin{bmatrix} T_1 \\ T_2 \\ . \\ . \\ T_n \end{bmatrix} E\sqrt{DG} = \begin{bmatrix} CW_{(1,1)} & CW_{(1,2)} & ..... & CW_{(1,a)} \\ CW_{(2,1)} & CW_{(2,2)} & ..... & CW_{(2,a)} \\ . & . & ..... & . \\ . & . & ..... & . \\ CW_{(a,1)} & CW_{(a,2)} & ..... & CW_{(a,A)} \end{bmatrix} E\sqrt{DG} \quad (20)$$

where

$\hat{T}_n = T_n * E\sqrt{DG}$,
$CW_{(1,\,1)} \rightarrow$ the frequency weight of "concept *l*" in document *m*,
$a \rightarrow$ the documents number,
$A \rightarrow$ the concepts number.

4.3. The Proposed Hybrid Genetic-Whale Optimization Algorithm for Arabic Ontology Learning

In the ordinary WOA, the exploitation phase relies on computing the distance between the whale (search agent) and the best one known in this iteration. To optimize the exploitation capability of WOA and solve the premature convergence issue of the WOA, in this study, the genetic operations of GA were combined into WOA. The core of the proposed algorithm, G-WOA, is the hybridization of the WOA's operators along with GA's operators [71] to optimize the ontology learning from Arabic text by optimizing the WOA's exploration-exploitation trade-off. The operator of G-WOA is mainly a hybrid operator (as shown in lines 7 to 29 of Algorithm 2), which integrates GA's mutation, crossover, selection, and the WOA's components, called, encircling prey, bubble-net attacking, and searching for prey.

4.3.1. Initial Population

The GA is embedded into the WOA algorithm in order to develop a number of whales (search agents) in the form of chromosomes. Every chromosome is a hypothesis for the best solution (preys). Therefore, every search agent contains genes, each of which represents a concept/semantic relation of the ontology. A set of random agents $c^t_{p,j}$ is generated initially. After generating the random solutions, the hybrid G-WOA starts to search for the best solution through a number of iterations (t).

Algorithm 2: The proposed hybrid G-WOA Algorithm for ontology learning from Arabic text

Input: A vector R assigns the document's mapped features.
//The G-WOA algorithm parameters:
$Pop_{Size} \leftarrow$ population size, $Cr \leftarrow$ crossover rate, $MR \leftarrow$ mutation rate, $E \leftarrow$ The stopping criterion, $h \leftarrow$ constant defines the logarithmic spiral shape, $r \leftarrow$ random variable, where $r \in [-1, 1]$, $K \rightarrow$coefficient vector of WOA, and $e \leftarrow$is linearly decreased from 2 to 0 along iterations (t).
//Fitness function parameters
$w_f \leftarrow$ weight of false alarm rate, $w_d \leftarrow$ weight of detection rate, and $w_c \leftarrow$ selected features weight.
Output: R^* the solution with the optimal concept/semantic relation set contributing to the ontology.

Represent each document $d_1, d_2, \ldots, d_O$ by a single whale c to obtain a pool O of whales. $C = \{c_1, c_2, \ldots, c_O\}$.
Evaluate the fitness for each whale $c_i \in C$ using Equation (21).
Get the best individual c_{best} and set it as c_G^0.
Initialize the counter of iteration, $t = 1$
While (stopping criterion E is not met)
  $\{\widetilde{C} \Leftarrow \varphi$
    For each $p \leftarrow 1$ to Pop_{Size}
      Choose a random integer u_{rand} from $\{1, 2, \ldots U\}$.
      Randomly select two whales $c^t_{rand1,\,j}, c^t_{rand2,\,j} \in C$ ($c_{rand1} \neq c_{rand2}$):
      Update K, A, e, h , and r.
      For each gene j in the solution $c_{p,\,j}$
      IF $rand\ (0, 1) \leq Cr$ or $j == u_{rand}$ **Then**
        $offspring^t_{p,j} = c^t_{rand1,\,j} + M_{p,\,j}\left(c^t_{rand2,\,j} - c^t_{rand1,\,j}\right)$.
      ELSE
        IF $m < 0.5$ **THEN**
          IF $|K| \geq 1$ **THEN**
            Choose a random individual $c^t_{rand} \neq c^t_p$.
            $S^t{}_{p,\,j} = \left| K^t_p . c^t_{rand,\,j} - c^t_{p,\,j} \right|$.
            $offspring^t_{p,\,j} = c^t_{G,\,j} - A^t_p . S^t_{p,\,j}$.
          ELSE
            $S^t{}_{p,\,j} = \left| K^t_p . c^t_{G,\,j} - c^t_{p,\,j} \right|$
            $offspring^t_{p,\,j} = c^t_{G,\,j} - A^t_p . S^t_{p,\,j}$
          End IF
        ELSE
          $offspring^t_{p,\,j} = \left| c^t_{G,\,j} - c^t_{p,\,j} \right| . exp\ (hr) . cos(2\,\pi\, r) + c^t_{G,j}$
        End IF
      End IF
    End For
  End For
Evaluate the fitness of the offspring $offspring^t_p$.
Return to population.
For each $p \leftarrow 1$ to Pop_{Size}
  IF $offspring^t_p \leq MR$ **THEN**
    Replace c^t_p with $offspring^t_p$.

Algorithm 2: The proposed hybrid G-WOA Algorithm for ontology learning from Arabic text

1. **End IF**
2. **End For**
3. Choose the best individual c^t_{best} among the updated population.
4. **IF** $c^t_{best} \leq MR$ **THEN**
5. Replace c^t_G with c^t_{best}.
6. **End IF**
7. $t = t + 1$
8. **End while**

4.3.2. Fitness Evaluation

An internal classifier was used to evaluate the fitness value of each agent (whale). In this work, it was proven that the SVM showed the best performance among the other classifiers. We used fitness function for measuring each agent's false alarm rate, detection rate, and the number of concepts selected in each iteration until reaching the best solution. The optimal solution will be the one that decreases the False Alarm Rate (FAR), increases the Detection Rate (DR), and decreases the number of selected concepts. A standalone weighted fitness function was used to deal with this Multi-Criteria Decision Making. Three weights w_f, w_d, and w_c were used to define FAR, DR, and the number of selected features, respectively.

$$Fitness = w_f[False\ \ Alarm\ \ Rate] + w_d[Detection\ \ Rate] + w_c[F] \tag{21}$$

where

$$False\ \ Alarm\ \ Rate\ \ (FAR) = \frac{False\ \ Positive}{False\ \ Positive + True\ \ Negative} \tag{22}$$

$$Detection\ \ Rate\ \ (DR) = \frac{True\ \ Positive}{True\ \ Positive + False\ \ Negative} \tag{23}$$

$$F = \left[1 - \frac{\sum_{k=1}^{M} G_k}{M} \right] \tag{24}$$

$$G_k \leftarrow \begin{cases} 0 & if\ the\ concept\ is\ selected\ through\ selecting\ its\ representative\ gene\ of\ the\ whale \\ 1 & if\ the\ concept\ is\ neglected\ through\ neglecting\ its\ representative\ gene\ of\ the\ whale \end{cases} \tag{25}$$

$M \leftarrow$ Number of concepts.

4.3.3. Mutation

The mutation operator, which is the core of the G-WOA algorithm, was used to produce a mutant vector. In this regard, a mutation rate MR is defined as a prerequisite. If the gene of the picked solution is lower than the MR value, then the algorithm will mutate each gene within the parent solution using Equation (26). Where $offspring^t_{p,j}$ is the new generated solution $c^t_{rand1,\ j}$ and $c^t_{rand2,\ j}$ are two randomly selected parents, $M_{p,\ j}$ is a random value in the range $[0,\ 1]$, t denotes the current iteration number, and p represents the whale number.

$$offspring^t_{p,j} = c^t_{rand1,\ j} + M_{p,\ j}\left(c^t_{rand2,\ j} - c^t_{rand1,\ j}\right) \tag{26}$$

4.3.4. Crossover

In the encircle prey phase, the uniform crossover operator is performed between the mutant vector, namely, $offspring^t_{g_{p,\ j}}$, and a randomly selected solution $c^t_{rand,\ j}$. The ordinary WOA algorithm uses a random variable to compute the distance between the best whale and the search agent without considering the fitness value for neither the current solution nor the functioned one. On the contrary, the G-WOA implements the crossover operator of GA in the encircle prey phase so that it selects a neighbor solution around the optimal solution. The crossover rate *Cr* is defined as a parameter for the G-WOA algorithm. The parent solution is integrated with the neighbor solution to generate the child based on the *Cr* value, using the following equation:

$$offspring^t_{p,\ j} = \begin{bmatrix} offspring^t_{p,\ j} & rand\ (0,\ 1)\ \leq\ Cr\ or\ j\ ==\ u_{rand} \\ c^t_{G,\ j}\ -\ A^t_p.\ S^t_{p,\ j} & rand\ (0,\ 1)\ >\ Cr\ or\ j\ ==\ u_{rand}\ \&\ m < 0.5\ \&\ |K|\ \geq\ 1 \\ c^t_{G,\ j}\ -\ A^t_p.\ S^t_{p,\ j} & rand\ (0,\ 1)\ >\ Cr\ or\ j\ ==\ u_{rand}\ \&\ m < 0.5\ \&\ |K|\ <\ 1 \\ \left|c^t_{G,\ j}\ -\ c^t_{p,\ j}\right|.\ exp\ (hr).\ cos(2\ \pi\ r)\ +\ c^t_{G,j} & rand\ (0,\ 1)\ >\ Cr\ or\ j\ ==\ u_{rand}\ \&\ m \geq 0.5 \end{bmatrix} \quad (27)$$

