QATAR UNIVERSITY

COLLEGE OF ENGINEERING

ONTOLOGY BASED TEXT-TO-PICTURE MULTIMEDIA M-LEARNING SYSTEM

BY

ABDEL GHANI KARKAR

A Dissertation Submitted to

the Faculty of the College of

Engineering

in Partial Fulfillment

of the Requirements

for the Degree of

Doctorate of Philosophy in Computer Science

January 2018

COMMITTEE PAGE

The members of the Committee approve the Dissertation of Abdel Ghani Karkar defended on 20/12/2017.

Prof. Jihad Mohamad AlJA'AM
Thesis/Dissertation Supervisor

Prof. AbdelAziz Bouras
Committee Member

Prof. Adam Fadlalla
Committee Member

Prof. Nicola Whitton
Committee Member

Approved:

Khalifa Al-Khalifa, Dean, College of Engineering

ABSTRACT

KARKAR, ABDEL GHANI, Doctorate: January: 2018, Doctorate of Philosophy in Computer Science

Title: An Ontology based Text-to-Picture Multimedia m-Learning System

Supervisor of Dissertation: Jihad, Mohamad, AlJA'AM.

Multimedia *Text-to-Picture* is the process of building mental representation from words associated with images. From the research aspect, multimedia instructional message items are illustrations of material using words and pictures that are designed to promote user realization. Illustrations can be presented in a static form such as images, symbols, icons, figures, tables, charts, and maps; or in a dynamic form such as animation, or video clips. Due to the intuitiveness and vividness of visual illustration, many text to picture systems have been proposed in the literature like, *Word2Image*, *Chat with Illustrations*, and many others as discussed in the literature review chapter of this thesis. However, we found that some common limitations exist in these systems, especially for the presented images. In fact, the retrieved materials are not fully suitable for educational purposes. Many of them are not context-based and didn't take into consideration the need of learners (i.e., general purpose images). Manually finding the required pedagogic images to illustrate educational content for learners is inefficient and requires huge efforts, which is a very challenging task. In addition, the available learning systems that mine text based on keywords or sentences selection provide incomplete pedagogic illustrations. This is because words and their semantically related terms are not considered during the process of finding illustrations. In this dissertation, we propose new approaches based on the semantic conceptual graph and semantically distributed weights to mine optimal illustrations that match Arabic text in the children's story domain. We combine these approaches with best keywords and sentences selection algorithms, in order to improve the retrieval of images matching the Arabic text. Our findings show significant improvements in modelling Arabic vocabulary with the most meaningful images and best coverage of the domain in discourse. We also develop a mobile Text-to-Picture System that has two novel features, which are (1) a conceptual graph visualization (CGV) and (2) a visual illustrative assessment. The CGV shows the relationship between terms associated with a picture. It enables the learners to discover the semantic links between Arabic terms and improve their understanding of Arabic vocabulary. The assessment component allows the instructor to automatically follow up the performance of learners. Our experiments demonstrate the efficiency of our multimedia text-to-picture system in enhancing the learners' knowledge and boost their comprehension of Arabic vocabulary.

Keywords: *Multimedia Systems*; *Text-to-Picture; Conceptual Graph*; *Ontology*; *Visualization*; *Learning Technology; Mobile Learning*.

DEDICATION

I dedicate my dissertation and give special thanks to my loving parents and my family. Special words of gratitude to them and to many of my colleagues and friends.

ACKNOWLEDGMENTS

I thank Allah SW to give me the inspiration, motivation, intellect, cleverness to continue my graduate studies and complete successfully this thesis.

I would like to thank my supervisor Professor Jihad Mohamad AlJa'am, who guided me to carry out this research work. His feedback helped me a lot to improve my work and publish my findings in conferences and journals.

I thank Professor Ali Jaoua, Professor Abdelaziz Bouras, and Dr. Tamer ElSayed, who attend several times my presentation and gave me important feedbacks. Many thanks to Dr. Sumaya AlMaadeed, the CSE department head for her encouragement. Special thanks also to Prof. Sebti Foufou and Dr. Arif Mahmood for their advices.

Last but not least, I would like to thank the College of Engineering at Qatar University to select me among the best students for the Graduate Teaching Assistant award during my PhD studies and for providing me the required financial support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF ALGORITHMS

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

Multimedia Text-to-Picture (TTP) applications have a broad prospect in enhancing user realization, education and training [1, 2]. Different research works in TTP support the assertion that textual illustrations can broadly improve the performance of learners in terms of diverse cognitive outcomes [2]. Thus, a user can learn more efficiently from words associated with pictures than with words only. Experimental analysis, that compares realization from text with illustrations [3], strongly demonstrate that depiction with illustration facilitates learning of textual content. However, just adding pictures and symbols to words may not be an efficient approach to build an efficient multimedia TTP system. In fact, a good learning system should be able to mine text and link the words semantically where the leaners can complement her/his knowledge and enrich her/his understanding. We should then provide complete instructional illustrations in the light of mental representation and the context of words [4]. Many multimedia TTP systems have been proposed to automatically illustrate a given topic [5] by depicting inspirational pictures [6], or enriching textual content with illustrations [7]. The automatic depiction requires the identification of substantial concepts in a given text. While several multimedia systems have been proposed [5], [8], [9], there are some common limitations in the existing systems that can be summarized as follows:

(1) The current TTP systems can retrieve pictures automatically from the Internet and generate illustrations [5], [8], [10]. However, the obtained illustrations might not be appropriate for the required learning tasks (i.e., inappropriate

images).

(2) Most of the existing TTP systems provide incomplete pedagogical information. In that case, a manual search should be applied. Many pictures are available on the web, but manually finding the right pedagogic illustrations is very difficult and time consuming [11, 12]. This task cannot be performed during the learning sessions. The instructors should then spend a lot of time to prepare their multimedia-based tutorials.

(3) The TTP systems can illustrate simple text based on sentence or keywords selection [9, 12, 13]. However, these systems do not consider all semantic associations between terms resulting in incorrect or not related illustrations [6, 14]. For instance, searching for the keyword "Mouse" as a computer device is different from searching for the keyword "Mouse" as an "Animal". Thus, the domain of discourse should be considered during the search process.

1.1 Scope Of The Research

The main scope of this dissertation is to propose new approaches to address the challenges of the current TTP systems [5, 8, 11, 15]. These limitations include the lack of accurate retrieval of educational illustrations pertaining to textual content and the lack of consideration of semantic associations between terms in the retrieval process by using conceptual graphs [16]. For the educational illustrations, we consider the available images in online book libraries, such as Scribd, in order to build a multimedia educational repository. With respect to the semantic associations between terms, after careful examination of the techniques used in current TTP systems [5, 8, 11, 15], we found that the employment of semantic knowledge would improve accuracy of results for the retrieval of images that represent the educational input text. In fact, our research work consists of providing an efficient multimedia TTP system that can be accessible through smart devices and stresses on the usage of semantic knowledge base (i.e., ontologies). We first give a detailed study about the current TTP system [17] and the concept of semantic knowledge [18, 19]. Therefore, we show that the employment of semantic conceptual graphs obtained by utilization of the semantic knowledge base can promote the depiction of illustrations for a given textual content. We target the domain of children's educational stories in order to define the scope of the textual content.

1.2 Problem Statement

Based on extensive research of TTP systems that utilize the best sentence selection [5, 8] and best keywords selection weighting techniques [11, 15], we conclude on several common limitations of these systems: 1) they do not have accurate text-to-picture retrieval techniques; 2) they do not consider semantic relations as a part of the text processing phase; 3) they do not consider cognitive applications, and do not retrieve content that can be usable for educational purposes, i.e., images that represent educational text. Our research problem can be formulated as such; the development of a textual processing technique, that takes into consideration semantic associations between concepts in text. This is significant as it addresses the lack of semantic relation utilization in other studies. Semantic associations are to be mapped with the usage of conceptual graphs. The main objective of the proposed technique is to achieve optimized results in terms of retrieval of educationally related images pertaining to a given input text.

1.3 Research Questions

The purpose of our investigation is to explore new approaches to solve some limitations of current TTP systems. We list our research questions as follows:

1) What are the existing Text-to-Picture Systems?
2) What are the TTP systems designed for educational use?
3) How do existing TTP systems work?
4) What are the limitations of best-sentence selection and best-keywords selection techniques in TTP systems?
5) How to overcome the limitation of omitting semantic information?
6) How the employment of conceptual graphs can improve the results of the techniques?
7) What are the conceptual graphs designed for educational use?
8) What is the effect of employing conceptual graphs with existing techniques?
9) What is the effect of integrating conceptual graphs with existing techniques?
10) What are the effects of CG integration on the processing performance?
11) How the distribution of TF-IDF weight (semantic weight) over semantically associated terms affects the results?

12) What is the effect of the semantic weight on the the processing performance?

13) Why the proposed TTP system is useful for education?

14) Where the proposed system can be used?

15) What is the effect of the system on the learning performance?

1.4 Objectives

We impose some substantial objectives in order to achieve our goal. These objectives include (1) conducting research about existing multimedia TTP systems, (2) identifying the semantic relationships and their impact in TTP systems (3) building a multimedia repository that stores pedagogic illustration designed for children, (4) building a semantic knowledge base to store the semantic association between concepts, (5) automatic mining of optimal illustrations that present the user textual content, (6) to critically assess our system by comparing common state of the art TTP techniques (i.e., selection of best keywords, mining using best sentence, etc.) with our semantically based techniques, (7) the usability of system by instructors, and eventually (8) performance analysis of students.

1.5 Proposed Approaches

In order to solve the problem of efficiently presenting educational pictures, we build a multimedia repository that stores a huge number of educational pictures related to the domain of discourse. Each educational image in the repository consists of text within the picture itself. We employ a basic image processing approach to extract text from picture and use it during the retrieval process of pictures. Thereafter, we propose two new approaches to solve the limitation of lack of consideration of semantic relations between words (or concepts). The benefit of this technique is the inclusion of context and meaning of a specific phrase, for example, when the word *Samir* is extracted we will be able to know that *Samir* is a *male* and that a *male* is a *person*. In the first approach, considering an input passage of text that is not linked to any image, we employ conceptual graphs to represent concepts and their semantic associations in a given text [20]. We also employ the use of conceptual graphs to represent the text within each picture in the developed multimedia repository, also taking into consideration the semantic associations of the concepts within the input text. The construction of the conceptual graph is detailed in Chapter 4. The obtained conceptual graph is then used to select the most relevant pictures by matching the similarity of each their accompanied graphs, i.e., matching the similarity of the conceptual graph pertaining to the input text, with the conceptual graph pertaining to the input text. In the second approach called "semantic weight", we consider the semantic distribution of the importance score over terms that are semantically linked in the conceptual graph. The semantic weight calculation of terms is detailed in Chapter 4. Finally, we integrate the above two proposed approaches, conceptual graph and semantic weight, with TTP algorithms for best sentence selection [5, 8] and best keywords selection [11, 15]. This is to consider the semantic association in previous TTP systems, and to obtain a novel TTP system with an effective approach. Thus, the approach can accurately retrieve related pedagogic images in the repository with respect to a paragraph of text input by a user.

On the end user application side, we design a mobile based TTP system which we call “Illustrate It!”. In order to provide a complete educational system, we propose two novel features including the Conceptual Graph Visualization (CGV) that visualizes the content of pictures and the visual illustrative assessment feature that enables users to write sentences to describe the content of a picture using their natural language. Therefore, a grade for the entered text will be automatically calculated to evaluate the understanding of the user. The calculation is done by matching the conceptual graph of the picture and the graph of the entered text.

