Journal of King Saud University – Computer and Information Sciences (2014) **26**, 462–472

King Saud University

Journal of King Saud University – Computer and Information Sciences

www.ksu.edu.sa
www.sciencedirect.com

Automatic extraction of ontological relations from Arabic text

Mohammed G.H. Al Zamil [*], **Qasem Al-Radaideh**

Department of Computer Information Systems, Yarmouk University, Irbed, Jordan

Available online 28 September 2014

KEYWORDS

Arabic ontology;
Lexical syntactic patterns;
Automatic extraction of relationships

Abstract Automatic extraction of semantic relationships among Arabic concepts to formulate ontology models is crucial for providing rich semantic metadata. Due to the annual increase of Arabic content on the Internet, the need for specialized tools to analyze and understand Arabic text has emerged. This research proposes a methodology that extracts ontological relationships. The goals of this research are: to extract semantic features of Arabic text, propose syntactic patterns of relationships among concepts, and propose a formal model of extracting ontological relations.

The proposed methodology has been designed to analyze Arabic text using lexical semantic patterns of the Arabic language according to a set of features. Next, the features have been abstracted and enriched with formal descriptions for the purpose of generalizing the resulted rules. The rules, then, have formulated a classifier that accepts Arabic text, analyzes it, and then displays related concepts labeled with its designated relationship. Moreover, to resolve the ambiguity of homonyms, a set of machine translation, text mining, and part of speech tagging algorithms have been reused. We performed extensive experiments to measure the effectiveness of our proposed tools. The results indicate that our proposed methodology is promising for automating the process of extracting ontological relations.

1. Introduction

The term Ontology has been defined by Gruber (1993) as “a specification of conceptualization”, a formal modeling of a linguistic component along its semantic relationships with respect to other concepts. Ontology can be seen as a pattern of how a given concept is designed to be correlated with other existing ones in a given context.

Developing ontologies from Arabic text is a complex process, as the extraction of the semantic relationships among linguistic components still depends on the syntactic structure of the language. However, interpreting domain independent text requires determining what type of information will be processed and the way it will be expressed. Rather than explaining everything in the text (i.e., syntactic analysis), one could only search for well-known lexical relationships. Thus, meaningful information could be found with simple and soft algorithms, which leads to soft automation of the process.

[*] Corresponding author.
E-mail addresses: Mohammedz@yu.edu.jo (M.G.H. Al Zamil), qasemr@yu.edu.jo (Q. Al-Radaideh).

Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

http://dx.doi.org/10.1016/j.jksuci.2014.06.007

Table 1 Examples of Hearst-style hyponyms.

Text	Lexical Relation (Hyponym–Hypernym)	Relations
"*Input–output devices, such as Printers*"	Hyponym (Input-Devices, Printer)	NP such as NP
"*Temples are civic buildings*"	Is-a (temple, civic building)	NP is a Adj-Phrase
"*Most European countries, especially France, England, and Spain*"	Kind-of (France, European Country)	NP especially NP
	Kind-of (France, European Country)	
	Kind-of (France, European Country)	

Consider the Hearst-style examples in Table 1 in which a set of useful relations have been extracted in a simple manner. Hearst, (1992) has proposed an acquisition algorithm to detect hyponyms automatically by constructing lexical patterns of knowledge. For instance, consider the example "*input–output devices, such as Printers*" in which a human can easily conclude that printers are a type of input–output device. To make it machine-interpretable, Hearst has proposed the following lexical pattern that can be reused to extract such a relation:

$$NP \text{ such as } \{NP,\}^{*}\{(\text{or}|\text{and})\}NP \tag{1}$$

The disadvantage of Hearst's algorithm is the requirement of collecting real examples as a training set, which is considered a pure supervised activity that requires human intervention. In fact, many researchers found it to be a good feature in that it allows for application of the algorithm to independent text, accents, and special languages by feeding the algorithm with a training set of manual examples. Research on sentimental text analysis in social networks benefits from application of this algorithm, which could be an interesting future direction for Arabic text understanding. Furthermore, specialists in domains such as crime and terrorism detection found this methodology interesting for investigating suspicious text in social networks to detect specific conversations (Ressler, 2006; Salton et al., 1990).

To the best of our knowledge, automatic extraction of semantic relationships from Arabic text based on soft-computing principles has not received significant concern compared with English text. In fact, previous research has focused on the syntactic analysis of Arabic statements and dictionary-based analysis to understand Arabic text for different applications, such as summarization, retrieval, and stemming. Therefore, developing soft and intelligent algorithms for extracting lexical semantic relationships dedicated to Arabic text is of great interest.

In this paper, we consider the application of an enhanced version of Hearst's Algorithm to an Arabic corpus. The proposed methodology has been designed to adapt Hearst's algorithm with additional enhancements to fit our needs, analyzing Arabic text to extract ontological relationships. Such enhancements include: pattern enrichment, pattern filtering, the application of negative patterns, and pattern evaluation. However, experiments have been designed for different types of Arabic text. The results indicated that our proposed methodology is a good candidate to formulate Arabic ontological relations.

This paper is organized as follows: Section 2 discusses the related work in addition to background information about Hearst's algorithm. Section 3 illustrates the framework for applying Hearst's algorithm. Section 4 describes the experiments performed to evaluate the proposed methodology and reports the results. Section 5 discusses and justifies the analysis results. Finally, Section 6 summarizes the conclusions and discusses some future directions.

2. Related work

The most recent decade has witnessed an increasing concern for building Arabic Ontology. Efforts have focused on adapting Arabic ontologies in different natural language processing tasks, such as information retrieval (Moawad et al., 2010), text summarization (Imam et al., 2013), text annotation (Hazman et al., 2012; Dukes and Habash, 2010), improving question answering systems (Abouenour et al., 2008), and building semantic mining of knowledge (Beseiso et al., 2010). The expressive power of the Arabic language makes it difficult to extract ontological relations automatically. Therefore, efficient automatic elicitation of such relations is a complex task that is still reliant on dictionaries (Jarrar, 2013) and cross-language translation, such as Arabic WordNet (Ruiz-Casado et al., 2007; Black et al., 2006; Diab, 2004; Elkateb et al., 2006).

Automatic extraction of ontological relations among language concepts has attracted many researchers. For instance, the ARTEQUAKT project (Alani et al., 2003) has constructed a tool to extract relations to create a biography of a given artist using lexical analysis. Furthermore, many promising techniques have been proposed to handle the problem of creating, managing, and populating Arabic ontology. Al-Yahya et al. (2011) introduced an efficient linguistic approach that restricts its application on fully structured text, such as the Holy Qur'an. Similarly, Al-Rajebah and Al-Khalifa (2013) have incorporated a semantic field in which the meaning of a concept is given according to the concepts around it.