4.3.5. Selection

The selection operator was implemented in G-WOA to determine if the target or offspring survived to the following iteration. The selection operator in G-WOA is expressed as in Equations (28) and (29). If every gene value of the generated solution is higher than the mutation value, then the G-WOA will replace the parent solution with the generated one. This comparison will be performed for each solution in the population. Then, the best solution is selected from the updated population based on the fitness value computed using Equation (21). The new best generated solution c^t_{best} will be replaced with the old one c^t_G if the each gene value of the best solution is lower than the mutation value:

$$c^t_p\ =\ \begin{bmatrix} offspring\ ^t_p & if\ \ offspring\ ^t_p\ \leq\ MR \\ c^t_p & otherwise \end{bmatrix} \quad (28)$$

$$c^t_G = \begin{bmatrix} c^t_{best} & if\ \ c^t_{best}\ \leq\ MR \\ c^t_G & otherwise \end{bmatrix}. \quad (29)$$

4.3.6. Termination Phase

In the G-WOA algorithm, the new position of *i th* individual in the following generation is the fittest one between parent c^t_p and child $offspring^t_p$. In this context, solutions should regard boundary constraints. In case the constraints are violated, Equation (30) can be used to apply the following repairing rule:

$$c_{p,\ j}\ =\ \begin{cases} u_j\ +\ rand(0,\ 1)\ \times\ (l_j\ -\ u_j) & if\ \ c_{p,\ j}\ <\ u_u \\ l_j\ +\ rand(0,\ 1)\ \times\ (l_j\ -\ u_j) & if\ \ c_{p,\ j}\ >\ l_u \end{cases} \quad (30)$$

where u_j and l_j represents upper and lower bounds of the solution's *j th* dimension, respectively. $c_{p,\ j}$ refers to the *j th* dimension of the *p th* solution. $rand(0,\ 1)$ represents a random number (between 0 and 1). Furthermore, the G-WOA algorithm checks the current iteration index. If the current iteration index reached the limit of the predefined criterion (E), then the new solutions generated are chosen, which are the solutions with the highest fitness. Then, the database is updated with the new solutions for Arabic ontology structure. Otherwise, the G-WOA algorithm will proceed the iteration process.

5. Experimental Results

This section discusses the validation results of the proposed approach for Arabic ontology learning based on text mining and G-WOA algorithms. Extensive experiments have been conducted using different bio-inspired optimization algorithms and over different Arabic corpora. Furthermore, to discuss and evaluate the how the proposed approach works for the non-Arabic setting, we applied

it to two publicly available non-Arabic corpora and compared the results to the state-of-the-art works that use the same corpora. The details of the experiments are illustrated in the following section.

5.1. Corpora

The Arabic corpora tested in this work are automatic content extraction (ACE) [72,73] corpora, ANERcorp [74,75] dataset, and a private corpus of Arabic biomedical texts. In the previously published computational linguistic work, the ACE and ANERcorp were frequently utilized for the purposes of evaluation and comparison with the existing systems. Three ACE corpora were investigated in this study: ACE 2003 (Broadcast News (BN), and Newswire (NW)), as well as ACE 2004 (NW). They are publicly available and were all tested by the proposed algorithm. For each dataset, the types of concepts (named entities) and their representation are demonstrated in Table 3. With the goal of identifying certain types of Arabic biomedical named entities in this work, we created a private corpus for evaluating the proposed approach of Arabic ontology learning. This task was accomplished by collecting a number of the Arabic open source texts in the biomedical domain, which were assessed by expert physicians. The private corpora information was illustrated in Table 3, where we represent each class in each Arabic domain by a number of documents that contained the number of unique words the concept mining and ontology learning algorithms will operate on.

Table 3. Information of the Arabic corpora tested in this work.

	Corpus					**Total**
	ACE 2003 (BN)	**ACE 2003 (NW)**	**ACE 2004 (NW)**	**ANERcorp**	**Private Corpus**	
Person	517	711	1865	3602		6695
Date	20	58	357	-		435
Time	1	15	28	-		44
Price	3	17	105	-		125
Measurement	14	28	51	-		93
Percent	3	35	54			92
Location	1073	1292	493	4425		7283
Organization	181	493	1313	2025		4012
Healthcare Provider	-	-	-	-	8097	8097
Health Disorder	-	-	-	-	13,502	13,502
Cancers	-	-	-	-	9072	9072
Surgeries	-	-	-	-	7055	7055

Furthermore, the non-Arabic corpora tested in this work include two publicly available ones that belong to the biomedical domain and are related to the protein–protein interactions. These corpora are Learning Language in Logic (LLL) [76] and the Interaction Extraction Performance Assessment (IEPA) [77]. The LLL corpus presents the task of gene interaction from a group of sentences related to *Bacillus subtilis* transcription. The IEPA dataset comprises 303 abstracts obtained from the repository of PubMed, each one including a particular pair of co-occurring chemicals.

5.2. Performance Measures

The performance validation measures used in this paper are precision (*PRE*), recall (*REC*), and F-score (*F*), [58]. The F-score is used in information retrieval to represent the harmonic incorporation of the values computed from precision (*PRE*), and recall (*REC*) measures. These metrics were calculated for each *k*-fold using Equations (31)–(33), then we finally estimated the overall average of their values:

$$PRE = \frac{True\ \ Positive}{True\ \ Positive\ \ +\ \ False\ \ Positive} \tag{31}$$

$$REC = \frac{True\ \ Positive}{True\ \ Positive\ \ +\ \ False\ \ Negative} \tag{32}$$

$$F = \frac{2\ True\ \ Positive}{2\ True\ \ Positive\ +\ False\ \ Positive\ +\ False\ \ Negative}. \quad (33)$$

5.3. Cross Validation

In this work, we used *k*-fold cross-validation to evaluate the quality of the solution obtained using the G-WOA algorithm, in which *k* is equivalent to ten. Each corpus was randomly separated into ten sub-samples which were equally sized. From each corpus, a single sub-sample was set as a validation set so that it was used in performance testing, then the $k - 1$ sub-samples were employed as a training set. This procedure was repeated 10 times. In each fold, each *k* sub-sample was employed exactly once as the validation set. The *k*-outcomes of the folds were then averaged so that they provided a single rating.

5.4. Discussion

5.4.1. Comparison to the State-of-the-Art

The comparison to the state-of-the-art was composed of three experiments: (1) comparisons with the other bio-inspired optimization algorithms existing in the literature regarding Arabic ontology learning, (2) comparisons with the previously published approaches on Arabic ontology learning from the text, and (3) comparisons with the state-of-the-art on learning ontology from non-Arabic settings. Firstly, to validate the performance of the proposed G-WOA algorithm in learning ontology from Arabic text, we compared the solution results returned by it to those returned by the ordinary GA and WOA. Moreover, extensive comparisons were conducted by comparing the performance of the G-WOA algorithm to three other bio-inspired algorithms: PSO [44], moth flame optimization (MFO) [45], and the hybrid differential evolution-whale optimization (DE-WOA) [46]. To compare these bio-inspired algorithms, the parameter setting had to be determined for each. Table 4 presents the parameter list used in this work, which was taken from [32,37,44,78]. In each experiment, the tested algorithm was first implemented into one of the previously mentioned corpora. Then, the measures of *PRE*, *REC*, and *F* were computed using Equations (31)–(33). This process was repeated for each dataset. Then, the average values of the three measures across all datasets were computed.

For each algorithm of the G-WOA, GA, WOA, PSO, MFO, and DE-WOA, we demonstrated the detailed validation results obtained across all the investigated Arabic corpora. The results are demonstrated in Tables A1–A6 of Appendix A, respectively. To sum up, Table 5 presents the total average measures of each algorithm across all the corpora. From Tables 5 and A1, Tables A2–A6, it is apparent that the proposed G-WOA algorithm outperformed the other algorithms in all folds and across all the datasets. The *PRE*, *REC*, and *F* results provided by the hybrid G-WOA algorithm were higher when compared to those from the ordinary GA, WOA, PSO, MFO, and the hybrid DE-WOA algorithm. Taking the ACE 2003 (BN) as an example, the results of *PRE*, *REC*, and *F* were 98.14%, 99.03%, and 98.59%, respectively. The F-score (*F*) values obtained using the G-WOA were also 98.79%, 98.44%, 98.57%, and 98.63% for ACE 2003 (NW), ACE 2004 (NW), ANERcorp, and the private corpus.

Compared to the GA algorithm, the obtained F-score (*F*) values were 93.75%, 93.63%, 93.73%, 93.65%, and 93.41% for the ACE 2003 (BN), ACE 2003 (NW), ACE 2004 (NW), ANERcorp, and the private corpus. On the other hand, the WOA achieved F-score (*F*) values of 96.66%, 96.87%, 96.79%, 96.94%, and 96.81%, respectively for the aforementioned corpora. From these results, the improvement in F-score (*F*) of the ontology learning in comparison to the basic GA algorithm was 4.84%, 5.16%, 4.71%, 4.92%, and 5.22%, respectively, for the same corpora. In comparison to the ordinary WOA, the improvement reached 1.93%, 1.92%, 1.65%, 1.63%, 1.82%, respectively, for the five corpora. These results indicate that the G-WOA was able to accelerate the process of global searching when learning ontology, with its ability to balance effectively both exploration and exploitation.