1.6 Research Methodology

Our methodology follows an iterative process consists from surveying and classifying most related TTP systems [11, 12, 15], assessing used technique, and proposing possible solutions to solve limitations. During the process of our research, we started by studying available multimedia systems designed for education [21]. We examined the importance the Arabic multimedia TTP system to ensure the proposed system is useful. We created a basic version from the targeted system to assess its usability, relevance, and required knowledge. In our preliminary assessment, we employed a pilot study (i.e., smoothing and forecasting) and diverse statistical tests (i.e., ANOVA, ANCOVA, Model Selection, etc.) to identify what kind of multimedia illustrations are valuable [22]. Therefore, we explored the possibility of employing conceptual graphs to generate illustrations. Since a conceptual graph can be obtained from a knowledge base (i.e., ontology), we conducted a survey about existing ontologies designed for educational use [18], and recent studies about question answering over linked data (QALD) [23]. We designed a knowledge base composed from an educational ontology that defines semantically required concepts used in children stories [24]. Consequently, we collected series of children educational stories from several resources, such as, bookstores and online libraries. We enriched our knowledge base with collected stories and illustrations. In addition, we added detailed information about terms by asking different questions, for instance, "*Where do the lions live?*", "*What do they eat?*", "*What are their behaviors?*" and so. We defined most general classes and designed their structure. For instance, the concept *Wild Animal* is sub class of *Animal* concept. Consequently, for the development of the system, we applied the IEEE 1074-2006 standard criterion [21] to develop our system. It is a standard for evolving software project life cycle procedure [25]. It consists from gathering requirements, design, development, and testing. We started with the development process of the ontologies. Therefore, we developed our proposed TTP system that is based on conceptual graph. We compared the results of our system with two former state of the art approaches used in previous TTP systems including selection of the best sentences [5, 8], and selection of the best keywords [11, 15]. Moreover, we integrated our proposed solution with these former approaches to study its effect on the performance. Eventually we studied the usability of our proposed TTP system with instructors and performed descriptive statistic to analyze its impact on learning performance for children.

1.7 Thesis Structure

The thesis is composed of six chapters summarized as follows. Chapter 2 gives a comprehensive background study about TTP systems. Chapter 3 consists of a literature review with regards to the state of the arts works on TTP systems. Chapter 4 describes the proposed multimedia TTP system. Chapter 5 shows experimental analysis and statistical evaluations of the system. Chapter 6 concludes the work and gives a discussion on the potential research direction.

CHAPTER 2: BACKGROUND

In this chapter, we present the required background for the different techniques used in this dissertation. The following topics/concepts are presented: 1) the concept of TTP systems, 2) techniques utilized to retrieve pictures, 3) the background of picture-to-text that is used to extract text embedded inside the image and to convert it into machine readable characters, 4) the background of Arabic text processing, 5) the weighting technique presented in the study, that is used to assign weights to Arabic terms, 6) the best sentence selected and best keywords selection algorithms, as well as 7) the concept of string metrics, 8) the background for semantic knowledge bases and ontological datasets, 9) some available editors that enable a user to edit an ontology, and 10) the concept of the multimedia repository that is used for this study.

2.1 Text-To-Picture Definition

Text-to-Picture is the process of building visual illustrations from words associated with pictures. Pictures can be in a static form such as images, symbols, figures, tables, charts, and maps or in a dynamic form such as animation, or video clips. Text-to-Picture (TTP) applications have expansive prospects in promoting user realization, recognition and understanding [1]. A user can more efficiently realize concepts from text accompanied with visual illustrations than by plain text alone. Over years, diverse research works emphasize that illustrated textual content can broadly improve the performance of people in terms of assorted cognitive outcomes [1, 2]. Supplying the text with pictures and symbols may not be an efficient way to obtain illustrated content. The aim is to provide visual illustrations aligned within the context of a given set of words [4]. Several multimedia TTP systems have been proposed to visually elucidate a given topic [5, 11, 12, 15] by illustrating inspirational images [6] or enriching textbooks with relevant illustrations [7].

2.2 Text-To-Picture Process

TTP is an autonomous process aimed at the generation of multimedia instructional pictures. The process of TTP is a type of automatic multimedia generation through an input represented as a natural text. The generation process of pictures should be confined within the scope of the textual content. It is concerned with inspecting the most appropriate pictures that can represent the text. Normally, the process of mining pictures is based on the best matching of textual content and relevant annotations associated with their pictures. Ordinarily, the best matching refers to the highest score of coincidence. Thus, the matching with the highest score signifies that the picture is most relevant. Fundamentally, there are three types of TTP generation, which include: 1) mining a single image that can represent a set of words [7], 2) mining a set of pictures (symbolic images) that illustrate a series of words [13], and 3) generation of 2D images or 3D models inside a scene [26] to form a virtual environment and to depict words. Two main approaches have been proposed for the mining process, including the selection of the best sentence technique, discussed in Section 5, and the selection of the best keyword technique, discussed in Section 6. Some TTP systems use online search engines as a general source of illustrations, such as Google Image[1], Yahoo[2], and flickr[3] [11, 15]. For instance, extracting pictures from flickr social network stresses the feature extraction of accompanied tags [15]. On the other hand, some TTP systems use their own multimedia repositories as a primary source to illustrate content [12, 13]. For instance, extraction from an ontology needs matching required concepts or logical relations [27]. In fact, the best sentence selection and best keywords selection techniques require processing the given text in order to process words, which we discuss in Section 3, and a numerical statistic technique is used to show the importance of words, which we discuss in Section 5.

2.3 Picture-To-Text Process

The process of transposing the picture-to-text (PTT) component requires the employment of image processing techniques to extract information from the picture [28]. For instance, extracting the text from a picture requires locating the text in that picture. Therefore, characters will be identified and converted into machine editable format [29]. Optical Character Recognition (OCR) is an automated process to convert text in images into machine editable format [30]. It requires pre-processing the image to avoid unneeded pixels (i.e., noisy pixels) during the conversion process. For example, converting the image into black-and-white (i.e., binary image), removal of boxes and lines, adjusting lines, and so. Figure 1 shows an example of a binarized picture.

[1] Google Image: https://www.google.com/imghp
[2] Yahoo: http://www.yahoo.com
[3] Flickr: http://www.flickr.com

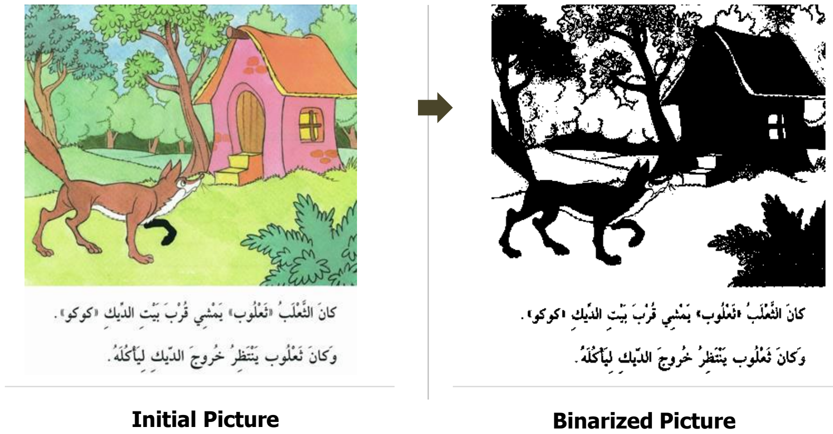


Figure 1. Sample image before and after binarization process

OCR employs training sets for the languages to find the best matching characters in the image and in the training sets [31]. During the matching process, it uses pattern recognition to match between characters embodied in matrix form. Tess4J[4] is an OCR library that supports the recognition of Arabic characters. It uses the Tesseract OCR interface [31] and training dataset to find optimal matching characters.

2.4 Arabic Natural Language Processing

Natural language processing (NLP) is the process of exploring natural text to build knowledge so that appropriate tools can be employed to perform desired tasks. NLP applications include text summarization, information retrieval, machine translation and others [32]. The core of the NLP goal is the automatic understanding of natural

[4] Tess4j OCR library: http://tess4j.sourceforge.net/

language. Thus, a NLP system starts with the analysis of morphological structure of words, and then moves to the sentences to determine the grammatical associations between words [32]. In fact, a word changes according to its context (e.g., past, present, and so), as well as for its prefixes and suffixes. The processing of languages differs from a language to another, for instance, in the Arabic language, a human can use diacritics with the word in order to differentiate its pronunciation from another word. For instance, the word played (لَعِبَ – la'eba) and toys (لُعَبُ – lo'ab) differ by adding different diacritics which completely changes their meanings. Several Arabic NLP (ANLP) applications have been developed for different objectives including [33]: 1) the transfer of knowledge and technology, 2) the modernization and enrichment of the Arabic language, 3) the optimization of Arabic linguistic tools, and 4) the performance of information retrieval, knowledge extraction, summarization, and translation [33]. Thus, knowledge extraction requires identification of important words, which we discuss in the subsequent section.

2.5 Importance Of Words

In diverse data mining applications, finding particular words that distinguish textual strings is necessary to rank results. For example, the textual contents that talk about "*lions*" tend to have several occurrences of terms like "*forest*", "*hunt*", "*run*" and so on [34]. Once these textual contents have been ranked, it will not be difficult to identify the required results. In general, the most frequent terms such as "the" or "and", can be eliminated during the ranking process because they do not carry any significance to identify results. Moreover, the most common stop words (e.g., comma ",", full stop

".", etc.) are often removed before processing text [34]. The Term Frequency - Inversed Term Frequency (TF-IDF) algorithm [34] is often used in order to weigh terms by convenient scores [15]. Thus, a term with a higher TF-IDF score denotes that the term is more important and can be selected to represent a document [34] (i.e., text summarization). For a given term *t* and a document *d*, where *t* appears in number *n* of *N* documents in a corpus repository, the TF equation is calculated as follows [34]:

$$TF\ (t,d) = \sum_{w\, \in d} \begin{cases} 1, & if\ w = t \\ 0, & otherwise \end{cases}$$

and IDF is calculated as follows [34]:

$$IDF\ (n,N) = \log(\frac{N-n}{n})$$

thus, the TDF-IDF is formulated as:

$$TF\ .\ IDF\ (t,d,n,N) = TF(t,d) \times IDF\ (n,N)$$

The details of using these equations are discussed in [34].

2.6 Best Sentence Selection

The best sentence selection technique from a document is a kind of text summarization. Thus, the best selected sentence can briefly represent the aim of the text [35]. The best sentence selection often passes through three phases including: (1) splitting the document into sentences using stop words, such as dot (.), comma (,), etc.; (2) scoring the sentences individually, and (3) selecting the sentence that attained the highest score. Thus, the score of a sentence is equal to the obtained TF-IDF score of all terms

[36]. For the sentence *s* with terms *t* in document *d*, the score of a sentence S(s) is computed as follows [36]:

$$S\ (\ s\ ) = \sum_{t_i \in\ s} TF\ .\ IDF\ (t_i, d, n, N) / \mid \{\ t_i \mid\ t_i \in S\ \ \} \mid,$$

Consequently, a sentence S_i in a document *d* that has the highest score is selected as the best sentence S_b that summarizes the content of the text, and is computed as [36]:

$$S_b = \{ \max_{S_i \in d, S_b \Leftrightarrow S_i} \ (S_i) \}$$

2.7 Best Keywords Selection

The best keywords selection method is the process of selecting the best terms that represent or summarize the content of the text [32]. The method is similar to the selection of best sentences technique, but the selection is based on the level of terms. Thus, a text is split into terms and every term is given a score. Consequently, terms with the highest score are selected as the best keywords [32]. Thus, For the terms *t* in document *d*, the score of a term S(*t*) is computed as:

$$S\ (\ t\ ) = \ TF\ .\ IDF\ (t, d, n, N),$$

Consequently, the set K of the top *m* keywords are selected according to the highest scores of the set of terms $\{\ S(t_1), S(t_2), \ldots S(t_n)\ \}$ in a document as follows [32]:

$$K = \left\{ \max_{m}\ \{\ S(t_1), S(t_2), \ldots S(t_n)\ \} \right\}$$

2.8 String Metrics

In the field of computer science, a string metrics is a measurement unity used to measure the similarity distance between two strings. For instance, the word “Samy” and “Sami” can be considered to be similar. In fact, the string metric gives a weighting number that indicates the distance between two text strings. The Jaro-winkler algorithm is one kind of string metrics that calculates the string similarity [37] between the two text strings. The Jaro similarity for a term with length v and an entity name with length s computes the common characters c and the number of transpositions p. It is computed as follows:

$$S_j(v,s) = \frac{1}{3}\left(\frac{c}{v} + \frac{c}{s} + \frac{c-p}{c}\right)$$

Consequently, Winkler [37] enhanced the Jaro similarity in order to improve the measure. It considers the beginning matching characters of the two strings m. Based on Jaro-winkler string similarity, the Jaro-winkler string distance is defined as [37]:

$$d_{jw}(v,s) = 1 - S_j(v,s) + \frac{m}{10}\left(S_j(v,s) - 1.0\right)$$

A lower value of the normalized distance $d_{jw}(v, s)$ means that the two strings are more similar. Thus, the value is 0 if the two strings are equal, and 1 if the two are completely different.