Ghneim et al. (2009) proposed a multilingual framework for Arabic Ontology learning based on previous domain knowledge. A Probabilistic Ontology Model (POM) is applied to represent the extracted ontology. Similar to our proposed technique, the framework learns new concepts and relations using Lexico-syntactical patterns. Another interesting technique is the one proposed by Al-Safadi et al. (2011), which is based on structuring Arabic text into classes, properties, and relationships. The experiments only showed how the developed ontology can be used for querying blogs using Arabic terms. The authors did not provide any experiment regarding the effect of using the proposed ontology for retrieval. While these interesting techniques have introduced ontological relation extraction, we argue that additional enhancements could improve such task.

Practically, there are three methods that have been proposed to automatically extract ontological relations (Wandmacher et al., 2007): repeated-segment, Co-occurrence techniques, and lexical patterns.

The repeated-Segment technique has been applied in Wandmacher et al. (2007), Hernandez (2005). The authors have hypothesized that the repetition of some concepts or phrases gives an indicator that such segments are related to a specific domain of text. By indexing segments with their actual positions in the text, the algorithm will be able to identify a quadratic window of repeated segments. Further, a filtering task is applied for the purpose of removing incorrect segments. Alternately, the co-occurrence technique (Koo et al., 2003) relies on hypothesizing that two concepts are related to each other if they both occur frequently in a domain text. It measures how concepts are attracted to other concepts in a specific domain corpus, i.e., statistical analysis.

Hearst (1992) has suggested the application of lexical syntactic patterns to provide a fine grain solution for extracting knowledge automatically. The main drawback of this technique is the inefficient performance on ambiguous text, such as Arabic text. Ambiguity is one factor that complicates the process of automatically extracting relations from Arabic text (Abouenour et al., 2013; Ratnaparkhi, 1998). Lahbib, 2013 applied vocalized text to reduce ambiguity by exploiting syntactic dependencies to infer semantic relations. Al Zamil and Can (2011) also introduced an interesting technique that is based on extending the theory of Inductive Logic Programming to include both positive and negative patterns.

In this paper, we introduce an improved version of Heart's technique to cope with our requirement to extract ontological relations automatically. Our improvements include pattern enrichment with contextually related concepts, pattern filtering to remove redundant low performance rules, the application of negative patterns to reduce ambiguity, and a pattern evaluation function for the validation phase.

3. Methodology

To apply Hearst's algorithm for extracting ontological relationships from an Arabic corpus, additional enhancements are necessary to adapt. In this section, we highlight the proposed enhanced technique that relies on modifying Hearst's algorithm and integrating it into a framework. Fig. 1 shows the framework, which consists of five functional components.

Formally, to resolve the ambiguity of having homonyms or concepts refer to different contexts, we partition the semantic patterns into two parts (Al Zamil and Can, 2011), positive and negative. Thus, the positive part represents the existence of the correct relation, while the negative one represents the irrelevant concept once it is suspected to create ambiguity.

The semantic relation SR consists of positive (p_r^+) and negative (p_r^-) rules, in which the classifier is seeking instances that hold the positive condition but not the negative one. Furthermore, a rule is a language of variants and constant relations (Lex_{Rel}), where variants are represented as part-of-speech components.

$$SR \leftarrow p_r^+ \wedge \nexists p_r^- \tag{2}$$

$$P_r^+ \leftarrow \{p_{r1}^+, p_{r2}^+, \cdots, p_{rn}^+\} \tag{3}$$

$$p_{ri}^+ \leftarrow \{*\}Lex_{Rel}\{*\} \tag{4}$$

$$p_r^- \leftarrow \{*\}Lex_{Rel}\{*\}|p_r^- \notin P_r^+ \tag{5}$$

3.1. Preprocessing and feature extraction

Preprocessing and feature extraction tasks play a significant role in facilitating the manipulation of the text during analysis. However, before illustrating our methodology, we found it mandatory to describe the input text in its new form. Furthermore, specifying and justifying the features to be used in detecting textual patterns are necessary and affect the overall accuracy of the proposed methodology. Features are defined, in this context, as the language components and the relationships among them. While it is proposed that language components are to be extracted automatically, the relationships, in contrast, will be specified manually to limit the scope of the experiments.

As shown in Table 2, we extract 4 different features, as we believe that these features satisfy the requirements of building lexical syntactic patterns of Arabic text. Table 3 shows the relationships to be examined in this research and descriptions of their structure as a set of examples. The Hyponym–Hypernym relationship states that there is a semantic connection between two concepts: Hyponym and Hypernym. The Cause-Effect relation models the causality relationship, in which the reason and the result of some action are modeled. In addition, Part-Whole, Is-a, Has-a, and kind-of model the hierarchical relationships among concepts.

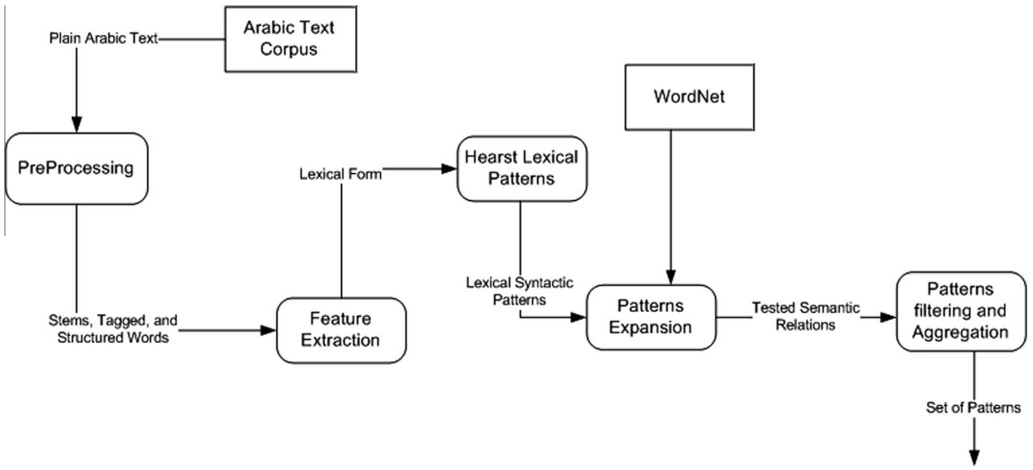


Figure 1 Framework of extracting the semantic relations.

Table 2 Arabic text features.