Table 4. The parameter list used in this work.

GA	WOA	PSO	MFO	GA-WOA	DE-WOA
Population size: 100	Population size: 100	Particles number P: 10	Population size: 100	Population size: 100	Population size: 100
Maximum generations: 500	The random variable $r : [-1, 1]$	Iterations number t: 10	The constant defines the logarithmic spiral shape h: 1	Maximum generations: 500	Scaling factor for DE: Random between 0.2 and 0.8
Crossover probability Cr: 0.9	Logarithmic spiral shape h: 1	Acceleration $c1$: 2	The random variable $r : [-1, 1]$	Crossover probability Cr: 0.9	DE mutation scheme: DE/best/1/bin
Mutation rate MR: 0.05	e: Decreased from 2 to 0	Acceleration $c2$: 2		Mutation rate MR: 0.05	The random variable $r : [-1, 1]$
Reproduction ratio: 0.18		Maximal weight of inertia: 0.7		Maximum iterations number: 10	Logarithmic spiral shape h: 1
Selection: weighted Roulette Wheel		Minimal weight of inertia: 0.1		The random variable $r : [-1, 1]$	e: Decreased from 2 to 0
				Logarithmic spiral shape h: 1	
				e: Decreased from 2 to 0	

Table 5. Average measures for each algorithm across all datasets (detailed results of algorithms can be followed in Tables A1–A6 of the Appendix A).

Algorithm	Average Measures		
	PRE (%)	*REC* (%)	*F* (%)
Proposed GA-WOA	98.48	98.73	98.6
GA	93.71	93.56	93.63
WOA	96.31	97.34	96.81
PSO	95.04	94.95	94.99
MFO	97.73	96.96	97.34
DE-WOA	97.07	95.93	96.57

Furthermore, the results also show that the MFO algorithm was more optimal than WOA in ontology learning. This is due to the good ability of MFO to switch between both exploration and exploitation, contrary to the WOA, which was trapped early in local optima throughout the optimization process [37]. Therefore, the MFO algorithm occupies the second rank after the G-WOA in terms of Arabic ontology learning. On the other side, the DE-WOA has a lower performance than G-WOA and MFO across all the datasets. Although the DE algorithm had robust global searchability, it was weak in the exploitation, and converged slowly. Thus, the DE algorithm needs to be optimized for it to be hybridized with other algorithms, as reported in [46]. Thus, the DE-WOA has the third rank in terms of the Arabic ontology learning.

In contrast, the convergence speed is a crucial criterion for evaluating the performance of any optimization method. Therefore, the convergence time for the proposed hybrid G-WOA algorithm was computed and compared to the time obtained by all algorithms versus the false alarms rate. The false alarms rate was computed in this paper using Equation (22). As depicted in Figures 2 and A1 (of

Appendix A), when following the WOA algorithm across all the Arabic datasets, we see that it took a lower convergence speed in comparison to the proposed hybrid G-WOA algorithm. This can be interpreted by the poor exploitation ability for the ordinary WOA algorithm, which requires a long time to search for the offspring and parents. On the contrary, the hybrid G-WOA algorithm overcame this drawback by combining the genetic operations into the WOA algorithm.

Secondly, to investigate the efficacy of the proposed approach that integrates the text mining and G-WOA algorithms to learn ontology from the Arabic text, we performed a comparison between it and the more recent works that use the same Arabic corpora, in terms of precision (*PRE*), recall (*REC*), and F-score (*F*). Table 6 shows the comparison. Compared to the other methods presented in the literature, as Table 6 shows, the proposed approach yielded superior results in terms of *PRE*, *REC*, and *F* measures. These results demonstrate the robustness of integrating text mining and G-WOA algorithms.

Table 6. Comparison to the state-of-the-art on Arabic ontology learning.

Reference	Year	Corpus	Approach	Accuracy
[79]	2019	ACE 2003 (NW), ACE 2003 (BN), ACE 2004 (NW), and ANERcorp.	The Long-Short-Term-Memory neural tagging model was augmented with the Convolutional Neural Network to extract the character-level features.	*F* = 91.2%, 94.12%, 91.47%, and 88.77%, respectively, for the four corpora.
[78]	2018	ANERcorp	A deep neural network-based method.	*PRE* = 95.76%, *REC* = 82.52%, and *F* = 88.64%.
[80]	2018	ANERcorp	Integration of some tree and polynomial kernels for feature representation. The universal dependency parsing was used for the relation extraction.	*F* = 63.5%.
[64]	2016	ANERcorp	A hybrid approach of the GA and SVM.	*F* = 82%.
[81]	2016	ACE 2003 (NW), ACE 2003 (BN), ACE 2004 (NW), and ANERcorp.	Hybridization of the rule-based and machine learning approaches. The feature space comprised the language-specific and language independent features. The decision tree classifier was used as a machine learning algorithm.	*PRE* = 92.7%, *REC* = 88.1%, and *F* = 90.3% for the ACE 2003 (BN), while they are 92.9%, 93.4%, 93.2%, for the ACE 2003 (NW), respectively. *PRE* = 82.8%, *REC* = 82%, and 82.4%, for the ACE 2004 (NW), while they are 94.9%, 94.2%, and 94.5%, respectively, for the ANERcorp.
[82]	2013	ACE 2003 (NW), ACE 2003 (BN), ACE 2004 (NW), and ANERcorp.	Hybrid rule-based/machine learning approach. The features comprised: Rule-based features estimated from the rule-based decisions, morphological features derived from morphological analysis of decisions, POS features, contextual features, Gazetteer features, and word-level features. The J48, Libsvm, and Logistic classifiers were used.	The highest results were achieved when applying the proposed method to the ANERcorp: *PRE* = 87%, *REC* = 60%, and *F* = 94%.
Proposed approach	2019		A text mining algorithm to extract the initial concept set from the Arabic documents. A proposed G-WOA algorithm to get the best solutions that optimize the ontology learning through selecting only the optimal concept set with their semantic relations, which contribute to the ontology structure.	*PRE* = 98.14%, *REC* = 99.03%, and *F* = 98.59%, for the ACE 2003 (BN) while their values are 99.27%, 98.32%, and 98.79%, respectively, for the ACE 2003 (NW). *PRE* = 97.9%, *REC* = 98.99%, and *F* = 98.44%, for the ACE 2004 (NW), while their values are 98.99%, 98.16%, and 98.57% for ANERcorp, and 98.12%, 99.15%, 98.63%, for the private corpus.

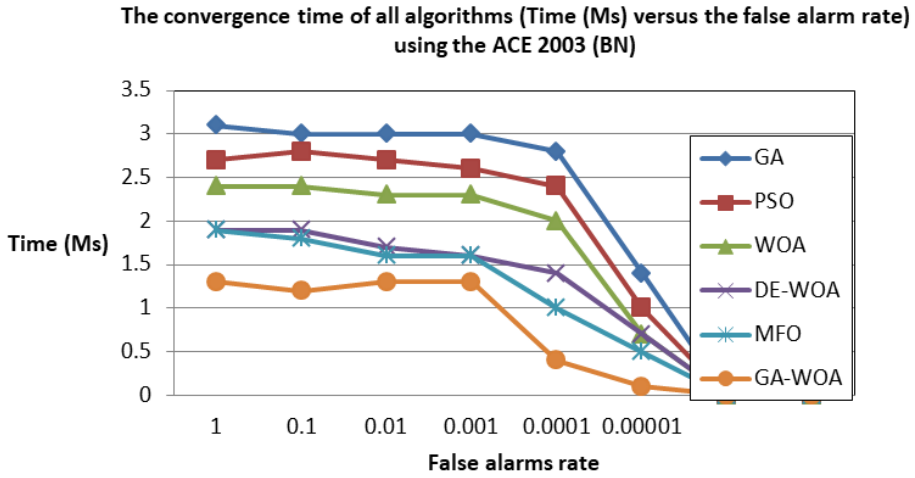


Figure 2. The convergence time versus the FAR rate for all algorithms using the ACE 2003 (BN) corpus.

Thirdly, to test the efficiency of the proposed approach to learn ontology from the non-Arabic text, we applied it to the two aforementioned publicly available corpora. Furthermore, we compared its performance to the other approaches presented in the literature to learn ontology from the non-Arabic text, in terms of *PRE*, *REC*, and *F* measures. The comparison is shown in Table 7, while the application results are demonstrated in Table A7 of Appendix A. The results demonstrate that the proposed G-WOA achieved superior results when applied to the non-Arabic corpora.

Table 7. Comparison to the state-of-the-art in non-Arabic settings.

Reference	Year	Corpus	Approach	Accuracy
[83]	2019	LLL and IEPA	A logic-based relational learning method for Relation Extraction utilizing the Inductive Logic Programming, namely OntoILPER, to generate symbolic extraction rules.	*F* = 79.9% and 76.1%, respectively, for the two corpora.
[84]	2015	LLL and IEPA	An optimized tree kernel-based *Protein–protein* extraction approach. The modal verbs together with the appositive dependency features were used for defining some relevant rules that expand and optimize the shortest dependency path in between two proteins.	*F* = 82.3% and 68.7%, respectively, for the two corpora.
[85]	2015	LLL and IEPA	Three word representational techniques including vector clustering, distributed representation, and Brown clusters. The SVM was used for unsupervised learning.	*F* = 87.3% and 76.5%, respectively, for the two corpora.
[86]	2012	LLL and IEPA	Tree kernel-based extraction approach, in which the tree representation produced from constituent syntactic parser, was refined utilizing the shortest dependency route in between two proteins estimated from the dependency parser.	*F* = 84.6% and 69.8%, respectively, for the two corpora.
Proposed approach	2019	LLL and IEPA	Integrating text mining and G-WOA algorithms.	98.1% and 97.95%, respectively, for the two corpora.