2.9 Semantic Knowledge Base And Ontology Definition

A semantic knowledge base is a collection of ontologies (ontology . ies) that can be used together to collect required information efficiently and conveniently [19]. The term "ontology" arrives from the Greek term *ovtoc* which refers to "of being" [38]. It is a formal explicit description (i.e., formal language) of concepts in a domain of discourse [39]. We discuss the formal language in the following section. An ontology can store information in a structured form, semi-structured form, and or unstructured one. It defines the relationship between the classes. Classes often refer to concepts. Each concept can have different properties which describe all the viable facts that a class can be associated with. Each property can have specific settings (e.g., multiplicity, roles restriction). For instance, the person cannot have two different blood types (i.e., this can be restricted using multiplicity), or the animal cannot be herbivorous and eat an herbivorous animal (i.e., this can be bounded using rules restrictions). An ontology that has a collection of defined classes and instances refer to knowledge base [40]. Consequently, there are various reasons to use the ontology: 1) to enable the reuse of existing domain knowledge bases; 2) to utilize semantic rules and semantic knowledge extraction, and 3) to maintain analysis of the domain knowledge. Since ontologies supply systems with knowledge and information about particular domains, they become a prime technology for semantic knowledge extraction and reasoning. Web ontologies are a pillar of the semantic web [41]. Semantic web is a standard of World Wide Web Consortium (W3C) that fundamentally stretches the web infrastructure with data interchange based on rational links and logical proofs [39].

Figure 2 shows an instance of an ontology. Yellow nodes are the defined concepts in the ontology. These concepts are connected to each other in a hierarchical taxonomy with the "is a" property. White nodes are the defined instances of concepts. These nodes can be connected between each other to describe their possible relations. For example, the property "can eat" connects the Lion instance to the gazelle instance.

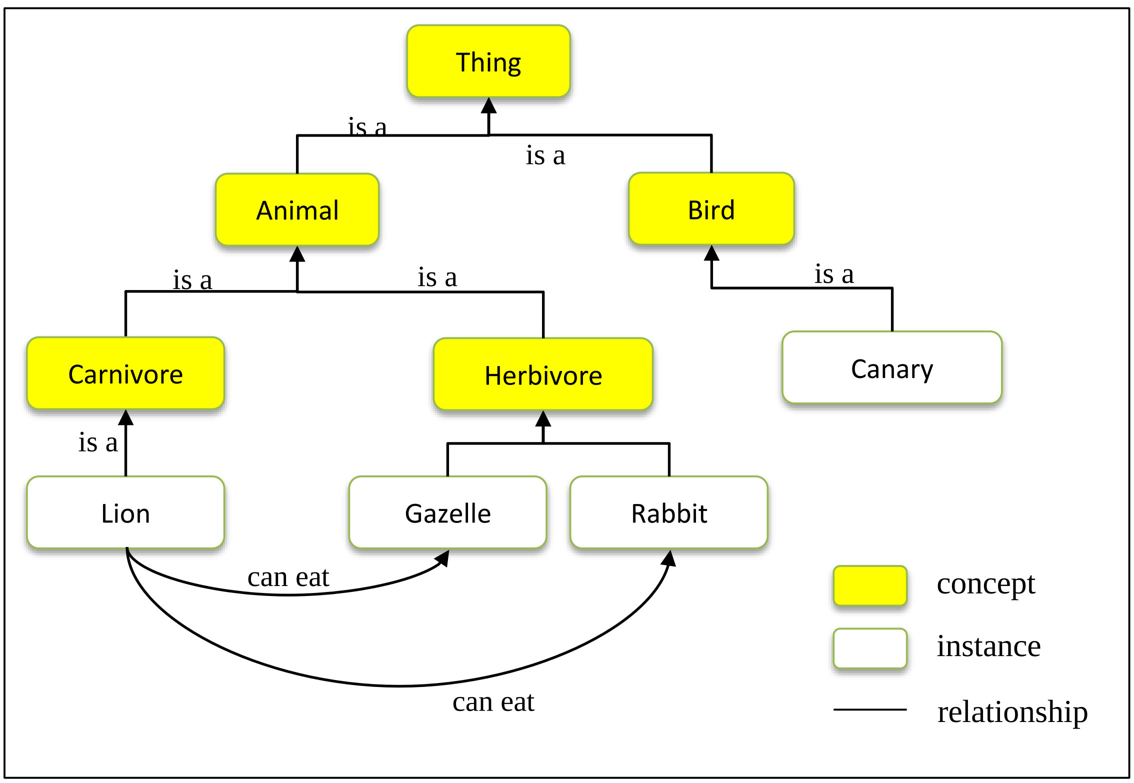


Figure 2. A section of an ontology

2.9.1 Formal Language

The ontology is a formal language specialization of common conceptual knowledge [42]. A formal language is a set of symbols adapted by formation rules. These symbols are alphabets that correspond to specific meanings to expound facts. Formation rules define the expressive form of alphabets (i.e., formal grammar) to guarantee the valid syntactical structure [43]. There are diverse kinds of writing formal languages to represent facts (e.g., objects, relationships, etc.). Foremost, a propositional statement in propositional logic stresses the usage of logic operators and letters to symbolize particular statements [44]. There are five logic operators: a) Negation (~); b) Conjunction (∧); c) Disjunction (∨); d) Conditional (⊃); and e) Biconditional (≡). Hence, according to De Morgan laws, it is possible to transform a propositional statement into another correct form [45], for example:

$$\sim(p \wedge q) \equiv \sim p \vee \sim q$$

First-order logic extends the propositional logic with quantified variables and relationships between objects. A quantified variable is the actual value that describes the object (e.g., Bob). A relationship describes the linkage between an object with another one (e.g., Bob *is a* Person). First-order logic enables the usage of atomic sentences to present the state facts (e.g., Father(Bob, Tom)). Universal quantifiers (∀), existential ones (∃), and implications between logical sentences ($\Rightarrow$ and $\Leftrightarrow$) can be also used to depict facts. For example, the statement *"all persons that are teaching are instructors"* can be represented by:

$$\forall \{ \text{Person}(x) \equiv \text{Teach}(x) \} \Rightarrow \text{Instructor}(x)$$

Moreover, temporal logic extends first-order logic by using the logic of time to make constructive evidence about not only *what an object is* and *what are its relations*, but also *what was*, *what will be*, *what always has been*, and *what always will be* [46]. It is possible to use time intervals to represent the permanence of objects (i.e., show specific time for an object). It is more meaningful than instant modeling, but additionally more complicated. In order to use time intervals, it is contingent to use the precedent ($<$), equality ($\leq$, $=$), inclusion ($\subseteq$),obligation ($\bigcirc$), permission (P) relations and more [47]. For instance, the sentence *"an alive teenage instance comes before an alive adult instance"* can be represented by:

$$\exists x,y : \text{teenage}(x) < \text{adult}(y) \subseteq \text{alive}(x) \wedge \text{alive}(y)$$

2.9.2 Ontological Datasets

Diverse ontologies [48-50] have been developed in recent decades to avail lexical and domain knowledge. They can be accessed online from the web or offline from local storage. The structure of these ontologies differs between each other according to their designated intention. However, it is possible to extract required knowledge with unified query languages (i.e., SPARQL). George [50] proposed a lexical database called

“WordNet” for English language to connect words between each other with labeled logical links (i.e., properties). It organizes synonyms into nouns, verbs, adverbs, and adjectives. In addition, it reserves additional effective integration of traditional lexical databases. Elkateb et al. [51] introduced Arabic WordNet (AWN) to build a lexical dictionary for modern standard Arabic. AWN acquaints the derivative of Arabic verbs, diacritization, and phonetics. Ishkewy et al. [52] proposed lexical Arabic ontology called Azhary. The structure of the proposed ontology enhances the AWN. The structure includes synonyms, hypernyms, antonyms, meronyms, hyponyms, holonyms, and logical relations. It has been compared with the AWN ontology to demonstrate the efficiency of Azhary.

On the other hand, diverse domain knowledge ontologies have been implemented [49, 53, 54], Bizer et al. [55] proposed a book ontology called “RDF Book Mashup”. The ontology has been integrated with bookstores and data sources (e.g., Amazon[5], Google[6], Yahoo[7], etc.) to improve their semantic search engines and crawlers. Amsterdam museum ontology [56] has been developed to transform museum metadata into semantically linked data. The proposed ontology enabled the museum to share its cultural heritage in the linked data cloud. DrugBank [54] ontology has been developed to provide information about drug products, ingredients, and their biological energy that evades different fundamental disorders. Gene ontology [57] has been developed to provide comprehensive and elaborated annotations

[5] Amazon: http://amazon.com
[6] Google: http://google.com
[7] Yahoo: http://yahoo.com

of orthologous genes 'reference' genomes including different key model organisms.

Besides, Semantically-Interlinked Online Communities (SIOC) [48] have been developed to provide services to interconnect conversations of online communities (e.g., weblogs, forums, messages, etc.). This enables cross-referencing to external ontologies such as AtomOwl Vocabulary Specification (aowl[8]), RSS 1.0 Content module (content[9]), Friend of a friend (FOAF) Vocabulary (foaf[10]), and others. Auer et al. [49] developed a considerable large scale ontology from the Wikipedia encyclopedia called DBpedia. The proposed ontology enhances the search capabilities with semantic search over structured andsemi-structured knowledge, interlinking with diverse open datasets (e.g., DrugBank[11], GeoLinkedData[12], Fish of Texas[13], etc.), and converting Wikipedia content into RDF multi-domain datasets (i.e., geography, animals, weather, etc.) [58]. DBpedia datasets can be accessed instantly online by diverse DBpedia user interfaces or downloaded and accessed by third party applications. Up to date, DBpedia is interlinked with more than 30 ontological datasets with 30 million outlinks, and linked from more than 180 ones with more than 39 million inlinks [58]. Eventually, Linked Open Data [53] enables additionally the access of diverse open interlinked-datasets obtainable on the web, such as DBpedia, GeoNames, SIOC, and different knowledge bases. It provides sharing resources and metadata of structured, semi-structured, and unstructured data. It enables the exploration of datasets using the SPARQL query [59].

---

[8] aowl: http://bblfish.net/work/atom-owl/2006-06-06/AtomOwl.rdf
[9] content: http://purl.org/rss/1.0/modules/content/
[10] FOAF: http://xmlns.com/foaf/0.1/
[11] DrugBank: http://wifo5-03.informatik.uni-mannheim.de/drugbank/
[12] GeoLinkedData: http://geo.linkeddata.es/
[13] Fishes of Texas: http://data.fishesoftexas.org/

2.9.3 Conceptual Graphs

Conceptual graphs are considerable as the foundation for technology and knowledge science involving formal language, natural language processing, and others [60]. Conceptual graphs supply information with the representation of knowledge and logic. The descriptive formal model in an ontology refers to a semantic conceptual graph in which concepts and entities denote vertices, while edges denote relations [60]. Using the ontology, the vertices in a derived conceptual graph can have hierarchical relations in which a parent node denotes the abstractive level of the child node. Figure 3 shows an example of a conceptual graph. The nodes “Ahmad” and “Mohamad” are concepts connected to each other through the relation “friend of”. Both concepts “Ahmad” and “Person” are connected through the relation “is a” to the concepts “Person” and “Abstractive Thing” respectively. This means that *Ahmad* is a *person,* and *person* is an abstractive element.