Feature	Description	Example
Word	Used to deal with the original text	- تنظيم المؤتمر السابع في العاصمة العمانية مسقط
POS Tag Feature	Used to identify the type of each linguistic component, such as: Nouns, Adjectives, Verbs, Imperative Verbal Nouns, Prepositions …	- تنظيم (IMPN) المؤتمر(N) السابع(NUM) في(P) العاصمة(N) العمانية(ADJ) مسقط(N)
Stem	Used to denote the roots of words. Stems are valuable to generalize solutions.	- نظم، أمر، سبع، عصم، عمن، سقط
Phrase	Combination of words that, in totality, gives a specific meaning.	- المملكة العربية السعودية - جامعة اليرموك - صحيفة الوطن

Table 3 Lexical relations.

Relation Type	Description	Example	Example Relation
Hyponym-Hypernym	$Hyponym\ (S_1,S_2) \rightarrow S_i = \{c_1, c_2, \ldots, c_n\}\ and\ \forall(c_i \subseteq S_1 \wedge c_j \subseteq S_2 \rightarrow Hypernym) \rightarrow\ Hyponym(c_i, c_j)$	***Hyponym***(“اربد”,”الاردن”) ***Hyponym***(“حرب أهلية”, “سورية”) ***Hyponym***(“فرعون”, “موسى”)	NP + NP + "عاصمة" NP + "هي عاصمة" + NP NP + "في" + NP + "مدينة" NP + "مقبلة على" + NP NP + "يا" + NP + "قال"
Cause-Effect	$Cause\ (S_1,S_2) \rightarrow S_i = \{c_1, c_2, \ldots, c_n\}\ and\ \forall(c_i \subseteq S1 \wedge c_j \subseteq S_2 \rightarrow Effect) \rightarrow\ Cause(c_i, c_j)$	***Cause***(“برد”, “انفلونزا”) ***Cause***(“سرعة”, “حادث”)	NP + "يسبب" + NP NP + "بسبب" + NP NP + "نتيجة" + NP
Is-a	$Isa\ (S_1,S_2) \rightarrow\ S_i = \{c_1, c_2, \ldots, c_n\}\ and\ \forall(c_i \subseteq C \wedge c_j \subseteq C)$	***Isa***(“الملك عبدالله”, “خادم الحرمين”) ***Isa***(“نهر”, “زجر”)	NP + "هو" + NP NP ***isa*** Syn(NP)
Part – whole	$PartOf\ (S_1,S_2) \rightarrow S_i = \{c_1, c_2, \ldots, c_n\}\ and\ S_2 \subseteq S_1\ and\ \forall(c_i \subseteq S_1 \wedge c_j \subseteq S_2\ \rightarrow Part) \rightarrow\ PartOf(c_i, c_j)$	***PartOf***(“مجلس التعاون”, “السعودية”) ***PartOf***(“مركبة”, “إطار”)	NP+ "عضو في" + NP NP + "من مكونات" + NP
Has-a	$Has-a\ (S_1,S_2) \rightarrow S_i = \{c_1, c_2, \ldots, c_n\}\ and\ S_2 \subseteq S_1\ and\ \forall(c_i \subseteq S_{\ 2} \rightarrow c_i \subseteq S_1\ )$	***Has-a***(“مكة”, “الكعبة”) ***Has-a***(“كارلوس”, “سجل اجرامي”)	NP + "تقع في" + NP NP + "له" + N
Kind-of	$Kind-of\ (c_1,c_2) \rightarrow c_{\ 1} \in S_1\ and\ c_2 \in\ S_1\ )$	***KindOf***(“اسد”, “حيوان”) ***KindOf***(“عنب”, “فاكهة”)	NP + "نوع من" + NP NP + "احد انواع" + NP

3.2. Lexical syntactic patterns

In this section, we describe the enhanced version of Hearst's algorithm on Arabic text. Fig. 2 illustrates the tasks required to apply the algorithm.

The first step is a manual one; linguistic experts in a specific domain of knowledge have to decide on the suitable relations. Then, examples can be extracted automatically to form the training set. Having the relations and a set of training examples, an algorithm runs to find similar occurrences in the text. Once the existing occurrences are available, the algorithm hypothesizes them into a common syntax that generalizes the pattern using part-of-speech tags. However, all extracted patterns are subjected to an evaluation task that tests the coverage of every pattern to remove dirty ones, which cover extremely few cases (according to a given threshold) or even none.

Formally, given a non-empty set $P = \{t_{i1}, t_{i2}\}$ of paired examples (patterns) in which there is a predefined relation $R \dashv \{t_{i1}, t_{i2}\}$, where t_i is a feature expression (such as *word or phrase*), the relation R is a mapping function between two different features, and the condition $\forall(i)|\nexists[P(t_{i1}, t_{i2}) \wedge (t_{i1} = t_{i2})]$ holds. The algorithm has to find similar paired examples in

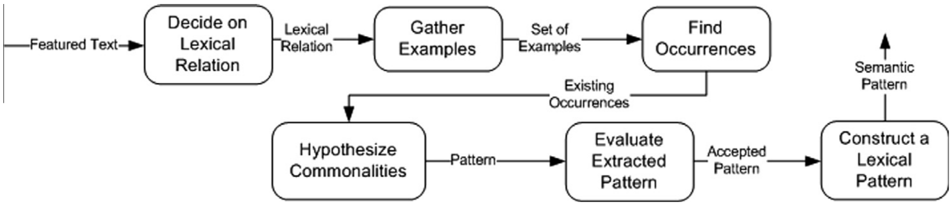


Figure 2 Hearst's algorithm for extracting semantic relations.

the text. However, existing occurrences are, then, subjected to a ranking (based on *inDegree* measure) task by assigning weights to each of them, such as:

$$\text{Weight(c)} = \frac{\max[(\log(f_c) - \text{Log}(f_{\min}), 0]}{\log(f_{\max}) - \text{Log}(f_{\max})} \tag{6}$$

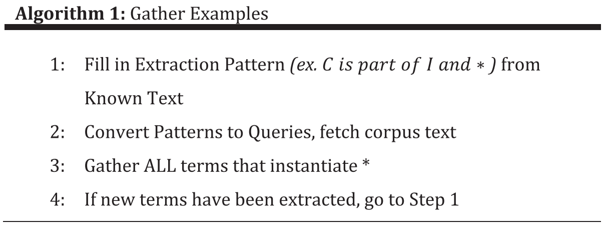
where f is the frequency of the tested pattern.