5.4.2. Contributions to the Literature

From the previous results, the proposed approach outperformed the state-of-the-art approaches to Arabic ontology learning. Likewise, it was also noted that very little research has used the evolutionary approaches, whereas no hybrid bio-inspired algorithms were investigated by the previously published works. The majority of studies have depended on the hybridization of rule-based and machine learning approaches [4,23], which have shortcomings, as previously discussed in the introduction section.

Some works used deep learning algorithms [78,79] like the long-short-term-memory and convolutional neural network, but the results are still below expectation. In [78], a deep neural network-based method was proposed. The application results to the ANERcorp were 95.76%, 82.52%, and 88.64%, in terms of *PRE*, *REC*, and *F* measures. These results are also lower than those obtained using the approach proposed in this work. In [79], the *F* measure results obtained using the presented deep learning approach were 91.2%, 94.12%, 91.47%, and 88.77%, respectively, for the ACE 2003 (NW), ACE 2003 (BN), ACE 2004 (NW), and ANERcorp. These results are also lower than those obtained by applying the approach proposed in this work to the same corpora, which reveals the efficiency of our approach. The results presented in [78,79] reveal the need for enhancing the performance of the deep learning methods and to overcome their shortcomings such as being stuck in the local optima, when applied to the natural language processing, for instance, through using the bio-inspired optimization algorithms.

The proposed ontology learning approach is also applicable with non-Arabic texts. Furthermore, the comparisons to the state-of-the-art approaches on learning ontology using the same non-Arabic corpora demonstrate higher results in favor of the proposed approach. These results confirm that the proposed approach outperforms the state-of-the-art methods on learning ontology from the non-Arabic texts.

5.4.3. Implications for Practice

As for ontology learning using the G-WOA algorithm, the contributions of GA and WOA enabled the GA-WOA to jump out easily of the local minima. Accordingly, it found a promising search direction toward global optimization. Specifically, the G-WOA algorithm has a robust capability to attain equilibrium between both global and local exploitation. Therefore, the proposed hybrid G-WOA algorithm outperformed the other compared algorithms in terms of speed.

The implications for practice show that the synergy of text mining and G-WOA algorithms can operate on either the Arabic or non-Arabic document by extracting the concepts and their semantic relations and then providing the solutions with the best set of concepts between the initial one. The obtained solutions can optimize the ontology construction from the Arabic or the non-Arabic text by returning only the important concepts that contribute to the ontology structure while ignoring the redundant or less important ones.

6. Conclusions

The majority of the state-of-the-art works on Arabic ontology learning from texts have depended on the hybridization of the handcrafted rules and machine learning algorithms. Contrary to the literature, this study presented a novel approach for Arabic ontology learning from texts which advances the state-of-the-art in two ways. First, a text mining algorithm was proposed for extracting the initial concept set from the text documents together with their semantic relations. Secondly, a hybrid G-WOA was proposed to optimize the ontology learning from Arabic text. The G-WOA integrates the genetic search operators like mutation, crossover, and selection into the WOA algorithm to achieve the equilibrium between both exploration and exploitation, in order to find the best solutions that exhibit the highest fitness. The experimental results revealed the following conclusions.

Firstly, as for learning Arabic ontology from texts, the proposed GA-WOA outperformed the ordinary GA, and WOA across all the Arabic datasets in terms of *PRE*, *REC*, and *F* measures. When comparing the solution results obtained using the G-WOA to those obtained using the ordinary GA, we found an improvement in F-score (*F*) by up to 4.84%, 5.16%, 4.71%, 4.92%, and 5.22%, respectively, for ACE 2003 (NW), ACE 2004 (NW), ANERcorp, and the private corpus. Furthermore, the improvement reached 1.93%, 1.92%, 1.65%, 1.63%, 1.82%, respectively for the same corpora when using the ordinary WOA algorithm. Secondly, the G-WOA also outperformed the PSO, DE-WOA, and MFO across all the Arabic corpora, in terms of the three measures. The MFO occupies the second rank after the G-WOA, in terms of ontology learning from Arabic text. This was interpreted by the good ability of MFO to switch between both exploration and exploitation. Thirdly, the G-WOA

outperformed the other algorithms in convergence speed. Taking the WOA as an example, it is found to have low convergence due to its poor exploitation. Thus, the G-WOA algorithm is superior when compared to the other bio-inspired algorithms in terms of convergence speed.

Furthermore, the G-WOA exhibited low rates of false alarms across all the Arabic datasets, in comparison to the other algorithms. Fourthly, the proposed Arabic ontology learning approach, which is based on the synergy of text mining and G-WOA algorithms, outperformed the state-of-the-art in terms of precision (*PRE*), recall (*REC*), and F-score (*F*). This was due to its high capability to extract the concepts along with the semantic relations from the Arabic documents, then creating a population of search agents (solutions) that include genes represent the initial concepts. Moreover, the G-WOA starts to search for the best solution through a set of iterations, including embedding the genetic operators into the WOA architecture. Eventually, the algorithm returns the solution which recommends the best set of concepts/relations that can contribute to the ontology. Eventually, the proposed ontology learning approach is applicable to the non-Arabic texts. It achieved higher performance that outperformed the state-of-the-art approaches on learning ontology from the non-Arabic text.

Limitations and Future Research Directions

The proposed approach for Arabic ontology learning cannot deal with learning the hierarchical feature representation from the text. One advantage of the deep learning algorithms is that they are able to generate high-level feature representation from raw texts directly. Therefore, we tend to present a deep neural network model using latent features to improve learning ontology from Arabic texts. The proposed model will work on embedding the words and positions as latent features, therefore, it will not rely on feature engineering. To overcome the limitations of the deep network model, such as being stuck in the local optima, different bio-inspired optimization algorithms will be tested and compared in this regard.

Author Contributions: Conceptualization, R.M.G.; data curation, R.M.G., N.A., and K.S.; formal analysis, R.M.G., N.A., and K.S.; funding acquisition, N.A.; investigation, R.M.G.; methodology, R.M.G.; project administration, N.A. and K.S.; resources, R.M.G.; software, R.M.G.; supervision, R.M.G. and K.S.; validation, R.M.G.; visualization, R.M.G.; writing —original draft, R.M.G.; writing—review & editing, R.M.G. and K.S.

Funding: This research project was funded by the Deanship of Scientific Research at Princess Nourah bint Abdulrahman University, through the Research Funding Program.

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

Appendix A : Tables of Results

This section presents the tables and figures that summarize the application results of the proposed hybrid G-WOA and the other bio-inspired algorithms to the Arabic and non-Arabic corpora.

Table A1. Performance evaluation of ontology learning using the proposed G-WOA over the five corpora.

Hybrid G-WOA															
Fold	**ACE 2003 (BN)**			**ACE 2003 (NW)**			**ACE 2004 (NW)**			**ANERcorp**			**Private Corpus**		
	PRE (%)	*REC* (%)	*F* (%)	*PRE* (%)	*REC* (%)	*F* (%)	*PRE* (%)	*REC* (%)	*F* (%)	*PRE* (%)	*REC* (%)	*F* (%)	*PRE* (%)	*REC* (%)	*F* (%)
1	99	99.12	99.06	99.42	98.84	99.13	97.78	98.55	98.16	98.49	98.95	98.72	98.83	99.01	98.92
2	98.3	99.04	98.67	99.53	98.66	99.09	98.86	99.78	99.32	98.57	97.05	97.8	98.06	98.72	98.39
3	97.93	99.79	98.85	99.31	98.98	99.14	97.09	99.39	98.23	98.82	97.89	98.35	97.03	99.65	98.32
4	97.61	99.11	98.35	99.28	97.68	98.47	98.39	99.19	98.79	98.76	98.96	98.86	98.52	98.97	98.74
5	98.38	98.21	98.29	99.4	98.45	98.92	98.54	98.36	98.45	99.18	98.24	98.71	97.97	99.55	98.75
6	98.09	98.11	98.1	98.5	97.31	97.9	97.94	98.21	98.07	99.28	98.24	98.76	97.32	99.43	98.36
7	97.76	99.28	98.51	99.94	99	99.47	97.05	99.47	98.25	99.82	97.52	98.66	98.57	99.11	98.84
8	97.8	98.94	98.37	99.74	98.11	98.92	97.14	99.54	98.33	98.99	98.76	98.87	97.83	98.76	98.29
9	98.67	99.52	99.09	98.52	97.95	98.23	98.66	98.2	98.43	98.13	98.43	98.28	98.48	98.44	98.46
10	97.9	99.22	98.56	99.04	98.22	98.63	97.51	99.2	98.35	99.89	97.56	98.71	98.62	99.86	99.24
Average	98.14	99.03	98.59	99.27	98.32	98.79	97.9	98.99	98.44	98.99	98.16	98.57	98.12	99.15	98.63

Table A2. Performance evaluation of ontology learning using the GA over the five corpora.