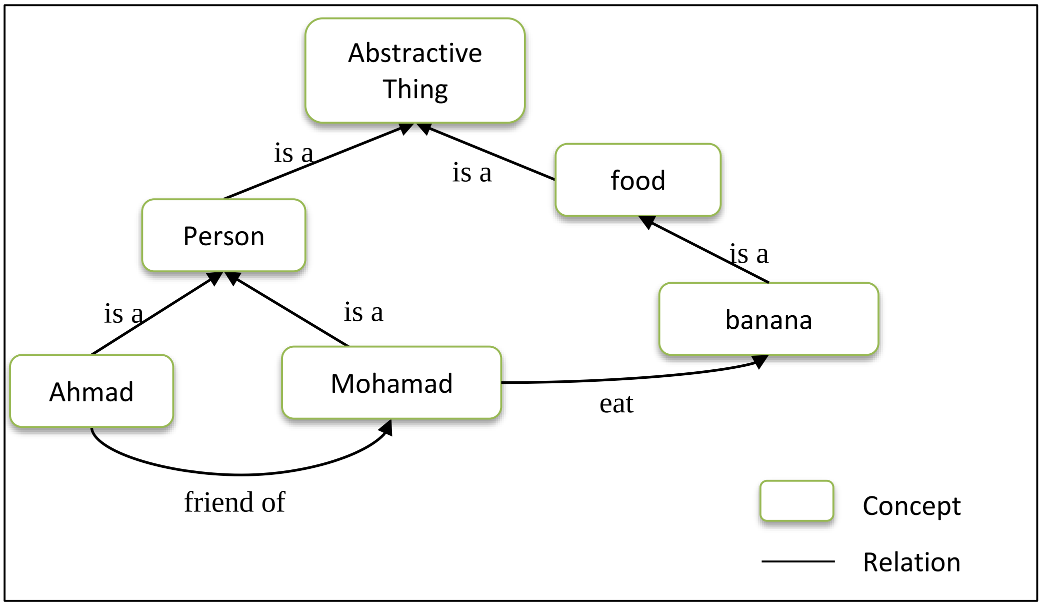


Figure 3. Sample conceptual graph

In graph theory, two graphs are said to be similar if they have the same number of vertices that are connected similarly [61]. Formally, graph *G* and *M* with vertices $V = \{ v_1, v_2, v_3, \ldots, v_n \}$ are similar if there is random permutation *p* of *V* such that {u, v} is in the set of graph edges as follows:

$$E(G) \Leftrightarrow \{ p(u), p(v) \} = E(M)$$

therefore,

$$E(G) - \{ [E(G) \cap E(M)] \cup [E(M) \cap E(G)] \} = \varnothing$$

and therefore,

$$E(G) = E(G) \cap E(M) = E(M)$$

Subsequently, the two graphs are similar if all edges in the two graphs are intersecting. Figure 4 shows the intersection between two graphs that are equal between graph G and graph M. On the right side, the intersection is presented by checking if all edges of the two graphs are equal to each other.

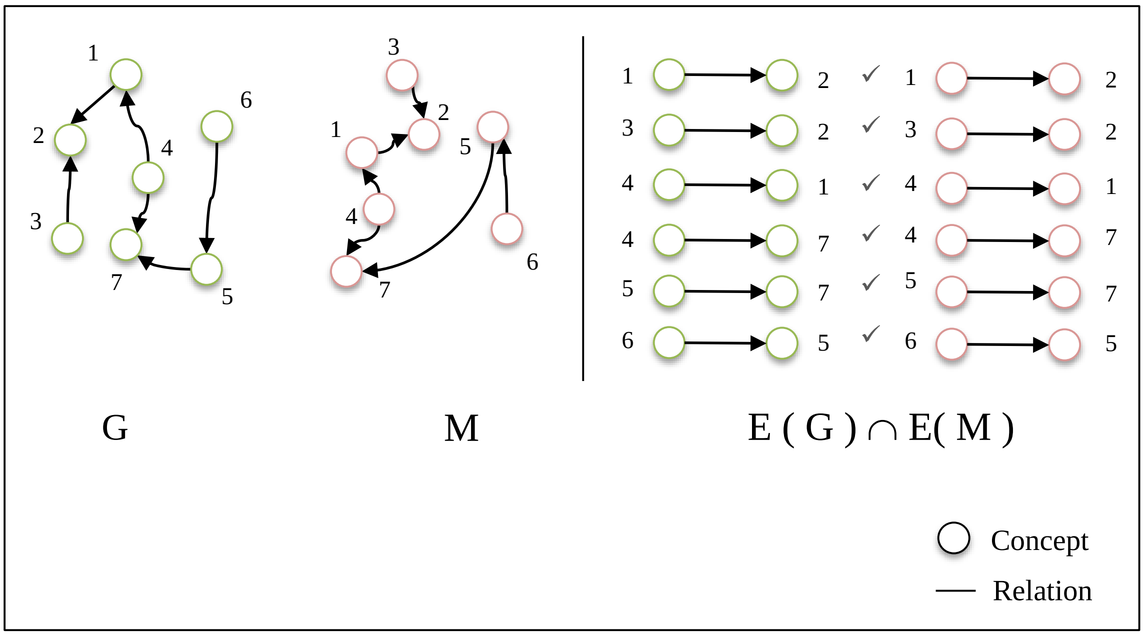


Figure 4. G ∩ M where G = M.

2.9.4 Ontology Editors

Diverse ontology editors have been proposed to modify small and large scale ontologies with graphical user interface (GUI) applications [62-64] or through application program interface components [65, 66]. The services of ontology editors can differ between each other according to their definitive purposes. For instance, the Protégé [62] editor enables the creation of ontologies for any purpose. It does not only enable the creation of classes and their instances , it also supplies different property characteristics and infers new knowledge using a configured reasoner (i.e., HermiT [67]). In this application, declaring a new instance of a specific concept requires the selection of the parent concept, creation of an instance, and filling values of the instance. On the other hand, OntoUML Lightweight Editor (OLED) [63] aims to develop and evaluate unified modeling language (UML) domain ontologies using OntoUML [68].

Furthermore, OwlAPI component [65] has been developed to provide ontology services through the Java programming language. It supports OWL2 [69] to enable the user with creating classes, properties and instances in the ontology. OwlAPI can also operate with reasoner interfaces, such as HermiT [67], JFact [70], and others, to derive new facts. Eventually, diverse solutions have been developed to enable the access of large scale ontologies [64, 71, 72], such as Open Linked data [73]. They enable the management of semantic queries and the search across semi-structured contents.

2.10 Multimedia Repository

A multimedia repository is a storage that contains arranged collections of digital files (i.e., multimedia elements), such as texts, images, graphics, video clips, and others [74]. Since the multimedia repository has more restricted querying abilities than databases [33], it stresses the usage of descriptive information (i.e., metadata) about files. Usage of descriptive files in a repository are detailed in [75]. Typically, descriptive files can be added manually or generated automatically to describe viable facts about digital files in structured form, unstructured form, or both forms. Descriptive files are carried with digital files to give easier searching and access to the digital files. This is particularly useful if the digital files are coded in a form which is hard to be indexed or to be searched for. Often there are rational relationships between the descriptive files and digital files such as the naming reference, coordination description in an image, timecode reference in a video file, and so [74]. Figure 5 shows an example of a descriptive file, named "canary.jpeg.meta", that is used with an image called "canary.jpg". The descriptive file contains a textual description that describes the canary bird as such, "Its color is yellow and black", and "It is standing on a wooden branch". The image file will be used along with its descriptive file in order to retrieve the desired information.

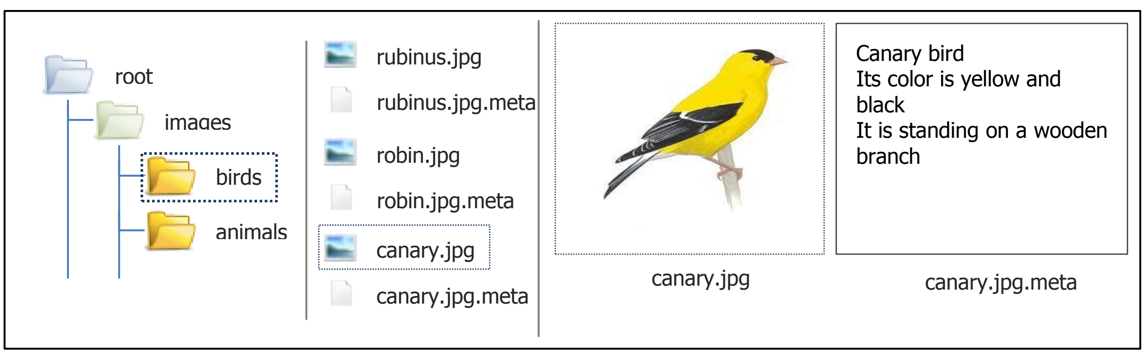


Figure 5. Sample images arranged in unstructured form in directories

2.11 Summary

In this Chapter, we presented the definition of TTP and the required phases to automatically generate pictures. In addition, we presented the different techniques that can be used during the process of TTP including best sentence selection and best keywords selection. We defined the TF-IDF method that can be used as a scoring method. We defined also the concept behind semantic knowledge base and presented some available ontological datasets. A conceptual graph can be obtained the defined relations between concepts and entities in the knowledge base. Thus, concepts and entities represent vertices and relations represent edges in the graph. We gave a background about the different ontology editors that can be used to manage the content of ontologies. Eventually, we defined the structure of the multimedia repository and show how elements can be sorted in directories.

CHAPTER 3: RELATED WORK

In this chapter, a survey of existing state of the art multimedia Text-to-Picture systems is conducted. Thus, the study shows the different techniques used in previous TTP systems and highlights their limitations. The survey gives a step to design a TTP system that overcomes some limitations of existing systems discussed in Chapter 1.

3.1 Text-To-Picture Systems

During the last years, a good number of multimedia systems that illustrate textual content with pictures have been proposed [11, 12, 15]. Zhu et al. [5] have proposed a TTP system that generates pictures pertaining to unrestricted natural text. The system combines the extraction of keyphrases, searching for the informative picture in a local library or on the Internet, and drawing the pictures in their relevant place. Goldberg et al. [8] have developed a general-purpose TTP synthesis to improve the comprehension of users with special needs, such as disabled or elderly people, through examining documents and representing pictorial summaries. The developed system targets the general English language and uses machine learning mechanisms. The pictorial representation of natural text passes through the following phases: 1) extraction of keywords with picturability, 2) usage of semantic role labeling (SRL), 3) retrieval of images using image search engines and ranking them using machine learning methods, and 4) optimization of the layout of the final picture according to the designed layout called ABC layout.

Mihalcea et al. [9] proposed a system that generates pictorial representations for simple sentences. An electronic illustrated dictionary called PicNet [76] has been used with a natural language processing tool in order to generate the required pictorial illustrations. The aim of the conducted research is to examine the level of understanding using the represented pictorial illustrations. Li et al. [10] developed a multimedia system called Word2Image. The system uses Flickr to explore images that provide visual explanations for a set of given words, from the web. Agrawal et al. [7] proposed techniques for retrieving images from the Internet to enrich textbooks with images, and to avoid the representation of similar images in the same chapter. Authors used a corpus of high school textbooks used in India. They used the Amazon Mechanical Turk platform to evaluate their proposed system. Ustalov [77] developed a TTP synthesis system for the Russian language. It combines a natural language processing subsystem, stage processing, and a rendering one. The system uses graphics gallery, thesaurus, ontology, and drawing rules as predefined information resources. A user case study has been conducted to assess the usability and the performance of the proposed TTP system. The developed systems [7, 9, 10, 76, 77] are not designed to provide complete educational system that can be used in classroom environments and to present pedagogic illustrations.

Ustalov et al. [27] presented an ontology-based technique to a TTP synthesis. It uses an RDF/XML based ontology to loose coupling of semantics, galley, thesaurus, and depiction rules. Bui et al. [13] developed an application that automatically converts text to pictures, called Glyph. It is based on natural language processing and computer graphics which aims to illustrate patient instructions. A medication database has been employed in order to obtain and illustrate medication illustrations. The generation of images passe through five processing phase: pre-processing, medication annotation, post-processing, image construction, and image rendering.

Aletras et al. [14] proposed an approach to provide images that are useful to represent topics. The approach generates a collection of images for topics using online search engines. The selection of images is based on graph-based methods which enables the usage of both visual information and text. Jiang et al. [15] proposed a novel assisted instant messaging program, called Chat with Illustration (CWI). It displays automatically illustrations linked with textual messages. The CWI searches for images in a built offline database based on matching keywords. The final representation of the picture is constructed from the set of the most relevant images to the text in the chat. Jain et al. [12] proposed a Hindi natural language processing system called Vishit. It visualizes the text to help in providing universal communication between cultures that use different languages. Authors prepared an offline image repository module consisting of semantic feature tags. Semantic features serve in the selection and representation of appropriate images.