This implies that identical expressions are weighted more. Finally, the algorithm ends by formulating the patterns into a machine-readable syntax that, ultimately, represents the final form of the semantic pattern that received the highest ranking according to some given threshold. The evaluation and the weighting steps are new enhancements of the original algorithm to improve its accuracy.

To facilitate implementing the algorithm on Arabic text that might hold a significant amount of complicated relations (Arabic Hyponym–Hypernym), Hearst's algorithm required some modifications to supervise part of it, leading to a semi-supervised methodology. Such requirement has been satisfied by converting every pattern into a query that is fetched for the purpose of finding matching text.

The loop, shown in Fig. 3, continues until no new terms have been collected for the same pattern. However, at this stage, our algorithm enriches the resulted patterns with stems, which guarantee discovering more accurate occurrences. In fact, such expansion of terms during stemming, tagging, and phrasing enrichment might create ambiguity and, thus, decrease the overall accuracy. To address this, redundancy must be eliminated later in the filtering task. Indeed, the filtering task is an additional enhancement to the original algorithm.

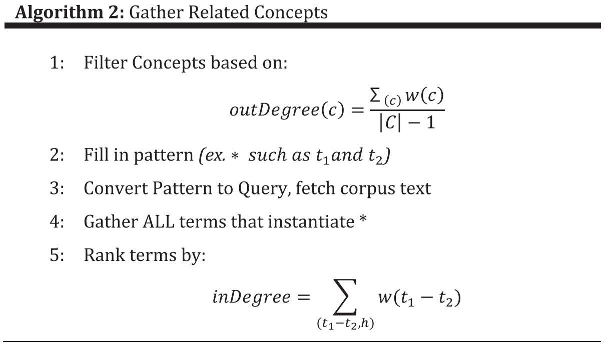
Algorithm 2, shown in Fig. 4, has been designed to fill the second part of related concepts. The *outDegree* score for a given concept (c) represents the weighted sum of outgoing relations (edges in WordNet) that have been normalized by the total number of other concepts in the corpus. The *inDegree* score measures the popularity of a given concept, the number of concepts that refer to the given one. Thus, *inDegree* is used to rank the resulted concepts.



Algorithm 1: Gather Examples

```
1:  Fill in Extraction Pattern (ex. C is part of I and *) from
    Known Text
2:  Convert Patterns to Queries, fetch corpus text
3:  Gather ALL terms that instantiate *
4:  If new terms have been extracted, go to Step 1
```

Figure 3 Gathering examples.



Algorithm 2: Gather Related Concepts

```
1:  Filter Concepts based on:
        $outDegree(c) = \frac{\sum_{(c)} w(c)}{|C| - 1}$
2:  Fill in pattern (ex. * such as $t_1$ and $t_2$)
3:  Convert Pattern to Query, fetch corpus text
4:  Gather ALL terms that instantiate *
5:  Rank terms by:
        $inDegree = \sum_{(t_1 - t_2, h)} w(t_1 - t_2)$
```

Figure 4 Gathering related concepts.

The example in Fig. 5 shows some extracted relations among concepts. It is important to note that some concepts might have more than one relation with another concept; for instance, the capital city of KSA is Riyadh, but, alternately, Riyadh is part of KSA as well. If the input text does not contain enough information to handle a relationship, the tool will fail to detect it. Therefore, comprehensiveness of the training set affects the overall performance of our methodology.

In cases of ambiguity and misunderstanding that result from the machine readability of text, our proposed framework applies negative patterns. Negative patterns are significant in maximizing accuracy while minimizing misclassified instances. For example, consider the following example:

1. Hyponym (“عمان”, “الاردن”)
2. Hyponym (“مسقط”, “عمان”)

It is clear that, without punctuation, the name of the capital city of Jordan is the same as the name of the state Oman. To overcome this problem, negative patterns are attached to detect such phenomena. However, after accumulating the extracted patterns from the training set, the accuracy of patterns is tested. If the negation of a given pattern P_j enhances the accuracy of the pattern P_i, such as $accuracy\ (P_i) < accuracy\ (P_i) < accuracy\ (P_i \wedge \{\sim P_j\}), \forall(j) : j \neq i$, then the pattern P_j is considered as a negative pattern of P_i.

Finally, given a set of patterns $P = \{p_1, p_2, \cdots, p_n\}$ in which $p_i = \{p_i^+ \wedge p_i^-\}$, there are a set of concepts $C = \{c_1, c_2, \cdots, c_m\}$ and a set of relations $R = \{R_1, R_2, \cdots, R_k\}$. The lexical syntactic pattern is formulated as follows:

$$P_i = R(C_i, C_j) \tag{7}$$

3.3. Expansion phase

To avoid having redundant patterns that refer to the same concepts, the evaluation phase expands the lexical structure

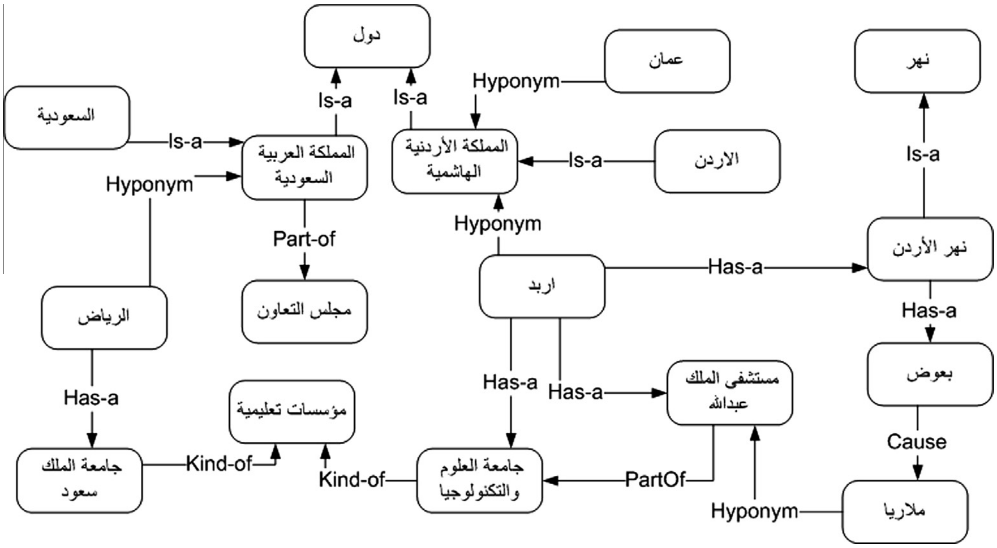


Figure 5 Example of extracted concepts from newspaper.

of the patterns to include synonyms. Indeed, we believe that this phase will optimize the total number of extracted patterns. Furthermore, such expansion might result in extracting new relationships among concepts that did not appear clearly in the text. To satisfy this goal, we created a program segment that calls the synonym routine in the Arabic WorldNet tool.