GA															
Fold	**ACE 2003 (BN)**			**ACE 2003 (NW)**			**ACE 2004 (NW)**			**ANERcorp**			**Private Corpus**		
	PRE (%)	*REC* (%)	*F* (%)	*PRE* (%)	*REC* (%)	*F* (%)	*PRE* (%)	*REC* (%)	*F* (%)	*PRE* (%)	*REC* (%)	*F* (%)	*PRE* (%)	*REC* (%)	*F* (%)
1	93.08	93.87	93.48	93.2	93.52	93.36	93.83	94.96	94.4	93.71	92.44	93.08	93.82	93.29	93.56
2	93.23	93.75	93.49	94.05	92.16	93.1	93.14	93.41	93.28	93.89	93.83	93.86	93.2	93.77	93.49
3	92.19	93.52	92.86	94.92	93.08	94	93.31	95.47	94.38	93.22	92.32	92.77	93.96	92.65	93.31
4	93.31	93.56	93.44	93.61	93.27	93.44	92.38	94.38	93.37	95.36	93.03	94.19	93.81	92.22	93.01
5	93.14	95.71	94.41	93.01	93.95	93.48	92.67	94.65	93.65	95.45	93.62	94.53	94.74	92.09	93.4
6	92.55	93.88	93.22	95.48	92.38	93.91	93.51	94.23	93.87	94.7	92.73	93.71	93.4	93.72	93.56
7	93.17	95.97	94.55	94.2	92.68	93.44	93.45	94.27	93.86	93.51	93.22	93.37	94.12	92.66	93.39
8	93.51	93.61	93.56	95.8	92.29	94.02	93.29	94.48	93.89	93.18	93.6	93.39	93.38	93.4	93.39
9	92.29	95.96	94.09	94.48	92.97	93.72	92.32	94.14	93.23	94.83	92.24	93.52	93.52	92.05	92.78
10	93.25	95.48	94.36	94.41	93.18	93.8	92.78	93.99	93.39	94.55	93.5	94.03	95.37	92.98	94.16
Average	92.97	94.53	93.75	94.32	92.95	93.63	93.07	94.40	93.73	94.24	93.05	93.65	93.93	92.88	93.41

Table A3. Performance evaluation of ontology learning using the WOA over the five corpora.

WOA															
Fold	**ACE 2003 (BN)**			**ACE 2003 (NW)**			**ACE 2004 (NW)**			**ANERcorp**			**Private Corpus**		
	PRE (%)	*REC* (%)	*F* (%)	*PRE* (%)	*REC* (%)	*F* (%)	*PRE* (%)	*REC* (%)	*F* (%)	*PRE* (%)	*REC* (%)	*F* (%)	*PRE* (%)	*REC* (%)	*F* (%)
1	95.25	97.16	96.2	97.44	95.87	96.65	95.43	98.09	96.74	96	97.6	96.79	96.09	97.02	96.55
2	96.29	97.03	96.66	97.86	95.74	96.79	96.22	98.03	97.12	95.02	97.25	96.12	95.77	97.2	96.48
3	96.76	97.28	97.02	97.24	95.02	96.12	95.49	97.7	96.58	96.09	97.53	96.8	95.89	97.09	96.49
4	96.16	97.78	96.96	98.64	95.74	97.17	95.38	97.84	96.59	96.32	97.19	96.75	95.13	97.31	96.21
5	95.32	97.82	96.55	98.58	95.65	97.09	96.11	97.73	96.91	95.64	97.12	96.37	95.96	98.19	97.06
6	95.55	97.57	96.55	97.07	96.26	96.66	95.08	97.37	96.21	96.38	97.07	96.72	95.63	98.55	97.07
7	96.44	97	96.72	97.06	95.39	96.22	96.82	97.53	97.17	96.76	98.79	97.76	96.01	97.82	96.91
8	95.34	97.77	96.54	98.99	95.65	97.29	95.75	97.46	96.6	95.27	98.01	96.62	96.97	98.45	97.7
9	95.53	97.43	96.47	99	96.1	97.53	95.46	98.97	97.18	96.43	98.97	97.68	95.32	98.8	97.03
10	96.57	97.28	96.92	97.93	96.41	97.16	95.71	97.83	96.76	96.57	98.99	97.77	95.67	97.62	96.64
Average	95.92	97.41	96.66	97.98	95.78	96.87	95.75	97.86	96.79	96.05	97.85	96.94	95.84	97.81	96.81

Table A4. Performance evaluation of ontology learning using the PSO over the five corpora.

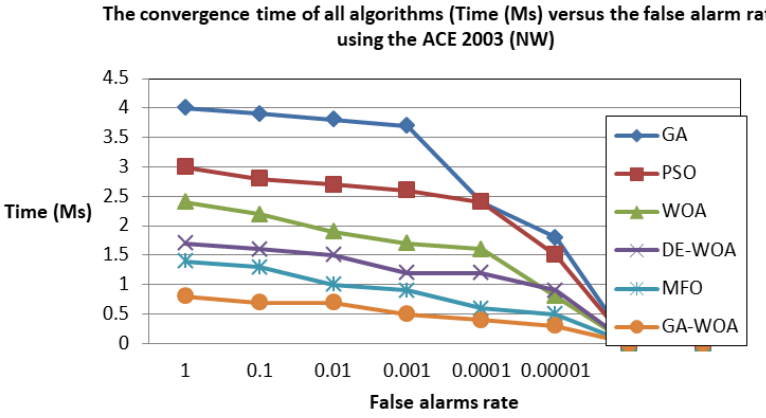
PSO															
Fold	**ACE 2003 (BN)**			**ACE 2003 (NW)**			**ACE 2004 (NW)**			**ANERcorp**			**Private Corpus**		
	PRE (%)	*REC* (%)	*F* (%)	*PRE* (%)	*REC* (%)	*F* (%)	*PRE* (%)	*REC* (%)	*F* (%)	*PRE* (%)	*REC* (%)	*F* (%)	*PRE* (%)	*REC* (%)	*F* (%)
1	95.79	95.7	95.75	94.37	94.09	94.23	94.63	94.77	94.7	95.88	94.15	95.01	94.01	95.41	94.71
2	94.46	95.5	94.98	95.71	94.17	94.94	94.41	95.66	95.04	95.42	94.41	94.92	94.15	95.4	94.78
3	95.35	94.06	94.71	94.03	95.93	94.98	95.76	95.77	95.77	94.1	94.64	94.37	95.39	95.1	95.25
4	94.16	94.47	94.32	94.86	95.46	95.16	95.85	95.18	95.52	95.45	95.53	95.49	95.98	95.2	95.59
5	95.25	95.03	95.14	95.91	94.59	95.25	95.51	94.24	94.88	95.45	94.33	94.89	95.33	95.4	95.37
6	94.54	95.59	95.07	95.95	94.64	95.3	95.6	94.75	95.18	94.47	94.31	94.39	94.97	95.91	95.44
7	96	94.87	95.44	95.72	94.72	95.22	94.27	94.92	94.6	94.47	95.25	94.86	95.7	95.1	95.4
8	94.19	94.1	94.15	94.13	94.17	94.15	94.8	94.9	94.85	94.12	95.21	94.67	95.57	94.15	94.86
9	94.29	95.46	94.88	95.75	95.89	95.82	94.36	94.53	94.45	94.79	95.36	95.08	94.34	94.1	94.22
10	95.47	95.78	95.63	95.55	95	95.28	95.4	94.29	94.85	95.88	95.85	95.87	94.25	94.22	94.24
Average	94.95	95.06	95	95.2	94.87	95.03	95.06	94.9	94.98	95	94.9	94.96	94.97	95	94.99

Table A5. Performance evaluation of ontology learning using the MFO over the five corpora.

DE-WOA															
Fold	**ACE 2003 (BN)**			**ACE 2003 (NW)**			**ACE 2004 (NW)**			**ANERcorp**			**Private Corpus**		
	PRE (%)	*REC* (%)	*F* (%)	*PRE* (%)	*REC* (%)	*F* (%)	*PRE* (%)	*REC* (%)	*F* (%)	*PRE* (%)	*REC* (%)	*F* (%)	*PRE* (%)	*REC* (%)	*F* (%)
1	97.36	96.35	96.86	97.2	96.71	96.96	97.33	96.74	97.04	98.21	97.24	97.73	98.9	96.43	97.65
2	97.42	96.01	96.71	97.58	97.69	97.64	98.58	97.04	97.81	98.45	97.63	98.04	97.5	96.03	96.76
3	97.02	97.33	97.18	97.96	96.16	97.06	97.06	97.01	97.04	98.91	97.87	98.39	98.59	97.9	98.25
4	97.25	97.21	97.23	97.07	97.3	97.19	97.62	97.75	97.69	97.37	96.61	96.99	97.8	97.64	97.72
5	97.84	97.14	97.49	97.26	97.45	97.36	97.48	96.15	96.82	97.47	97.27	97.37	97.73	97.15	97.44
6	97.42	96.87	97.15	97.99	96.95	97.47	98.43	97.45	97.94	97.72	97.86	97.79	98.25	97.19	97.72
7	97.49	96.14	96.82	97.01	97.24	97.13	98.9	96.74	97.81	98.08	96.35	97.21	97.2	96.16	96.68
8	97.32	96.73	97.03	97.56	96.89	97.23	97.53	96.23	96.88	98.82	96.58	97.69	97.8	97.67	97.74
9	97.27	96.7	96.99	97.07	96.09	96.58	98.32	97.03	97.68	97.31	97.9	97.61	98.36	96.49	97.42
10	97.27	96.43	96.85	97.88	96.1	96.99	97.13	96.61	96.87	97.01	97.9	97.46	98.48	97.61	98.05
Average	97.37	96.69	97.03	97.46	96.86	97.16	97.84	96.88	97.36	97.94	97.32	97.63	98.061	97.03	97.54