Li et al. [6] proposed a system for children that transforms texts to visual form of fairy pictures called VizStory. It uses a web search engine to explore and find suitable pictures to compose the visualization. Aramini et al. [11] proposed an approach to automatically represent illustrations of short texts using the Google Image search engine

[78]. It presents pictures without supervision or human intervention. The system stresses on the usage of semantic space in order to select the most relevant images. The assessment results of proposed approach were satisfactory.

The above mentioned Multimedia TTP systems are all centered on the generation of illustrative images to represent a set of text. They all have one thing in common, that is to get accurate results in terms of retrieval of images that are relevant to and descriptive of text. These systems were developed with a multitude of applications and domains in mind, and have all contributed generously to the research field of text-to-picture systems. However, none of them have achieved 100% accuracy rate in terms of image retrieval accuracy. The methods presented still lack efficiency and timeliness of impeccable or reliable results. And none of the systems have focused specifically on retrieving results that would be helpful in educational systems or purposes, such that images retrieved are used specifically for a classroom-based context. The system proposed in this study aims to target all these limitations: accuracy of results, efficiency and reduced computational complexity, as well as retrieving images that are appropriate for cognitive use and helpful for the learners' realization of concepts.

3.2 Text-To-Picture Process

The process of translating the text-to-picture passes through different process phases according to how the image will be illustrated. In the proposed TTP system proposed by Zhu et al. [5], keyphrases will be selected first in order to mine illustratable images and draw them in their relevant places. The selection of keyphrases is based on the extracting picturable terms built on a text summarization standard [79, 80]. Therefore, a picturable word is determined if the log-ratio between Google Web Search hit count and Google Image Search one is dominated across other validation features. The selection of images is based on the combination of two sources: (1) manually labeled images in a library, and (2) usage of online search engine to mine absent images from the web. Eventually, the layout of the final picture is based on the determined associations in the selected keyphrases. In the proposed system of Mihalcea et al. [9], PicNet knowledge base pictorial library has been employed to mine pictures. It depends on a augmenting databases placed on the web, and on pictorial pictures added by volunteers. WordNet ontology has been employed as an English lexical resource. The automatic translation of text into pictorial representation passes through several phases, which include: (a) text tokenization and part-of-speech (POS) tagging; [81] (b) employment of WordNet lemmatizer to get the lemmas of words; and (c) usage of PicNet to determine required pictorial illustrations for nouns and verbs.

The proposed system of Agrawal et al. [7] is based on three main components, which include: (1) Assignment of images: The relevance score of images will be computed and assigned in the chapter. The score of an image is computed according its appearance in articles, number of phrases contains the metadata of the image, and the number of matching words in the metadata of the image; (2) Mining images: Two mining algorithms were employed to mine pictures from the web: (a) The affinity algorithm stresses the usage of desired concept phrases to find closest articles and extract their images, and (b) the Comity algorithm stresses the usage of online search engine to mine top images that match concept phrases. Therefore, top K mined pictures will be assigned according to their relevance scores; and (3) Image Ensembling: Borda method [82] has been employed in order to ensemble the image. It allocates correspondence scores for positions in which images will be displayed in. Therefore, images will be placed according to their computed scores.

The system proposed by Ustalov [77] passes through several processing phases: (1) linguistic analysis, the text will be tokenized, syntactically parsed, and associated with semantic representation; (2) depictors generation, a collection of graphical illustrations will be obtained according to the determined semantic representation; (3) final staging for the picture: required pictures will be obtained from an online repository called Nour Project[14]. The obtained pictures will be positioned according to grammatical relations of words.

The TTP system of Bui et al. [13] passes through five phases in order to generate the instructional picture: (1) Pre-processing: the text will be split into sentences. Therefore, words will be normalized using a lexical variant generation (LVG) [83]; (2) Annotation: words such as weight, negation and medication will be located in the obtained normalized text and annotated; (3) Post-processing: unannotated words will be eliminated from the processing phase; (4) Image composition: annotated words will be used in to retrieve required images from the medical database. A set of grammar patterns has been defined to identify how illustrations will be presented. A grammar pattern will

[14] Noun Project: https://thenounproject.com/

be determined from the annotated words. And (5) image rendering: following the obtained grammar pattern, a rule engine will sort images and display them together.

In the proposed system of Aletras et al. [14], top five keyword English terms has been selected from a topic. Candidate images were restrictedly selected from Wikipedia. Google search engine has been employed for the search. The search was bounded to Wikipedia results that have images. SIFT features [84] has been employed to extract the visual information. It can determine the interesting areas in the picture. Eventually, the ranking of pictures is done using similarity scores for both textual and visual content to weight the edges of the graph.

The system proposed by Aramini et al. [11] is based on the analysis of short text. It extracts keywords in order to generate web image search query. The extraction of keywords is based on the selection of two terms most times. If the text is long, therefore, one image will be not enough to get a descriptive illustration. The two keywords are selected using five methods, which include, frequent words, maximum co-occurring words, and three methods based on the principle component analysis of normalized occurrence. Therefore, Google online image search engine has been employed in order to get the collection of images. Images will be selected according to their semantic space. The semantic space is based on Jaccard coefficient metric. It indicates the correlation between keywords and words in the image tag. Eventually, the selection of images is based on the matching semantic spaces of the text and on the web search image. The best matching between the two spaces leads to have better illustration [11].

3.3 Experiments And Performance Analysis

While there is no standard to assess text-to-picture systems [7], experimentations and performance analysis can differ from a system to another [7, 15]. However, the assessment of obtained relevant pictures that depict the textual content can be common between all systems. For the proposed system of Zhu et al. [5], the evaluations of the system performance were conducted in two scenarios: the visual summarization of news articles and illustrations of children's books. The results propose that TTP synthesis has a considerable prospect in increasing the human-human and human-computer communication instruments.

In the system proposed by Mihalcea et al. [9], users have been asked to generate pictorial images using their proposed approach. They considered to assess three ways for understanding levels. These levels include, complete representation of pictorial elements, mixed representation of linguistic form and conceptual illustrations, and the fully linguistic form using only words. The obtained result showed the usage of pictorial illustrations can increase significantly the level of understanding.

The system proposed by Agrawal et al. [7] has employed Amazon Mechanical Turk platform [85] to carry user studies. They considered also different subjects in fourteen books to use them as main corpus for the assessment. The proposed affinity algorithm has been used to mine required images from Wikipedia articles. The comity algorithm used restriction features while mining Wikipedia images. This enabled to compare if the mined images with the two algorithms are similar. In general, the automatically obtained illustrations for the book have been considered helpful to understand subjects.

For the evaluation of the system proposed by Ustalov [77], users has been asked to write five short sentences to obtain their illustrations by the system. A rating survey composed of three questions has been prepared and filled by evaluators. In general, volunteers found the TTP interesting, but it does not completely satisfy their anticipations.

The system proposed by Bui et al. [13] has been assessed with big number of pictorial instructions. A total number of 2000 illustratable instructions were mined for approximately hundred instructional collections. In addition, preliminary evaluation assessment has been conducted to present 49 patient instructions. Evaluators considered all of correctly represented pictures, not presented ones, and wrong ones. The overall agreement ratings of the evaluators reached 73.1%.

The proposed TTP system of Aletras et al. [14] has been evaluated using two datasets created with news articles and random Wikipedia articles. These articles cover different categories such as, sports, politics, and so. The average variance of judges in the human evaluation for image relevance reached 0.88.

The evaluation in the proposed text-to-picture system of Aramini et al. [11] was based on the comparison of the different approaches. In addition, they asked people to enter short texts and fill a surveys based on the obtained results. The obtained assessment results for the proposed semantic space approach were satisfactory.

Table 1: *Text-to-Picture systems*

No	System Name	Domain	Input	Resource / Engine	Output	Reference
1	-	Multi-domain	Key Phrase	Google Image Search / Local Database	Arranged Pictures	[5]
2	-	Special Needs	Key Phrase	Web Search	Arranged Pictorial Pictures	[8]
3	-	Multi-domain	Small sentences	PicNet	Text with Pictures	[9]
4	Word2Image	Multi-domain Secondary Education	Concept phrases	Flickr	One Picture for Concept Phrases	[10]
5	-	High-school textbooks	Concept phrases	Wikipedia	One picture for concept phrase	[7]
6	-	Multi-domain	Short sentences	Noun Project	Top Pictures	[77]
7	Glyph	Medical	Short Textual Instructions	Local Database	Arranged Patient Illustrations	[13]
8	-	Multi-domain	Top five keywords	Wikipedia / Google Engine	Pictures	[14]
9	Vishit	Multi-domain	Short Sentences	Web Search Engine	Arranged Pictures	[12]
10	VizStory	Children text-books	Keywords	Web Search	Fairy Pictures	[6]
11	-	(unspecified)	Short sentences (two keywords)	Google Search Image	Pictures	[11]
12	Chat with Illustration	Multi-domain	Short sentences	Local Database Flickr	Arranged Pictures	[15]

3.4 Summary

In this chapter, we presented diverse available multimedia TTP systems [11, 12, 15] including their generation process and conducted experiments. Table 1 summarizes the reviewed TTP systems. These systems are designed for different domains such as physics, children books, and others. Based on the study, it can be seen that there is a growing interest in the field of multimedia TTP. Existing systems [11, 12, 15] are not fully designed for classroom and educational environment. Thus, the commonly used systems based on the best sentence selection [5, 8] and best keywords selection [11, 15] do not present pedagogic illustrations and they do not semantic associations during the process of illustrating the textual content [20]. This is considered as our fundamental research matter to overcome some limitations in existing TTP systems. Such limitations are caused from the loss of omitting information and the absence of semantic associations between terms. On the end user application level, handheld TTP system is still missing, and retrieved pictures may not be suitable for educational use.

CHAPTER 4: PROPOSED SYSTEM AND TECHNICAL DETAILS

In this chapter, we present the main components of the proposed system. First, we introduce the construction of the multimedia repository. Second, we present the content of the knowledge base. Third, we elaborate on the processing phases of the text to select appropriate images. Next, we show the conceptual graph visualization. Finally, we present a novel visual illustrative assessment scheme. Figure 6 shows an overview of the components of the proposed system.

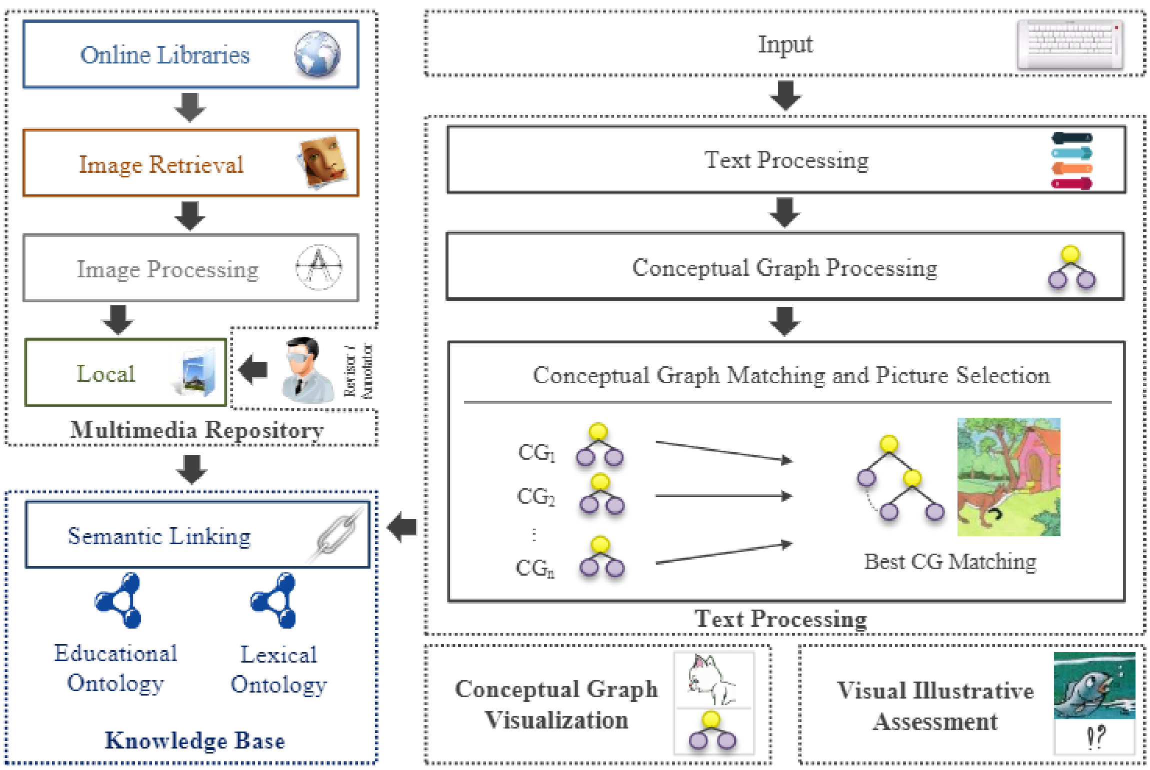


Figure 6. Architecture of the proposed system. It has five components: (1) multimedia repository, (2) knowledge base, (3) text processing, (4) conceptual graph visualization, and (5) visual illustrative assessment.