For example, our tool was able to extract the following pattern from the holy Qur'an: Hyponym (“انسان”, “الله”), which means that there is a semantic relationship among both concepts. After expanding this relation by enriching it with related concepts, the relation becomes: Hyponym (“انسان”, “الله”), Hyponym (“انسان”, “اله”), Hyponym (“انسان”, “رب”), Hyponym (“انسان”,“خالق”).

Although such expansion might contribute in discovering new relations, some existing patterns became redundant, i.e., they produce the same effect on the dataset. To avoid redundancy, another filtering approach needs to be applied to detect redundant patterns and remove them.

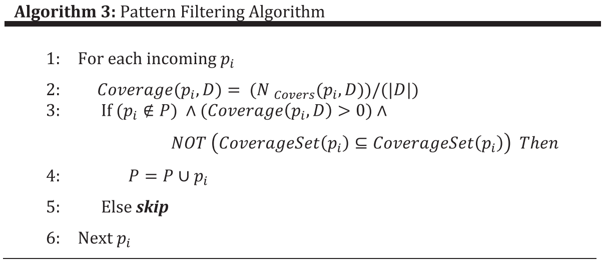
3.4. Pattern filtering and aggregation

Because the proposed method is semi-supervised and because of the effect of the expansion task on the resulted set, some output patterns might cover the same relationship. For example:

1. Hyponym (“الملك عبدالله”, “السعودية”), Hyponym (“الملك عبدالله”, “المملكة العربية السعودية”).
2. Hyponym (“نفط”, “بترول”), Hyponym (“بترول”, “أرامكو”).

To overcome such problems (i.e., transitive relations, synonyms, and concept representation), we applied the coverage metric that determines the coverage of a pattern in a specific dataset. If one pattern covers the same data instances of another pattern, the second one will be removed. The coverage of a given pattern P in a dataset D is defined as follows:

$$Coverage\ (P_i, D) = \frac{N_{Covers}\ (P_i, D)}{|D|} \tag{8}$$



Algorithm 3: Pattern Filtering Algorithm

```
For each incoming $p_i$
    $Coverage(p_i, D) = (N_{Covers}(p_i, D))/(|D|)$
    If $(p_i \notin P) \wedge (Coverage(p_i, D) > 0) \wedge$
            $NOT\ (CoverageSet(p_i) \subseteq CoverageSet(p_i))\ Then$
        $P = P \cup p_i$
    Else ***skip***
Next $p_i$
```

Figure 6 Pattern filtering.

$$Coverage\ Set\ (S, P_i, D) = \{c_1, c_2, \cdots, c_k\}|\ \text{such as}\ k \equiv Coverage\ (P_i, D) \tag{9}$$

where $N_{Covers}\ (P_i, D)$ is the total number of concepts covered by the pattern P_i, and $|D|$ is the total number of concepts in the dataset. Furthermore, the pattern P_i is considered redundant if at least one of the following conditions holds:

1. $\forall(i)\ \exists\ (j){:}(P_i = P_j) \wedge (i \neq j)$
2. $\forall(i)\ \exists\ (j){:}CoverageSet(i) \subseteq CoverageSet(j)$

Thus, applying these rules using a well-formed validation algorithm will result in minimizing the output patterns, which, in turn, optimizes the overall performance of our proposed framework. The following validation algorithm, shown in Fig. 6, applies the coverage rules for the purpose of filtering redundant patterns.

4. Experiments and results

In this section, we provide detailed descriptions of experiments on different datasets and report the results in terms of precision, recall, and *f*-measure. Moreover, we provide a sensitivity analysis that shows the effect of different phenomena on different performance metrics. Moreover, we provide a comparison with similar techniques on Arabic datasets. Finally, we highlight the main errors that our proposed technique produced during experiments. Notice that results have been justified in the discussion section to reason about robustness and weakness of our proposed work.

4.1. Corpora

To evaluate our proposed framework and provide an extensive analysis and discussion of the reported results, we ran our algorithm on three different Arabic datasets. These datasets represent Classical Arabic (Holy Qur'an), Modern Standard Arabic (newspapers), and unstructured Arabic texts (social blogs). This method allows us to detect biases and justify results. Furthermore, future research on certain types of Arabic datasets might benefit from our analysis for the purpose of comparing results. The holy Qur'an consists of 114 chapters, which have a total of 6236 verses. The newspaper dataset consists of 1000 documents from three different newspapers: the Middle East (صحيفة الشرق الأوسط), AL Watan Saudi Newspaper (الوطن السعودية صحيفة), and AL Rai Newspaper (صحيفة الرأي الأردنية). Finally, 400 statements have been collected from Arabic Facebook blogs. Notice that all datasets have been exposed to the part-of-speech tagging algorithm (http://nlp.stanford.edu/downloads/tagger.shtml) during indexing. Additionally, we ran the stemming algorithm of Khoja and Garside (1999) to obtain the roots of Arabic terms.

4.2. Performance evaluation

The performance evaluation of the proposed methodology takes three directions: measuring the correctness of extracted patterns with respect to existing correct ones using a recall metric, measuring the ability of our proposed methodology to detect patterns with respect to all retrieved information using a precision metric, and, finally, applying an *F*-measure that denotes the overall accuracy.

Given the number of correctly classified concepts, denoted as $|TP|$, the number of incorrectly classified concepts, denoted as $|FP|$, and the number of concepts that were not classified but should have been, denoted as $|FN|$, the precision, recall, and *F*-measure are defined as follows:

$$\text{Precision} = \frac{\sum_{c\in C}|TP|_c}{\sum_{c\in C}|TP|_c + |FP|_c},$$

$$\text{Recall} = \frac{\sum_{c\in C}|TP|_c}{\sum_{c\in C}|TP|_c + |FN|_c},$$

$$F\text{-measure} = \frac{2 \times (\text{Precision} \times \text{Recall})}{\text{Precision} + \text{Recall}}.$$

Table 4 reports the performance of each dataset. Our proposed framework was able to extract 317 lexical patterns from the holy Qur'an dataset, of which 209 of them were correctly classified, 65 of them were incorrectly classified, and 43 of them were misclassified. Similarly, the performance for the Newspapers and Blogs datasets has been reported according to these parameters. Therefore, the overall performance averages in terms of precision, recall, and *f*-measure were 78.57%, 80.71%, and 79.54%, respectively.