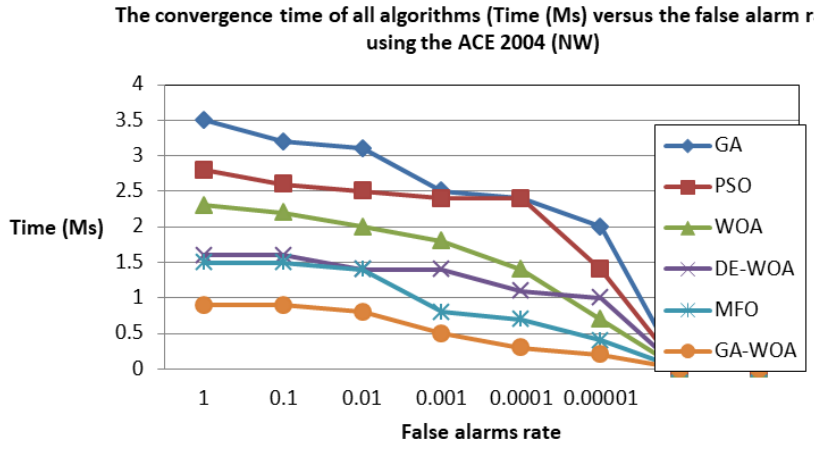
Table A6. Performance evaluation of ontology learning using the DE-WOA over the five corpora.

MFO															
Fold	**ACE 2003 (BN)**			**ACE 2003 (NW)**			**ACE 2004 (NW)**			**ANERcorp**			**Private Corpus**		
	PRE (%)	*REC* (%)	*F* (%)	*PRE* (%)	*REC* (%)	*F* (%)	*PRE* (%)	*REC* (%)	*F* (%)	*PRE* (%)	*REC* (%)	*F* (%)	*PRE* (%)	*REC* (%)	*F* (%)
1	97.12	95.28	96.2	96.57	95.46	96.02	97.43	95.16	96.29	96.33	95.15	95.74	98.13	96.36	97.24
2	97.27	95.16	96.21	97.21	95.69	96.45	96.27	95.06	95.67	97.09	96.39	96.74	97.23	96.9	97.07
3	97.29	95.5	96.39	97.74	95.95	96.84	97.71	96.51	97.11	96.3	96.53	96.42	97.5	96.03	96.76
4	97.89	95.5	96.69	96.35	96.25	96.3	96.08	95.11	95.6	96.64	95.43	96.04	98.9	95.51	97.18
5	97.1	95.39	96.24	97.57	95.15	96.35	97.71	96.25	96.98	97.37	95.72	96.54	97.75	96.37	97.06
6	97.05	96.53	96.79	96.48	96.74	96.61	97.44	96.52	96.98	97.4	96.31	96.86	98.24	96.17	97.2
7	97.47	95.16	96.31	96.74	95.85	96.3	96.58	96.22	96.4	97.54	95.48	96.5	98.34	95.2	96.75
8	97.07	95.8	96.44	97.16	95.66	96.41	97.67	96.32	97	96.95	96.12	96.54	97.1	95.22	96.16
9	97.1	96.44	96.77	96.61	96.24	96.43	97.91	96.43	97.17	97.81	96.11	96.96	97.29	95.84	96.56
10	97.71	96.96	97.34	96.22	96.81	96.52	96.31	96.45	96.38	96.44	96.9	96.67	97.5	95.13	96.31
Average	97.31	95.77	96.54	96.87	95.98	96.42	97.11	96	96.56	96.99	96.01	96.5	97. 8	95.87	96.83

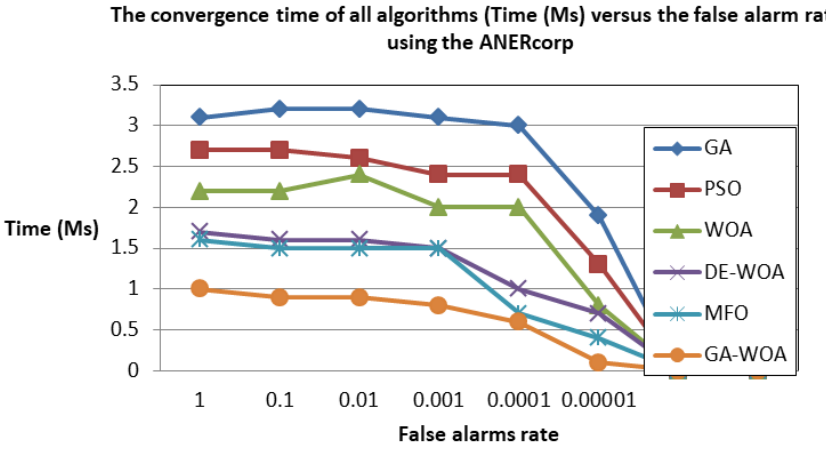


(**a**)

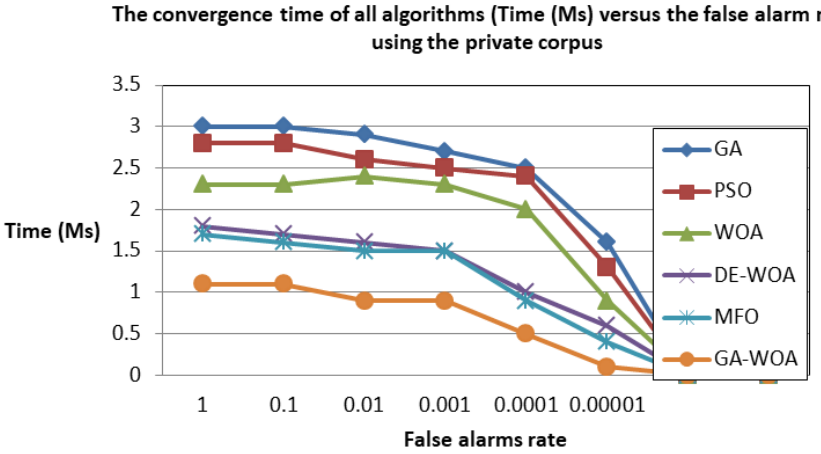
Figure A1. *Cont.*



(**b**)



(**c**)



(**d**)

Figure A1. The convergence time versus the FAR rate for all algorithms using (**a**) ACE 2003 (NW), (**b**) ACE 2004 (NW), (**c**) ANERcorp, and (**d**) the private corpus. Cont.

Table A7. Performance evaluation of ontology learning using the proposed G-WOA and the non-Arabic corpora.

Fold	Non-Arabic Corpus					
	LLL			IEPA		
	PRE (%)	*REC* (%)	*F* (%)	*PRE* (%)	*REC* (%)	*F* (%)
1	97.28	97.44	97.36	97.77	97.8	97.78
2	98.45	97.78	98.11	97.63	97.92	97.77
3	98.33	98.77	98.55	98.08	97.59	97.83
4	98.19	98.59	98.39	98.38	97.86	98.12
5	98.31	98.95	98.63	98.85	97.46	98.15
6	97.82	96.93	97.37	98.34	97.14	97.74
7	98.28	98.24	98.26	98.33	97.89	98.11
8	98.84	96.87	97.85	98.6	97.06	97.82
9	98.58	98.51	98.54	98.74	97.74	98.24
10	97.92	97.91	97.91	97.86	97.94	97.9
Average	98.2	98	98.1	98.26	97.64	97.95

References

1. Hawalah, A. A Framework for Building an Arabic Multi-disciplinary Ontology from Multiple Resources. *Cogn. Comput.* **2017**, *10*, 156–164. [CrossRef]
2. Al-Zoghby, A.M.; Elshiwi, A.; Atwan, A. Semantic Relations Extraction and Ontology Learning from Arabic Texts—A Survey. In *Intelligent Natural Language Processing: Trends and Applications Studies in Computational Intelligence*; Springer: Cham, Switzerland, 2017; pp. 199–225.
3. Mezghanni, I.B.; Gargouri, F. CrimAr: A Criminal Arabic Ontology for a Benchmark Based Evaluation. *Procedia Comput. Sci.* **2017**, *112*, 653–662. [CrossRef]
4. Mezghanni, I.B.; Gargouri, F. Deriving ontological semantic relations between Arabic compound nouns concepts. *J. King Saud Univ.-Comput. Inf. Sci.* **2017**, *29*, 212–228. [CrossRef]
5. Gruber, T.R. A translation approach to portable ontology specifications. *Knowl. Acquis.* **1993**, *5*, 199–220. [CrossRef]
6. Hazman, M.; El-Beltagy, S.R.; Rafea, A. A Survey of Ontology Learning Approaches. *Int. J. Comput. Appl.* **2011**, *22*, 36–43. [CrossRef]
7. Benaissa, B.-E.; Bouchiha, D.; Zouaoui, A.; Doumi, N. Building Ontology from Texts. *Procedia Comput. Sci.* **2015**, *73*, 7–15. [CrossRef]
8. Zamil, M.G.A.; Al-Radaideh, Q. Automatic extraction of ontological relations from Arabic text. *J. King Saud Univ.-Comput. Inf. Sci.* **2014**, *26*, 462–472. [CrossRef]
9. Benabdallah, A.; Abderrahim, M.A.; Abderrahim, M.E.-A. Extraction of terms and semantic relationships from Arabic texts for automatic construction of an ontology. *Int. J. Speech Technol.* **2017**, *20*, 289–296. [CrossRef]
10. Al-Zoghby, A.M.; Shaalan, K. Ontological Optimization for Latent Semantic Indexing of Arabic Corpus. *Procedia Comput. Sci.* **2018**, *142*, 206–213. [CrossRef]
11. Albukhitan, S.; Helmy, T.; Alnazer, A. Arabic ontology learning using deep learning. In Proceedings of the International Conference on Web Intelligence-WI 17, Leipzig, Germany, 23–26 August 2017.
12. Sadek, J.; Meziane, F. Extracting Arabic Causal Relations Using Linguistic Patterns. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.* **2016**, *15*, 1–20. [CrossRef]
13. Kaushik, N.; Chatterjee, N. Automatic relationship extraction from agricultural text for ontology construction. *Inf. Process. Agric.* **2018**, *5*, 60–73. [CrossRef]
14. Alami, N.; Meknassi, M.; En-Nahnahi, N. Enhancing unsupervised neural networks based text summarization with word embedding and ensemble learning. *Expert Syst. Appl.* **2019**, *123*, 195–211. [CrossRef]
15. Chi, N.-W.; Jin, Y.-H.; Hsieh, S.-H. Developing base domain ontology from a reference collection to aid information retrieval. *Autom. Constr.* **2019**, *100*, 180–189. [CrossRef]