4.1 The Multimedia Repository

4.1.1 The Educational Illustrations

In order to present pedagogic illustrations, a multimedia repository is constructed to facilitate the access to the pedagogic content. This is because it is difficult to search for pedagogic illustrations on the web. We mainly used the Scribd[15] online book library in order to download educational books and stories, including their illustrations. We targeted stories marked for open printing, sharing, and distributing. We managed our system to periodically download new books and stories to keep our system scalable and to be able to present new pictures pertaining to any content. We input various criteria in order to get intended educational contents, such as: a) the stories should target learners of the age between 5 and 9 years, b) they must contain numerous pictures, and c) they should match selected search keywords. Since the Scribd library is bounded to syntactic search, we used different combinations of search keywords to get the required educational content including pictures, such as childrens' stories (Kesas atfal – قصص أطفال) and child stories (Kesas Tefl – قصص طفل). The textual content in most pictures is embedded within the illustration itself, as shown in Figure 7. For this reason, we employed an image processing technique using OCR in order to automatically extract textual content from the illustration. We detail the text extraction in the following section. We store these illustrations locally along with their extracted textual content. We mark these illustrations as ready for revision in a descriptive file to validate the extracted textual content by a reviewer. We detail the content of the revision file in the Section 1.3.

[15] Scribd: https://www.scribd.com/

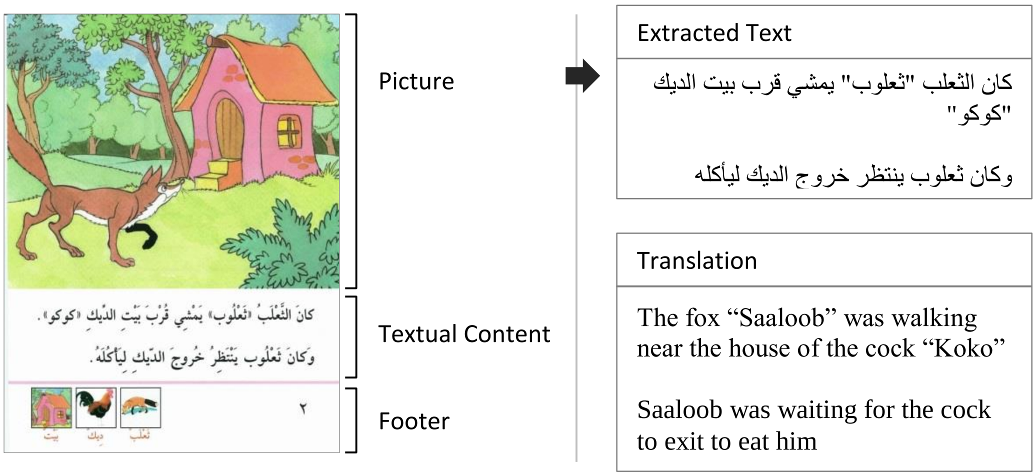


Figure 7. Extracted text from a sample page of the story “Fox treasure”

4.1.2 Text Extraction From Images

Images marked for processing may contain educational text. In order to extract that text, the image is first transformed to binary format. We use color threshold in order to assign a binary image pixel black (0) or white (255) values according to the average color intensity. For a pixel at index $B(i, j)$, we use the average value of red $R(i, j)$, green $G(i, j)$, and blue $B(i, j)$ to binarize that pixel. We use a threshold γ to set the value of a pixel to 0 or 255 as follows:

$$B(i,j) = \begin{cases} 0, & if \dfrac{(R_{i,j} + G_{i,j} + B_{i,j})}{3} \leq \gamma \\ 255, & otherwise \end{cases}$$

Then we detect lines in the binary image B. A line splitter S_p is used to separate possible textual lines. It is defined with minimum number of rows $\delta = 20$ that should not have any black pixels. Each row B_r is identified as a splitter if all its columns j have no black pixels. It can be defined as follows:

$$B_r \leftarrow true \leftrightarrow \{\forall B(i,r) \in B_r, B(i,r) = 255\}$$

For this purpose, we employ the library Tess4J[16] for optical character recognition (OCR) to transform the textual content in the image into machine readable characters.

4.1.3 Local Repository Files

The files in the local repository are automatically generated and sorted. New downloaded books and stories are placed in a new directory named with the current date of the system. For example, if the system attempts to download new files on March 15, 2017, the new directory for these files will be named as "2017.03.15". The new directory will be placed in the root directory of the repository as shown in Figure 8. For each story, we generate one directory for every picture, and we place the picture separately in that directory. For instance, if the story is composed of 10 images, 10 directories will be created for each image. We name each directory with the name of the picture. Consequently, we generate three descriptive files including, textual content "{picture-name}.content", annotation "{picture-name}.tag", and revision file "{picture-name}.rev". The textual content file stores the extracted text from the picture. The annotation file is

[16] Tess4j OCR library: http://tess4j.sourceforge.net/

written manually by a person to precisely describe the content of the image. It is used for automatic assessment purposes. A revision file stores the revision information pertaining to files, such as revision number and author details.

Figure 8. Multimedia repository sample directories and files.

4.2 The Knowledge Base

We have created a semantic knowledge base that consists of two ontologies including, taxonomy ontology, and educational ontology. We detail the developed ontologies in the following sections. At the first stage of the knowledge base design, we have surveyed existing educational ontologies [18] and developed our own ontologies on top of Arabic WordNet [86]. This is done in order to benefit from the existing knowledge bases, with respect to the most commonly used domains, such as animals, foods, and sports. We used the Protégé [62] ontology editor in order to edit our ontologies. In this application, declaring a new instance of a specific concept requires selection of the parent concept, creation of an instance, and filling values of the instance. Therefore, we developed an ontology utility that can be used by users that do not have experience in ontology to edit information in the knowledge base. We detail the developed ontology utility in Section 4.6.

4.2.1 The Lexical Taxonomy Ontology

The taxonomy ontology is a multi-domain ontology which is created to provide detailed structures representing the relations between concepts and entities. We used Arabic WordNet [86] in order to automatically obtain required concepts, entities and their associations. In fact, Arabic WordNet [86] contains detailed structures about concepts. We considered ignoring these details to make the relations between words simple and easy to understand by learners. Figure 9 shows a section of the taxonomy ontology and a section of the Arabic WordNet ontology.

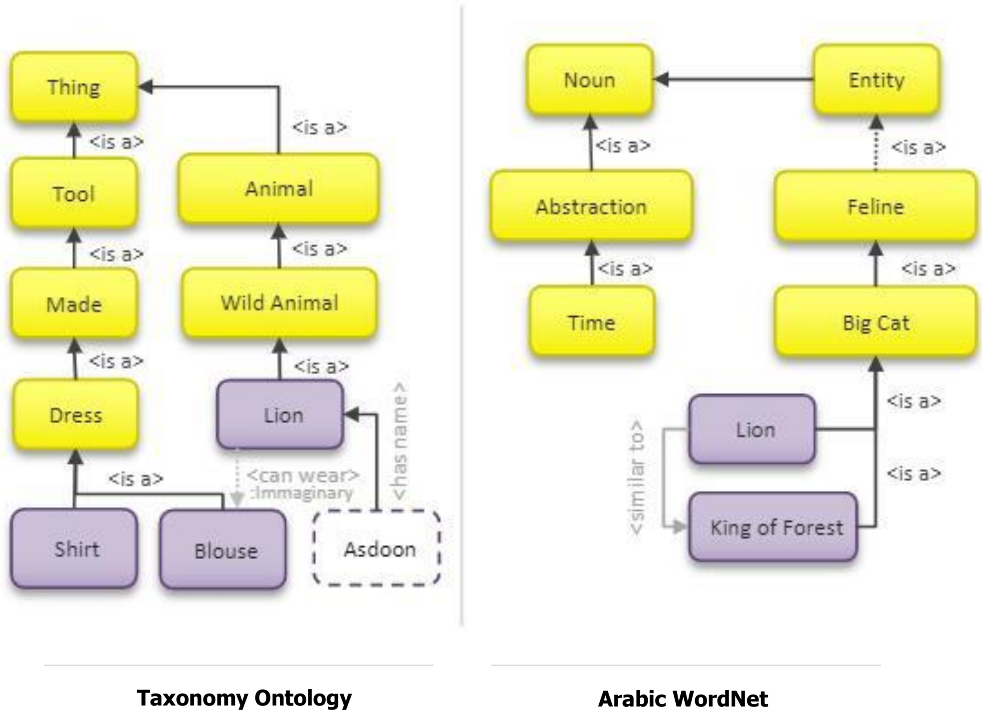


Figure 9. The structure of the entity *lion* using Arabic WordNet and Taxonomy ontology

In order to automate the process of adding concepts and entities, we considered the most repetitive words that are used in the collected stories. This is to make sure that the most repeated items will be used in CGV and facilitate learners' memorization of presented words. We detail the CGV in Section 4.5. The content of the taxonomy ontology was manually revised to add or remove concepts. In addition, Arabic WordNet [86] does not define all possible semantic association entities. For instance, the entity *Lion* is connected semantically with the entity *King of forest*. But there is no semantic relation shows that *Lion can eat meat*. We also consider the imaginary connection between entities. Imaginary connections are those that are not applicable in the real world. For instance, the *Lion is wearing cloths*, will enable us to know from the ontology that *Lion can wear a blouse*. We manually defined a set of entities that can be associated with other entities. We considered contradictory terms through the declaration of negation terms, such as "*not playing*" is a negation of "*playing*". We used the Arabic WordNet [86] lexical ontology to enable fetching synonymous semantic entities. Moreoever, we defined a set of semantic rules to associate entities instead of linking them manually. For instance, the semantic rule of "*all herbivorous animals can eat plants*" will automatically associate all herbivorous entities with grass entities through the property "*can eat*", as follows:

$$\forall (x,y) : herbivorous(\ x\ ), grass\ (\ y\ ) \Rightarrow can - eat\ (\ x,y\ )$$

where *x* is any instance from *herbivorous* concept, and *y* is all instances from the concept *grass*. For example, sheep is an instance from herbivorous, and ryegrass is an instance from grass. Using the precedent semantic rule, we will obtain *sheep can eat ryegrass*.

The names of personalities and actor that are defined in the educational ontology are considered. This enables the learner to fetch possible semantic information about personalities and actors. For example, for the actor named *Birdy*, the learner can get related information about its clothes and what kind of action it can do. We detail the educational ontology in the next section.

4.2.2 The Educational Ontology

The educational ontology defines the semantic structure of stories. Each story entity has a set of properties including *title*, *actor*, *tool*, and *moral*. We used multiplicity restriction to set only one title for each story. The extracted text from images will be used to automatically fill values of some properties. We normally use the first sentence from the first page of the story as the title. We applied some conditions to avoid footnotes of the book being placed in the title. For example, if the extracted sentence contains "*printed in"*, therefore, we will exclude it and not use it in the title. The actor property will store the names of the personalities. For example, for the sentence "Birdy is playing with Mousy", both personalities *Birdy* and *Mousy* will be added as actors in the story.