The results showed that the performance among different datasets is not systematic. However, the Newspapers dataset experienced the highest performance compared with other datasets. Alternately, the Blogs dataset experienced the lowest performance.

Table 4 Performance Metrics and Results.

Corpus	*N*	*TP*	*FP*	*FN*	Precision (%)	Recall (%)	*F*-measure (%)
Qur'an	317	209	65	43	76.28	82.94	79.47
Newspaper	205	158	18	29	89.77	84.49	87.05
Blogs	110	62	27	21	69.66	74.70	72.09
Average					78.57	80.71	79.54

4.3. inDegree and outDegree evaluation

Measuring the connectivity among concepts that are candidates to formulate a WordNet is crucial as it reflects the robustness of the output ontology. However, *inDegree* and *outDegree* parameters, in this context, model the ingoing and outgoing relationships among concepts (nodes). Fig. 7 depicts the relation between the number of nodes and the degree parameters. For instance, we can notice that both parameters have an inverse relationship with the number of nodes; the higher the number of concepts is, the lower the degree of in and out connections.

Table 5 shows the WordNet construction characteristics of extracted taxonomies. The average depth shows the average number of relational levels, among the lexical relationships, that have been extracted during the experiment. The maximum depth shows the maximum number of levels reached. The minimum depth shows that some concepts appeared with no depth relations. In fact, the algorithm restricted the minimum depth to 1. However, the maximum average depth and the maximum depth are achieved by the holy Qur'an dataset.

4.4. Sensitivity analysis

The sensitivity analysis highlights the parameters that affect our proposed technique. Such analysis is essential as it justifies the results and provides future research areas that could be handled on the basis of this work. Our sensitivity analysis was based on three directions: the type of text, features, and the effect of the filtering task on the reported results.

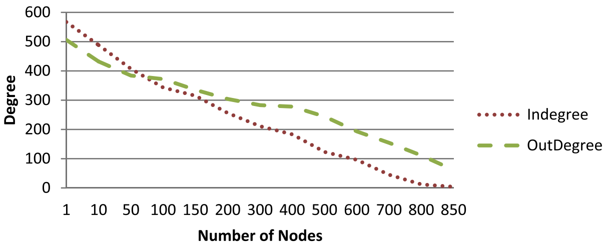


Figure 7 The cumulative distribution of *inDegree* and *outDegree*.

Table 5 WordNet depth analysis.

	Qur'an	Newspaper	Blogs
Average depth	9.21	6.2	3.8
Max depth	18	11	7
Min depth	1	1	1

Table 6 Precision at different intervals.

Dataset	P@20%	P@30%	P@40%	P@50%	P@60%	P@70%	P@80%	P@90%	P@100%
Qur'an	87%	81%	78%	73%	71%	68%	65%	73%	76.28%
Newspaper	91%	92%	89%	87%	86%	88%	90%	91%	89.77%
Blogs	76%	69%	71%	72%	67%	63%	65%	70%	69.66%

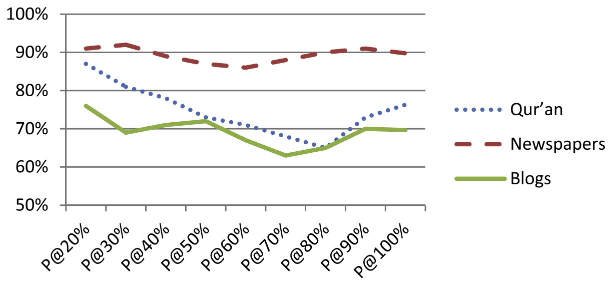


Figure 8 The precision fluctuation among different intervals of the datasets.

4.4.1. *The effect of dataset type on the overall performance*

During experiments, we observed that the Newspapers dataset received the highest performance in terms of recall and precision measures, while the holy Qur'an and Blogs datasets had roughly the same level of accuracy. Therefore, we noticed that the type of dataset (classical, modern, or social) might be a significant factor that affects the performance measures. To prove our hypothesis, we analyzed the precision of each dataset at different intervals in Table 6, where P@N implies the precision at *N*% of the extracted patterns (P@20%, P@30%… P@100%). This way, we can study the performance at different intervals to ensure that the effect of the type of the dataset is real; i.e., not by chance.

To clarify the comparison, Fig. 8 shows a line-chart that depicts data in Table 6. It shows how the Newspapers dataset outperformed the others. Moreover, it shows that the holy Qur'an dataset performed better than Blogs at low precision values, while it fluctuated at higher precisions.

4.4.2. *The effect of features on the overall performance*

Table 7 shows the performance of our technique using different combinations of suggested features. For instance, the results in Table 7 show how adding more features to the elementary feature "word" enhances the overall performance in terms of recall and precision.

Accordingly, it is clear that the effect of using phrases along with the word feature enhances the performance compared with the use of stems. In fact, the phrases feature can accurately detect required information because of its descriptive power compared with stems. Alternately, stems play a significant role in generalizing the textual patterns, which enhance the performance as a complementary feature. Eventually, combining the three features resulted in the best performance, as such combination inherits the advantages of using each one of them.

4.4.3. *The effect of the filtering task on the overall performance*

Filtering out redundancy enhances the precision factor as it minimizes false-positives. In addition, applying negative patterns played a significant role in enhancing the recall measure, as they maximize the correctness factor. Table 8 shows how the filtering task affected the performance on the three datasets.

The reported results, presented in Table 8, showed that the Newspapers dataset achieved the highest enhancement in all comparisons except the number of patterns; the highest enhancement for that goes to Blogs. These results support our conclusion that the filtering task was the reason behind the good performance reported by the Newspapers dataset.

4.5. *Error analysis*

We believe that highlighting frequent classification errors might be an entrance to enhance our proposed algorithm, especially for commercial use. In particular, we found that there are four erroneous cases that appeared frequently. These cases include proper noun extraction, spelling variants, wrong assertions, and broken expressions.

4.5.1. *Type 1: incorrect proper name extraction*

This error resulted from incorrect extraction of some proper names. The problem is that the extraction process is a syntactic

Table 7 Precision during the application of different features.