16. Al-Arfaj, A.; Al-Salman, A. Towards Ontology Construction from Arabic Texts-A Proposed Framework. In Proceedings of the 2014 IEEE International Conference on Computer and Information Technology, Xi'an, China, 11–13 September 2014.
17. Al-Rajebah, N.I.; Al-Khalifa, H.S. Extracting Ontologies from Arabic Wikipedia: A Linguistic Approach. *Arab. J. Sci. Eng.* **2013**, *39*, 2749–2771. [CrossRef]
18. Albukhitan, S.; Alnazer, A.; Helmy, T. Semantic Web Annotation using Deep Learning with Arabic Morphology. *Procedia Comput. Sci.* **2019**, *151*, 385–392. [CrossRef]
19. Boujelben, I.; Jamoussi, S.; Hamadou, A.B. Enhancing Machine Learning Results for Semantic Relation Extraction. In *Natural Language Processing and Information Systems Lecture Notes in Computer Science*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 337–342.
20. Albarghothi, A.; Saber, W.; Shaalan, K. Automatic Construction of E-Government Services Ontology from Arabic Webpages. *Procedia Comput. Sci.* **2018**, *142*, 104–113. [CrossRef]
21. Bentrcia, R.; Zidat, S.; Marir, F. Extracting semantic relations from the Quranic Arabic based on Arabic conjunctive patterns. *J. King Saud Univ.-Comput. Inf. Sci.* **2018**, *30*, 382–390. [CrossRef]
22. Kramdi, S.E.; Haemmerl, O.; Hernandez, N. Approche générique pour l'xtraction de relations partir de texts. *Journées Francoph. D'ingénierie Des Connaiss.* **2009**, 97–108. Available online: https://hal.archives-ouvertes.fr/hal-00384415/document (accessed on 26 June 2019).
23. Boujelben, I.; Jamoussi, S.; Hamadou, A.B. A hybrid method for extracting relations between Arabic named entities. *J. King Saud Univ.-Comput. Inf. Sci.* **2014**, *26*, 425–440. [CrossRef]
24. Karimi, H.; Kamandi, A. A learning-based ontology alignment approach using inductive logic programming. *Expert Syst. Appl.* **2019**, *125*, 412–424. [CrossRef]
25. Juckett, D.A.; Kasten, E.P.; Davis, F.N.; Gostine, M. Concept detection using text exemplars aligned with a specialized ontology. *Data Knowl. Eng.* **2019**, *119*, 22–35. [CrossRef]
26. Petruccia, G.; Rospocher, M.; Ghidini, C. Expressive Ontology Learning as Neural Machine Translation. *SSRN Electron. J.* **2018**, *52–53*, 66–82. [CrossRef]
27. Luan, J.; Yao, Z.; Zhao, F.; Song, X. A novel method to solve supplier selection problem: Hybrid algorithm of genetic algorithm and ant colony optimization. *Math. Comput. Simul.* **2019**, *156*, 294–309. [CrossRef]
28. Alsaeedan, W.; Menai, M.E.B.; Al-Ahmadi, S. A hybrid genetic-ant colony optimization algorithm for the word sense disambiguation problem. *Inf. Sci.* **2017**, *417*, 20–38. [CrossRef]
29. Gaidhane, P.J.; Nigam, M.J. A hybrid grey wolf optimizer and artificial bee colony algorithm for enhancing the performance of complex systems. *J. Comput. Sci.* **2018**, *27*, 284–302. [CrossRef]
30. Elrehim, M.Z.A.; Eid, M.A.; Sayed, M.G. Structural optimization of concrete arch bridges using Genetic Algorithms. *Ain Shams Eng. J.* **2019**. [CrossRef]
31. Liu, P.; Basha, M.D.E.; Li, Y.; Xiao, Y.; Sanelli, P.C.; Fang, R. Deep Evolutionary Networks with Expedited Genetic Algorithms for Medical Image Denoising. *Med. Image Anal.* **2019**, *54*, 306–315. [CrossRef] [PubMed]
32. Ghoniem, R.M. Deep Genetic Algorithm-Based Voice Pathology Diagnostic System. In *Natural Language Processing and Information Systems Lecture Notes in Computer Science*; Springer: Cham, Switzerland, 2019; pp. 220–233.
33. Gupta, R.; Nanda, S.J.; Shukla, U.P. Cloud detection in satellite images using multi-objective social spider optimization. *Appl. Soft Comput.* **2019**, *79*, 203–226. [CrossRef]
34. Nguyen, T.T. A high performance social spider optimization algorithm for optimal power flow solution with single objective optimization. *Energy* **2019**, *171*, 218–240. [CrossRef]
35. Jayaprakash, A.; Keziselvavijila, C. Feature selection using Ant Colony Optimization (ACO) and Road Sign Detection and Recognition (RSDR) system. *Cogn. Syst. Res.* **2019**, *58*, 123–133. [CrossRef]
36. Chen, L.; Xiao, C.; Li, X.; Wang, Z.; Huo, S. A seismic fault recognition method based on ant colony optimization. *J. Appl. Geophys.* **2018**, *152*, 1–8. [CrossRef]
37. Aziz, M.A.E.; Ewees, A.A.; Hassanien, A.E. Whale Optimization Algorithm and Moth-Flame Optimization for multilevel thresholding image segmentation. *Expert Syst. Appl.* **2017**, *83*, 242–256. [CrossRef]
38. Elaziz, M.A.; Mirjalili, S. A hyper-heuristic for improving the initial population of whale optimization algorithm. *Knowl.-Based Syst.* **2019**, *172*, 42–63. [CrossRef]
39. Mirjalili, S.; Lewis, A. The Whale Optimization Algorithm. *Adv. Eng. Softw.* **2016**, *95*, 51–67. [CrossRef]