A list of names was manually defined to store all possible names that are used in stories. This is because there is no standard name for personalities in children's stories, and existing ontologies do not define these names. The names were stored along with their type in order to automatically associate them with their entity in the knowledge base when needed. The *tool* property stores the used tools in the story. The taxonomy ontology is used in order to automatically identify if the entity of a word, is an instance from the *tool* concept. Examples of tools are *ball*, *ewer*, *shirt*, and so on. A set of word combinations is included to automatically fill the moral of the story. These words include *lie*, *hurt*, *fox*, *danger*, *gift*, and so on. For example, if the two words *lie* and *fox* appeared in the same story, we will conclude that the story is talking about cheating. We added these details in the ontology to enable instructors and learners to find stories using any required features. Figure 10 shows a section of the educational ontology.

Figure 10. Section of the educational ontology

The content of the ontology was manually revised in the end, in order to validate its content. The objective was to disable users from finding unrevised stories. This is in order to guarantee the quality of the information. The revision process in the educational ontology involves reviewer understanding of the content of the story and then making required changes in the ontology.

4.3 Conceptual Graphs and Picture Selection

4.3.1 Entity-Relationships Extraction

In order to extract the relationship between entities, we employed a formal concept approach based on the composition of entity-property matrices. It is based on the multiplication of binary matrices [87]. The entity-sentence matrix M_S is constructed according to the occurrence of the entities (e.g., lion, tiger, etc.) and their corresponding sentences. The property-sentence matrix M_P is constructed according to the occurrence of property values (e.g., brown → color, walk → behavior) and their corresponding sentences [24, 88] as shown in Figure 11.

	S1	S2
E1	1	0
E2	0	1

(a)

	S1	S2
P1	1	0
P2	0	1
P3	1	0

(b)

	S1	S2
E1	1	0
E2	0	1

×

	P1	P2	P3
S1	1	0	1
S2	0	1	0

=

	P1	P2	P3
E1	1	0	1
E2	0	1	0

(c)

Figure 11. Entity-Property matrix

The composition of the two matrices M_E will provide the contextual-relation of all entities and their properties as shown in Figure 11. The matrix multiplication is formulated as:

$$M_E = M_S \times M_P$$

As an example, for the Arabic sentence “the brown fox walked near the house – مشى الثعلب البني قرب البيت”, will provide the contextual relations {fox: [behavior: walk]} and {fox: [color: brown]}. Thus, the variable r_{value} of the first row in the matrix contains {*fox*}. It has two properties (or columns). The name of the first property p_{name} is {*behavior*} and the second is {*color*}. Consequently, the value of the first property p_{value} is {*walk*}, and the second is {*brown*}.

4.3.2 Conceptual Graph Construction

The construction of the graph passes through several phases. First, the text is split into paragraphs. Each paragraph is split into sentences. Therefore, each term *t* of a sentence is linked with its relevant entity *e* defined in the knowledge base. The linkage is based on the minimum distance of between the term *t* and the name of the entity e_n in the knowledge base. We use the Jaro-winkler string similarity [28] algorithm to compute the minimum distance between the two texts. We consider a matching similarity between the term *t* and the name of the entity e_n if their Jaro-Winkler distance [28] (d_{jw}) is less than a defined threshold α, see Algorithm 1. We consider the addition of *Al* (الـ) in Arabic such as *Madina* (مدينة) and *Al-Madina* (المدينة) should not make the two strings different. In order to select the best matching between the entity e_n and term *t*, we apply the threshold α on the lowest distance $l = d_{jw}(e_n, t)$. For example, considering the default value of α is 0.05 (α = 0.05), and $d_{jw}(e_n, t)$ for the term *lion* (t = "*lion*") and the entity *lion* (e_n = "*lion*") is equal to 0, then the term *lion* will be associated with the entity *lion* and the value α for these two strings will decrease from 0.05 to 0.

Algorithm 1: Select Best Matching Entity $e \in G$

Input: Graph **G**, term **t**.
Initialization: $selected \leftarrow null, min \leftarrow \alpha.$
for ***each*** $entity\ e$ ***in*** G **do**
  if $(l \leftarrow d(e_n, t)) \leq min$ **then**
    selected $\leftarrow e$
    min $\leftarrow l$
  end
end
Output: selected

After obtaining all the entities of terms, we use each entity and its associated parents to create sub-conceptual graphs G_E from the knowledge base graph G that contains the definition of all entities. The conceptual G_E contains only entities that are associated with the required text, (i.e., input text of the user, or textual content in an image). Figure 12 depicts an example of a created conceptual graph for the entered text "The lion Asdoon wears a blouse and trouser".

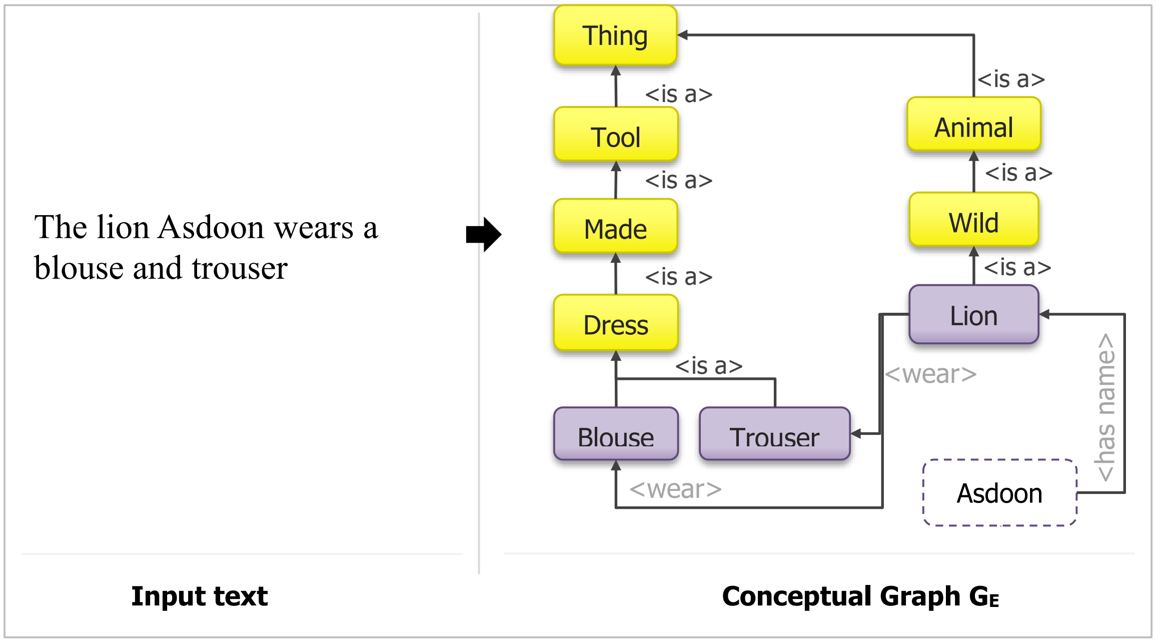


Figure 12. Conceptual graph example for an input text

In Algorithm 2, we show the different required steps in order to create the conceptual graph for the text. We use the function *getEntityByName*() to select the best matching entity which is most similar to the term *t* described in Algorithm 1. We denote the value *e.parent* as the parent entity of the entity *e*, and the function $add(arg_{\{1\}},arg_{\{2\}}..arg_{\{n\}})$ to add entities to the graph. Eventually, the obtained entity-relationship matrix M_E is used to associate entities with each other using the same approach.

Algorithm 2: Build Conceptual Graphs of Text

```
Input: Ontology graph G, sentences S, Entity Relation
       matrix M_E.
Initialization: G_E ← empty Graph.
▷ creating hierarchical relations
for each term t in S.split() do
    selected ← getEntityByName(G, t)
    prnt ← selected.parent ▷the parent entity
    prev ← selected ▷the previous entity
    ▷ adding the hierarchical relations of entities
    while prnt is not null do
        G_E.add(prev, "is a", prnt)
        prev ← prnt
        prnt ← prnt.parent
    end
    G_E.add(prev, "is a", "root")
end
▷ linking entities between each other
for each row r in M_E.rows do
    ▷ passing over properties
    for each column p in r.cols do
        entity ← r.value
        value ← p.value
        G_E.add(entity, p.name, value)
    end
end
Output: G_E
```

4.3.3 Integration With Former Approaches

We have integrated our proposed approach with two former approaches, the retrieval based on the selection of the best sentence, and the retrieval based on the best keywords selection. Thus for every term in the best selected sentence and in the best selected keywords, we create a conceptual graph for them. Therefore, we employ graph matching discussed in the next section in order to retrieve illustrations.

4.3.4 Conceptual Graph Matching And Picture Selection

The obtained graph G_E is used in order to select the best picture. The selection of the best picture I_b is based on the maximum intersection between G_E and the conceptual graphs G_p of the pictures in the multimedia repository R:

$$I_b = \left\{ \max_{G_P \in R, I_b \Leftrightarrow G_p} \left(G_E \cap G_p \right) \right\}$$

We adopted the method proposed by Jaro [37] in order to compute the intersection score. The intersection between edge e_E and edge e_P is considered if all their elements are equivalent, as shown in Figure 13.

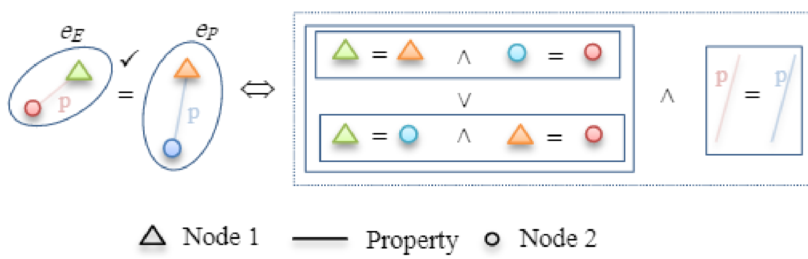


Figure 13. Equivalence of edges $e_E \in G_E$ and $e_P \in G_P$ of the two conceptual graphs.

As shown in Algorithm 3, the intersection score $S(E,P)$ between graph G_E and G_P ($G_E \cap G_p$) is obtained by obtaining the number of intersecting edges with the size (total number of edges) of the two graphs. The score is equal to 1 if the two graphs are exactly similar.

Algorithm 3: Compute Graph Intersection Score

Input: Picture graph $\mathbf{G}_{\mathrm{p}}$, text graph $\mathbf{G}_{\mathrm{E}}$.
Initialization: $score \leftarrow 0,\ c \leftarrow 0.$
▷ *getting the intersection of the two graphs*
for ***each*** *edge* e_P ***in*** $G_P.edges$ **do**
  for ***each*** *edge* e_E ***in*** $G_E.edges$ **do**
    if $e_P = e_E$ **then**
      ▷ *counting intersecting edges*
      c ← c + 1
    end
  end
end
▷ *computing the score based on the intersection*
$S(E,P) \leftarrow \frac{1}{2} \times (\frac{c}{G_E.size} + \frac{c}{G_P.size})$
Output: score

4.4 Semantic Weight And Picture Selection

4.4.1 Semantic Weight

The obtained conceptual graph G_E of paragraph is used to compute the weighting score of entities. The weight of an entity, or its associated term, is obtained using TF-IDF algorithm [34]. We alter the method of [89] and employ the ontological knowledge base to partition the TF-IDF weight over entities. The score is distributed equally on the entity and its hierarchical parents. For example, if the weight of an entity term fox is equal to 40 ($C_w(e) = 40$) and it has three parents, then the weight will be divided over four and it will be distributed on the entity itself and its parent. This will increase the weight of abstractive concepts such as *wild animals* and *animals*. Thus, if a paragraph is talking about foxes and wolfs, we will be able to identify that the paragraphs is talking about wild animals instead of animals in general. This is because the concept *wild animal* will have weighted value even it is not mentioned in the given paragraph. Figure 14 shows an example for the distribution of semantic weight of entities in the short sentence "The fox and the wolf".