	Word (%)	Word + Phrase (%)	Word + Stem (%)	Word + Phrase + Stem (%)

Qur'an	Pr = 60.10
	Re = 62.45
	F1 = 61.25
		Pr = 71.46
		Re = 78.21
		F1 = 74.68
			Pr = 65.43
			Re = 66.35
			F1 = 65.89
				Pr = 76.28
				Re = 82.94
				F1 = 79.47

Newspapers	Pr = 65.34
	Re = 64.80
	F1 = 65.07
		Pr = 84.74
		Re = 80.90
		F1 = 82.78
			Pr = 72.45
			Re = 71.92
			F1 = 72.18
				Pr = 89.77
				Re = 84.49
				F1 = 87.05

Blogs	Pr = 48.20
	Re = 57.46
	F1 = 52.42
		Pr = 62.64
		Re = 70.34
		F1 = 66.27
			Pr = 57.63
			Re = 63.72
			F1 = 60.52
				Pr = 69.66
				Re = 74.70
				F1 = 72.09

The bold values are represent the best values among other observations.

Table 8 The Filtering Effect on the Performance Metrics.

	Dataset	Before filtering	After filtering	Enhancement percentage
Number of patterns	Qur'an	418	317	≈24%
	Newspapers	341	205	≈40%
	Blogs	212	110	≈48%
Precision	Qur'an	69.11%	76.28%	≈11%
	Newspapers	71.57%	89.77%	≈25%
	Blogs	62.24%	69.66%	≈12%
Recall	Qur'an	73.75%	82.94%	≈12%
	Newspapers	72.64%	84.49%	≈15%
	Blogs	68.83%	74.70%	≈10%
F-Measure	Qur'an	71.35%	79.47%	≈11.38
	Newspapers	72.10%	87.05%	≈20.74
	Blogs	65.36%	72.09%	≈10.30

one, while removing such errors requires understanding text. For example: "دول مثل فرنسا والدول الناطقة بالعربية مثل لبنان". The hyponym ("دول", "دول ناطقة") has been mistakenly extracted. In the previous example, our lexical rules extracted the phrase "دول ناطقة" as a country name.

4.5.2. *Type 2: spelling variants*

There are many concepts that have different spellings in the Arabic language. One of the most common sets of these concepts is foreign names, i.e., names that have been literally transformed to Arabic. For example, "بكين" and "بيجين" are two correct transformations of the name of the capital city of China, Beijing. Furthermore, the name of the prophet Abraham is used in two different forms, "ابراهيم" and "ابراهم", in the holy Qur'an.

4.5.3. *Type 3: wrong assertions*

In the modern and social text, people communicate phrases that refer to an entity as an entity type. For example, in many cases, we found text that asserts "أمريكا الجنوبية" as a country "دولة". Further, "افريقيا الجنوبية" and "جنوب افريقيا" assert the equality of south-Africa and Southern-Africa as well.

4.5.4. *Type 4: broken expressions*

Broken expressions posed a serious problem, especially in the Newspapers dataset. Writers and publishers save space by using special characters to divide an individual word or phrase into two consecutive parts; for example, "المملكة العربية السعو- دية". Reformulating breaking symbols might result in errors as well, which could minimize the total effect.

4.6. *Comparison with the original Hearst algorithm*

For the purpose of comparing the performance of our proposed techniques with similar ones, we compared it with the original Hearst's algorithm (Hearst, 1992), Repeated-Segment-based extraction (Mazari et al., 2012), and co-occurrence-based extraction (Koo et al., 2003) of ontological relations. As shown in Table 9, the performance measures have been reported in terms of precision, recall, and *F*-measure on different datasets.

Our proposed technique achieved the highest performance compared with existing methods, including the original algorithm. The added enhancements that lead to these results include: expanding, filtering, adapting negative conditions, and validating patterns. Notice that we observed that both the repeated-segment and co-occurrence algorithms performed better on the Blog dataset compared with the holy Qur'an dataset. Our proposed technique, on the other hand, performed well on Newspapers compared with the Qur'an and Blogs datasets.

Table 9 Performance comparison.

Average measurements	Dataset	Precision (%)	Recall (%)	*F*-Measure (%)
Original Hearst's Algorithm (Arabic Text)	Qur'an	46.74	50.32	48.48
	Newspaper	51.23	61.35	55.84
	Blogs	47.43	53.45	50.26
Repeated-Segment (Arabic Text)	Qur'an	64.34	62.66	63.49
	Newspaper	68.87	64.86	66.80
	Blogs	69.67	70.55	70.11
Co-Occurrence (Arabic Text)	Qur'an	57.36	58.67	58.00
	Newspaper	60.46	62.55	61.49
	Blogs	64.56	66.25	65.39
Enhanced Version of Hearst's Algorithm (Arabic Text)	Qur'an	76.28	82.94	79.47
	Newspaper	89.77	84.49	87.05
	Blogs	69.66	74.70	72.09

5. Discussion

The empirical results indicated that our proposed framework is efficient to be implemented for generating ontological relations among Arabic concepts. During the experiments, we found that our tool earned 79.47% on the holy Qur'an dataset, 87.05% on the Newspapers dataset, and 72.09% on the Blogs dataset in terms of *F*-measure. We believe that enhancing the proposed algorithm with the expansion phase, the filtering task, the validation task, and the application of negative patterns plays a significant role in improving the accuracy measures.

Compared with similar techniques, our proposed methodology performed better than the original algorithm of Hearst. In addition, it outperformed the Repeated-Segment and Co-occurrence algorithms in terms of precision and recall. During the experiments, we noticed that the later algorithms rely on statistical analysis while ignoring the semantic aspects of ontological relations.

Moreover, the inverse relationship between the degree and the number of nodes in the In/Out degree analysis showed that our technique was able to generate connections (relations) as the number of incoming nodes (concepts) increases. Such result leads us to conclude that the proposed technique performed well in constructing WordNets.

Moreover, we studied the effects of many factors on the performance of the proposed framework. The results indicated that the type of data directly affected the performance, as classical and sentimental language negatively affected the performance. In contrast, modern standard language (Newspapers) positively affected the performance of our proposed technique. For instance, the lack of ambiguity and special purpose phrases were factors that distinguished the modern language text.

Additionally, the results confirmed that selecting representative features is crucial in improving performance. Indeed, stems and phrases participate in generalizing the meaning of simple words and provide more description. Thus, multiple features appear to be better than a single one. Future research might study this phenomenon and suggest more features to improve the extraction of ontological relations.

Likewise, filtering out redundant patterns and appending the negative conditions within generated rules have a high positive effect on the overall performance of our proposed methodology. The results showed that this task improved the recall, precision, and, thus, *f*-measure with all datasets.

Finally, we reported the frequent classification errors that negatively affect the performance of the proposed technique. Incorrect proper noun extraction, spelling variants, and wrong assertions require understanding the text, while broken expressions require constructing a list of tokens that are used in Arabic to break concepts.