40. Goldbogen, J.A.; Friedlaender, A.S.; Calambokidis, J.; Mckenna, M.F.; Simon, M.; Nowacek, D.P. Integrative Approaches to the Study of Baleen Whale Diving Behavior, Feeding Performance, and Foraging Ecology. *BioScience* **2013**, *63*, 90–100. [CrossRef]
41. Habib, Y.; Sadiq, M.S.; Hakim, A. *Applications and Science of Neural Networks, Fuzzy Systems, and Evolutionary Computation*; Society of Photo Optical: Bellingham, WA, USA, 1998. [CrossRef]
42. Xue, X.; Chen, J. Using Compact Evolutionary Tabu Search algorithm for matching sensor ontologies. Using Compact Evolutionary Tabu Search algorithm for matching sensor ontologies. *Swarm Evol. Comput.* **2019**, *48*, 25–30. [CrossRef]
43. Afia, A.E.; Lalaoui, M.; Chiheb, R. A Self Controlled Simulated Annealing Algorithm using Hidden Markov Model State Classification. *Procedia Comput. Sci.* **2019**, *148*, 512–521. [CrossRef]
44. Ghoniem, R.M.; Shaalan, K. FCSR-Fuzzy Continuous Speech Recognition Approach for Identifying Laryngeal Pathologies Using New Weighted Spectrum Features. In Proceedings of the International Conference on Advanced Intelligent Systems and Informatics 2017 Advances in Intelligent Systems and Computing, Cairo, Egypt, 9–11 September 2017; 2017; pp. 384–395.
45. Das, A.; Mandal, D.; Ghoshal, S.; Kar, R. Concentric circular antenna array synthesis for side lobe suppression using moth flame optimization. *AEU-Int. J. Electron. Commun.* **2018**, *86*, 177–184. [CrossRef]
46. Pourmousa, N.; Ebrahimi, S.M.; Malekzadeh, M.; Alizadeh, M. Parameter estimation of photovoltaic cells using improved Lozi map based chaotic optimization Algorithm. *Sol. Energy* **2019**, *180*, 180–191. [CrossRef]
47. Prabhu, Y.; Kag, A.; Harsola, S.; Agrawal, R.; Varma, M. Parabel: Partitioned label trees for extreme classification with application to dynamic search advertising. In Proceedings of the 2018 World Wide Web Conference on World Wide Web-WWW, Lyon, France, 23–27 April 2018.
48. Khandagale, S.; Xiao, H.; Babbar, R. Bonsai-Diverse and Shallow Trees for Extreme Multi-label Classification. Available online: https://arxiv.org/abs/1904.082492109 (accessed on 6 August 2019).
49. Babbar, R.; Partalas, I.; Gaussier, E.; Amini, M.-R. On Flat versus Hierarchical Classification in Large-Scale Taxonomies. In Proceedings of the 27th Annual Conference on Neural Information Processing Systems (NIPS 26), Lake Tao, NV, USA, 5–10 December 2013; pp. 1824–1832.
50. Babbar, R.; Partalas, I.; Gaussier, E.; Amini, M.-R.; Amblard, C. Learning taxonomy adaptation in large-scale classification. *J. Mach. Learn. Res.* **2016**, *17*, 1–37.
51. Moradi, M.; Ghadiri, N. Different approaches for identifying important concepts in probabilistic biomedical text summarization. *Artif. Intell. Med.* **2018**, *84*, 101–116. [CrossRef]
52. Mosa, M.A.; Anwar, A.S.; Hamouda, A. A survey of multiple types of text summarization with their satellite contents based on swarm intelligence optimization algorithms. *Knowl.-Based Syst.* **2019**, *163*, 518–532. [CrossRef]
53. Ababneh, J.; Almomani, O.; Hadi, W.; El-Omari, N.K.T.; Al-Ibrahim, A. Vector Space Models to Classify Arabic Text. *Int. J. Comput. Trends Technol.* **2014**, *7*, 219–223. [CrossRef]
54. Fodil, L.; Sayoud, H.; Ouamour, S. Theme classification of Arabic text: A statistical approach. In Proceedings of the Terminology and Knowledge Engineering, Berlin, Germany, 19–21 June 2014; pp. 77–86. Available online: https://hal.archives-ouvertes.fr/hal-01005873/document (accessed on 25 May 2019).
55. Al-Tahrawi, M.M.; Al-Khatib, S.N. Arabic text classification using Polynomial Networks. *J. King Saud Univ. -Comput. Inf. Sci.* **2015**, *27*, 437–449. [CrossRef]
56. Al-Anzi, F.S.; Abuzeina, D. Toward an enhanced Arabic text classification using cosine similarity and Latent Semantic Indexing. *J. King Saud Univ.-Comput. Inf. Sci.* **2017**, *29*, 189–195. [CrossRef]
57. Abuzeina, D.; Al-Anzi, F.S. Employing fisher discriminant analysis for Arabic text classification. *Comput. Electr. Eng.* **2018**, *66*, 474–486. [CrossRef]
58. Al-Anzi, F.S.; Abuzeina, D. Beyond vector space model for hierarchical Arabic text classification: A Markov chain approach. *Inf. Process. Manag.* **2018**, *54*, 105–115. [CrossRef]
59. Alkhatib, M.; Barachi, M.E.; Shaalan, K. An Arabic social media based framework for incidents and events monitoring in smart cities. *J. Clean. Prod.* **2019**, *220*, 771–785. [CrossRef]
60. Ben Hamadou, A.; Piton, O.; Fehri, H. Multilingual extraction of functional relations between Arabic named entities using Nooj platform. In Proceedings of the Nooj 2010 International Conference and Workshop, Komotini, Greece, 27–28 May 2010; Gavriilidou, Z., Chadjipapa, E., Papadopoulou, L., Silberztein, M., Eds.; 2010; pp. 192–202.

61. Boujelben, I.; Jamoussi, S.; Ben Hamadou, A. Rules based approach for semantic relations extraction between Arabic named entities. In Proceedings of the International NooJ 2012 Conference, Paris, France, 14–16 June 2012; pp. 123–133.
62. Mesmia, F.B.; Zid, F.; Haddar, K.; Maurel, D. ASRextractor: A Tool extracting Semantic Relations between Arabic Named Entities. *Procedia Comput. Sci.* **2017**, *117*, 55–62. [CrossRef]
63. Celli, F. Searching for Semantic Relations between Named Entities in I-CAB 2009. Available online: http://clic.cimec.unitn.it/fabio (accessed on 28 April 2019).
64. Shahine, M.; Sakre, M. Hybrid Feature Selection Approach for Arabic Named Entity Recognition. In *Computational Linguistics and Intelligent Text Processing Lecture Notes in Computer Science*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 452–464.
65. Kadir, R.A.; Bokharaeian, B. Overview of Biomedical Relations Extraction using Hybrid Rule-based Approaches. *J. Ind. Intell. Inf.* **2013**, *1*, 169–173. [CrossRef]
66. Landauer, T.K.; Dumais, S.T. A solution to Platos problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychol. Rev.* **1997**, *104*, 211–240. [CrossRef]
67. Alkhatib, M.; Monem, A.A.; Shaalan, K. A Rich Arabic WordNet Resource for Al-Hadith Al-Shareef. *Procedia Comput. Sci.* **2017**, *117*, 101–110. [CrossRef]
68. Salton, G.; Buckley, C. Term-weighting approaches in automatic text retrieval. *Inf. Process. Manag.* **1988**, *24*, 513–523. [CrossRef]
69. Wei, Y.-Y.; Wang, R.-J.; Hu, Y.-M.; Wang, X. From Web Resources to Agricultural Ontology: A Method for Semi-Automatic Construction. *J. Integr. Agric.* **2012**, *11*, 775–783. [CrossRef]
70. Zhang, X.; Chan, F.T.; Yang, H.; Deng, Y. An adaptive amoeba algorithm for shortest path tree computation in dynamic graphs. *Inf. Sci.* **2017**, *405*, 123–140. [CrossRef]
71. Shojaedini, E.; Majd, M.; Safabakhsh, R. Novel adaptive genetic algorithm sample consensus. *Appl. Soft Comput.* **2019**, *77*, 635–642. [CrossRef]
72. AbdelRahman, S.; Elarnaoty, M.; Magdy, M.; Fahmy, A. Integrated Machine Learning Techniques for Arabic Named Entity Recognition. *IJCSI Int. J. Comput. Sci. Issues* **2010**, *7*, 27–36.
73. Oudah, M.; Shaalan, K. Person Name Recognition Using the Hybrid Approach. In *Natural Language Processing and Information Systems Lecture Notes in Computer Science*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 237–248.
74. Benajiba, Y.; Rosso, P.; Benedíruiz, J.M. ANERsys: An Arabic Named Entity Recognition System Based on Maximum Entropy. In *Computational Linguistics and Intelligent Text Processing Lecture Notes in Computer Science*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 143–153.
75. Abdul-Hamid, A.; Darwish, K. Simplified Feature Set for Arabic Named Entity Recognition. Available online: https://www.aclweb.org/anthology/W10-2417 (accessed on 24 April 2019).
76. Nédellec, C.; Nazarenko, A. Ontologies and Information Extraction. 2005. Available online: https://hal.archives-ouvertes.fr/hal-00098068/document (accessed on 18 April 2019).
77. Ding, J.; Berleant, D.; Nettleton, D.; Wurtele, E. Mining Medline: Abstracts, Sentences, Or Phrases? *Biocomputing 2002* **2001**, *7*, 326–337. [CrossRef]
78. Gridach, M. Deep Learning Approach for Arabic Named Entity Recognition. In *Computational Linguistics and Intelligent Text Processing Lecture Notes in Computer Science*; Springer: Cham, Switzerland, 2018; pp. 439–451.
79. Khalifa, M.; Shaalan, K. Character convolutions for Arabic Named Entity Recognition with Long Short-Term Memory Networks. *Comput. Speech Lang.* **2019**, *58*, 335–346. [CrossRef]
80. Taghizadeh, N.; Faili, H.; Maleki, J. Cross-Language Learning for Arabic Relation Extraction. *Procedia Comput. Sci.* **2018**, *142*, 190–197. [CrossRef]
81. Oudah, M.; Shaalan, K. Studying the impact of language-independent and language-specific features on hybrid Arabic Person name recognition. *Lang. Resour. Eval.* **2016**, *51*, 351–378. [CrossRef]
82. Shaalan, K.; Oudah, M. A hybrid approach to Arabic named entity recognition. *J. Inf. Sci.* **2013**, *40*, 67–87. [CrossRef]
83. Lima, R.; Espinasse, B.; Freitas, F. A logic-based relational learning approach to relation extraction: The OntoILPER system. *Eng. Appl. Artif. Intell.* **2019**, *78*, 142–157. [CrossRef]
84. Ma, C.; Zhang, Y.; Zhang, M. Tree Kernel-based Protein-Protein Interaction Extraction Considering both Modal Verb Phrases and Appositive Dependency Features. In Proceedings of the 24th International Conference on World Wide Web-WWW 15 Companion, Florence, Italy, 18–22 May 2015.

85. Li, L.; Guo, R.; Jiang, Z.; Huang, D. An approach to improve kernel-based Protein–Protein Interaction extraction by learning from large-scale network data. *Methods* **2015**, *83*, 44–50. [CrossRef] [PubMed]
86. Qian, L.; Zhou, G. Tree kernel-based protein–protein interaction extraction from biomedical literature. *J. Biomed. Inform.* **2012**, *45*, 535–543. [CrossRef] [PubMed]