6. Conclusion

In this paper, we presented an automatic technique for extracting ontological relations from Arabic text. The proposed technique relies on implementing an enhanced version of Hearst's algorithm. The goals of this research were (1) to extract semantic features of Arabic text, (2) propose syntactic patterns of relationships among concepts, (3) propose a formal model of extracting semantic ontological relations, and (4) automate the process of extracting semantic relations.

We performed extensive experiments to measure the performance of the proposed method, study the effect of different factors on the performance of the proposed technique, and compare our method with similar ones. The results indicated that our proposed technique is a good candidate for extracting ontological relations from Arabic text compared with existing techniques.

References

Abouenour, L., Bouzoubaa, K., Rosso, P. (2008). Improving Q/A Using Arabic WordNet. In Proc. The 2008 International Arab Conference on Information Technology (ACIT'2008), Tunisia, December.

Abouenour, L., Bouzoubaa, K., Rosso, P., 2013. On the evaluation and improvement of Arabic WordNet coverage and usability. Lang. Resour. Eval., 1–27.

Al Zamil, M.G., Can, A.B., 2011. ROLEX-SP: rules of lexical syntactic patterns for free text categorization. Knowledge Based Syst. 24 (1), 58–65.

Alani, H., Kim, S., Millard, D.E., Weal, M.J., Hall, W., Lewis, P.H., Shadbolt, N.R., 2003. Automatic ontology-based knowledge extraction from web documents. Intel. Syst., IEEE 18 (1), 14–21.

Al-Rajebah, N.I., Al-Khalifa, H.S., 2013. Extracting ontologies from Arabic wikipedia: a linguistic approach. Arab. J. Sci. Eng., 1–23.

Al-Safadi, L., Al-Badrani, M., Al-Junidey, M., 2011. Developing ontology for Arabic blogs retrieval. Int. J. Comput. Appl. 19 (4), 40–45.

Al-Yahya, M., Al-Khalifa, H., Bahanshal, A., Al-Oudah, I., 2011. Automatic generation of semantic features and lexical relations using OWL ontologies. In: Natural Language Processing and Information Systems. Springer, Berlin, Heidelberg, pp. 15–26.

Available at: http://nlp.stanford.edu/downloads/tagger.shtml.

Beseiso, M., Ahmad, A.R., Jais, J., 2010. Semantic Arabic Search Tool. In and Knowledge Engineering Conference (STAKE 2010), p. 40.

Black, W., Elkateb, S., Rodriguez, H., Alkhalifa, M., Vossen, P., Pease, A., Fellbaum, C., 2006. Introducing the Arabic WordNet project. In: Proceedings of the 3rd International WordNet Conference (GWC-06).

Diab, M., 2004. The feasibility of bootstrapping an Arabic WordNet leveraging parallel corpora and an English WordNet. In: Proceedings of the Arabic Language Technologies and Resources, NEMLAR, Cairo.

Dukes, K., Habash, N., 2010. Morphological Annotation of Quranic Arabic. In LREC.

Elkateb, S., Black, W., Rodríguez, H., Alkhalifa, M., Vossen, P., Pease, A., Fellbaum, C., 2006. Building a WordNet for Arabic. In: Proceedings of The fifth international conference on Language Resources and Evaluation (LREC 2006).

Ghneim, N., Safi, W., Said Ali, M., 2009. Building a Framework for Arabic Ontology Learning. In: Proceedings of Knowledge Management and Innovation in Advancing Economics: Analyses & Solutions, pp. 1730–1 1735.

Gruber, T.R., 1993. A translation approach to portable ontology specifications. Knowledge Acquisition 5 (2), 199–220.

Hazman, M., El-Beltagy, S., Rafea, A., 2012. An ontology based approach for automatically annotating documents segments. IJCSI Int. J. Comput. Sci. Issues 9 (2), 221–230.

Hearst, M.A., 1992. Automatic acquisition of hyponyms from large text corpora. In: Proceedings of the 14th conference on Computational linguistics, vol. 2, pp. 539–545. Association for Computational Linguistics.

Hernandez, N., 2005. Ontologies de domaine pour la modélisation du contexte en recherche d'information (Doctoral dissertation, Université Paul Sabatier-Toulouse III).

Imam, I., Nounou, N., Hamouda, A., Khalek Abdul, H., 2013. An Ontology-based Summarization System for Arabic Documents (OSSAD). Int. J. Comput. Appl. 74 (17), 38–43.

Jarrar, M., 2013. The Arabic ontology. In Qatar Foundation Annual Research Conference (No. 2013).

Khoja, S., Garside, R., 1999. Stemming Arabic text. Lancaster, UK, Computing Department, Lancaster University.

Koo, S.O., Lim, S.Y., Lee, S.J., 2003. Building an ontology based on hub words for information retrieval. In Web Intelligence, 2003. WI 2003. Proceedings. IEEE/WIC International Conference on IEEE, pp. 466–469.

Lahbib, W., Bounhas, I., Elayeb, B., Evrard, F., Slimani, Y., 2013. A Hybrid Approach for Arabic Semantic Relation Extraction. In The Twenty-Sixth International FLAIRS Conference.

Mazari, A.C., Aliane, H., Alimazighi, Z., 2012. Automatic Construction of Ontology from Arabic Texts. In ICWIT, pp. 193–202.

Moawad, I.F., Abdeen, M., Aref, M.M., 2010. Ontology-based Architecture for an Arabic Semantic Search Engine. In The Tenth Conference. On Language Engineering Organized by Egyptian Society of Language Engineering (ESOLEC'2010), pp 15–16.

Ratnaparkhi, A., 1998. Maximum entropy models for natural language ambiguity resolution (Doctoral dissertation, University of Pennsylvania).

Ressler, S., 2006. Social network analysis as an approach to combat terrorism: past, present, and future research. Homeland Security Affairs 2 (2), 1–10.

Ruiz-Casado, M., Alfonseca, E., Castells, P., 2007. Automatising the learning of lexical patterns: an application to the enrichment of WordNet by extracting semantic relationships from wikipedia. Data Knowledge Eng. 61 (3), 484–499.

Salton, G., Buckley, C., Smith, M., 1990. On the application of syntactic methodologies in automatic text analysis. Inf. Process. Manage. 26 (1), 73–92.

Wandmacher, T., Ovchinnikova, E., Krumnack, U., Dittmann, H., 2007. Extraction, evaluation and integration of lexical-semantic relations for the automated construction of a lexical ontology. In: Proceedings of the Third Australasian Workshop on Advances in Ontologies, Australian Computer Society Inc, vol. 85, pp. 61–69.