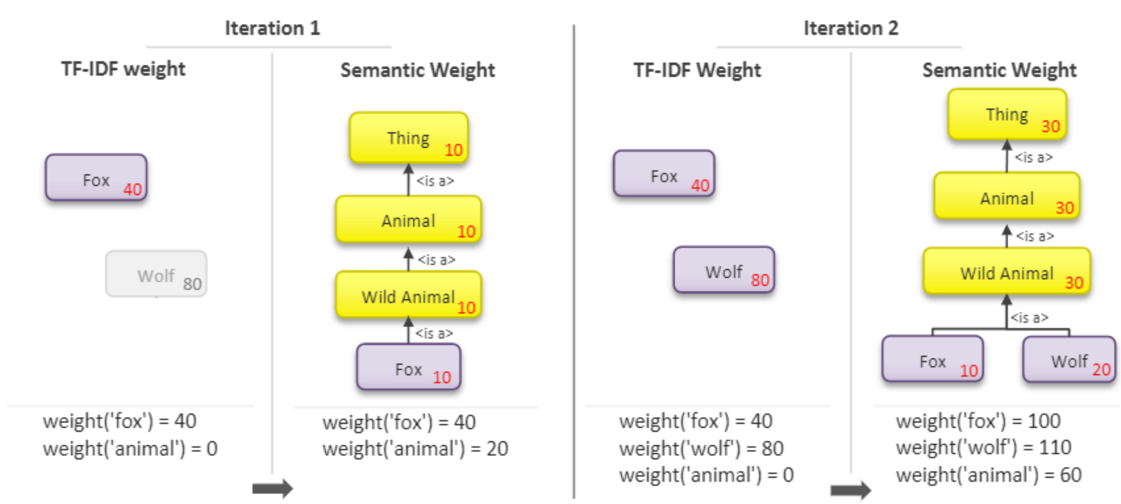


Figure 14. Sample semantic weighting for entities in the short sentence "The fox and the wolf".

Consequently, if a paragraph contains repeated terms, therefore, their semantic weight is multiplied by the number of their occurrences. Algorithm 4 shows the required steps to set the semantic weight of entities. The function *weight*(*t*) returns the weight of the term *t* in the relevant weighting method. And the function *numOfInternals*(*e*) returns the number of semantically linked entities with the entity *e*. Subsequently, the semantic weight of an entity $S_w[e]$ is equal to the sum of its assigned weight including all the weights of its hierarchical parents. Algorithm 5 shows the required steps to compute the semantic weight of an entity.

Algorithm 4: Semantic Weighting

```
Input:  Conceptual Graph G_E, Semantic weights table S_w,
        Text T.
Initialization: weight ← 0, C_w ← empty Collection.
for each term t in T.split() do
    e ← getEntityByName(t, G_E)
    if e in C_w then
        continue
    end
    C_w.add(e)
    weight ← tf · idf(e, T) ÷ numOfInternals(e)
    while e is not null do
        S_w[e] ← S_w[e] + weight
        e ← e.parent
    end
end
Output: score
```

Algorithm 5: Getting Semantic weight of an Entity

Input: Conceptual Graph G_E, Semantic weights table S_w.
Initialization: weight $\leftarrow 0$.
for *each* $entity$ e *in* G_E **do**
  $e_{temp} \leftarrow e$
  while e_{temp} *is not* $null$ **do**
    weight $\leftarrow S_w[e_{temp}]$ + weight
    $e_{temp} \leftarrow e_{temp}$.parent
  end
end
Output: weight

4.4.2 Ranking And Picture Selection

The semantic weight is used to select the best sentences and best keywords. Therefore, the selection of the best illustrative picture I_s is based on the minimum distance between the content of an illustration I_c in the multimedia repository R and the retrieval method r_m that is best selected sentences or best keywords:

$$I_s = \left\{ \min_{I_c \in R, I_b \Leftrightarrow I_c} (r_m, I_c) \right\}$$

Consequently, for the conceptual graph matching, the obtained graph G_E, for best sentences or best keywords, is used in order to select the best picture. The selection of the preferred illustration I_b is based on the maximum intersection between G_E and the conceptual graphs G_p of the pictures in the multimedia repository R as discussed in section 4.3.4. Therefore, in order to rank the results, we employ the modified merge-sort

[90] in order to arrange elements. Thus an element ranked at the lowest index means it matches to the best with the entered text.

4.4.3 Substitution In Previous Approaches

We employed our proposed semantic weight method in order to select the best sentence and the best keywords. Using the semantic weight, the best sentence selection is based on the maximum sum of semantically weighted entities. And the best keywords selection is based on the selection of top terms that have the highest semantic weights. In addition to that, we employ the conceptual graph over the best selected sentence and best selected keywords that are chosen using the semantic weight method.

4.5 Conceptual Graph Visualization And Assessment

4.5.1 Conceptual Graph Visualization

Each illustrative picture is associated with one semantic conceptual graph. We considered the conceptual graph representation is a useful tool for children to comprehend the relation between entities in the picture. The entities in the pictures are presented as nodes of the graph. The hierarchical relation is obtained from the defined taxonomy ontology. The root node is always the abstract node. Figure 15 shows an example of the conceptual graph visualization of a picture.

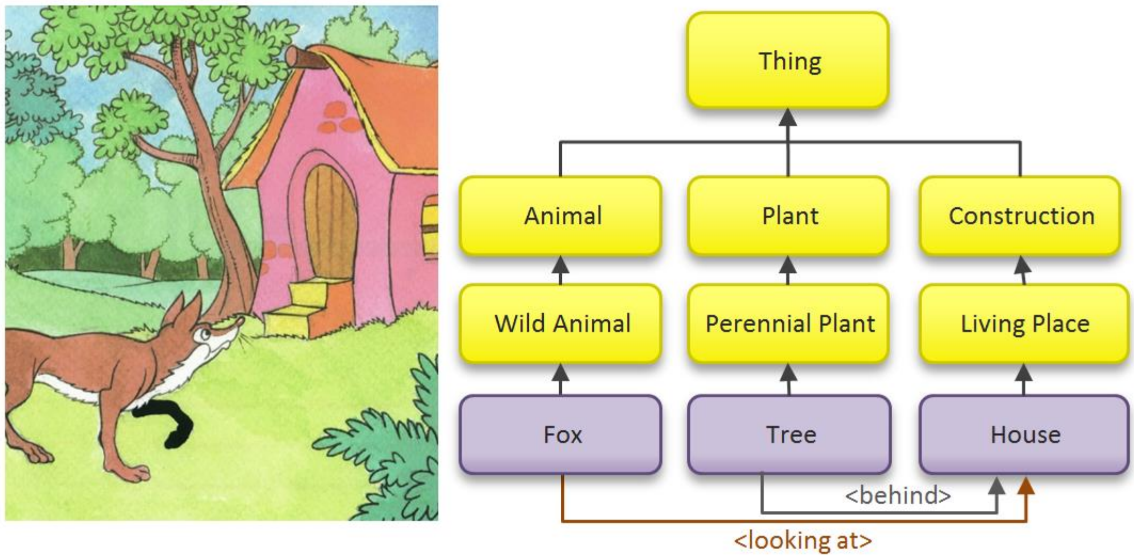


Figure 15. Conceptual Graph Visualization of a picture.

4.5.2 Visual Illustrative Assessment

In order to build a complete educational multimedia system, we enabled our system with three different student assessment models. First, the instructor can select a picture and ask his/her students to describe it. The description can be in form of sentences, or separate words to name entities in the picture. Second, the instructor can ask students to name entities including their hierarchical relations. Therefore, a graph G_S computed from the input of the student is compared with the picture conceptual graph G_P. A grade for a student is automatically calculated according to the maximum number of intersecting edges between the terms chosen by the student and the items in the picture. And third, the instructor can ask the student to write only a summary of a picture. For instance, in the example of *apple*, *banana*, and *peach*, the student has to name the entity *fruit*. The grade G, over total score T is computed as:

$$G = \frac{T}{2}\left(\frac{count\left(G_S \cap \mathrm{G_p}\right)}{G_P.size} + \frac{count(G_S \cap \mathrm{G_P})}{G_S.size}\right)$$

where the intersection (.∩.) is defined as the number of equivalent edges in the two graphs. The equivalence of two edges is defined in Figure 13.

4.6 System Development & Technical Details

In this section, we explain the architecture of our proposed system. We elaborate on the functions of the different components in the system. In addition, we present different screens for the system, and we provide the required technical details to deploy the system to make it available for public use.

4.6.1 System Design

The design of our system is based on the separation of concern (SoC) [91]. We designed the components to provide all the required services and to fulfill the requirement of the educational TTP multimedia system. We applied different required software engineering patterns to make the system robust and incomparable from its kind with existing TTP systems. These patterns include, singleton pattern, factory method, command pattern, and so [91].

Figure 16 shows the different modules in our system. We developed the server application mainly in Java programming language [92] to develop core services. The Controller module is the main module. It is responsible about executing core services. Services include resolving the type of the command, parsing it, and executing its relevant method. The Communication module is responsible about providing the system with the required networking services to exchange messages. We employed ActiveMQ [93] message broker in order to facilitate the management of messages. The multimedia repository module provides the required services to access and fetch stored content. The ontology manager module is responsible about loading the ontology, running the reasoner (i.e., HermiT reasoner [67]), and about providing ontological services, which include, loading entities, associations, and so. The language parser provides the different natural language processing services such as stemming, removing suffix or prefixes. It also enables the interaction with existing lexical parser systems such as Stanford parser tool[17]. The Helpers module provides common global services, which include, string features, file features, and so. The plugins module provides the ability of loading external libraries. These libraries can be placed with the system and loaded on demand. Such libraries include image processing, optical character recognition, and so. The applicable programmable interface (API) is a module defines all the interfaces, enumerations, abstract classes, and others, that are used in the system. It can be used to

[17] Stanford Parser: https://nlp.stanford.edu/software/lex-parser.shtml

access common features from the system itself or from another system. Eventually, the Logger component logs different information that can be used for statistical analysis. Information include the performance of the system, logged users, errors, and so.

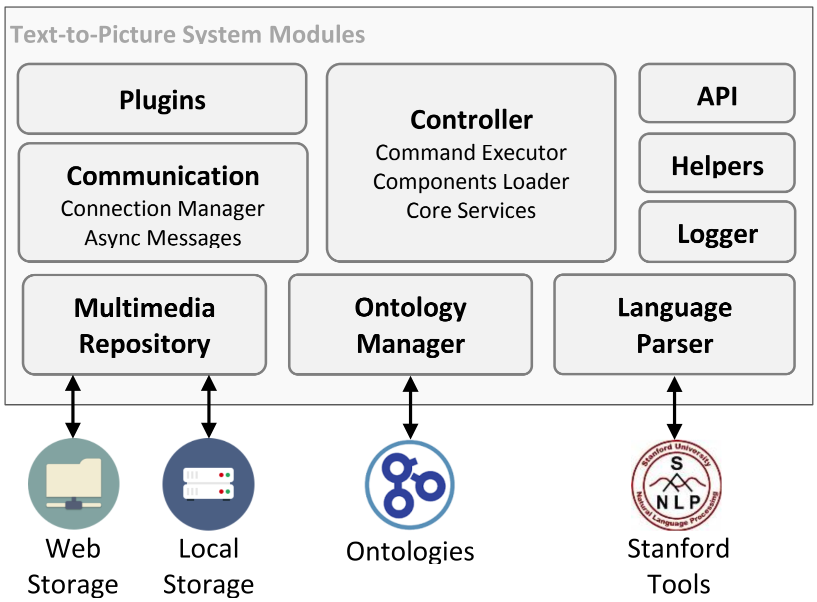


Figure 16. System Modules.

The modules of the system can be reused in different projects. We added supplementary explanations to keep the source code maintainable and understandable. We designed classes to make them easy to understand and accessible straightforwardly. For instance, if the user wants to read a file, he/she can call the method, Helpers.FileUtils.ReadFile(*file_path*).

With the implementation of the command pattern, it is possible to encapsulate a command with its parameters to get the desired result. This can be useful for interested parties (e.g., educational center, school, etc.) that want to integrate the system with their existing solutions. The system fetches the required functionality to execute it. We applied the dynamic execution of functions through searching for classes and required methods inside the code. This technique can be often used in flexible plugin based architecture for systems designed for large scale development. Thus, these functionalities can be placed separately in libraries according to the offered features of the system. For instance, a set of plugins (dll, or jar files) can be loaded from external resources depending on the need.

4.6.2 System Architecture

We applied the client-server architecture in our system. The system is composed of three applications, which include: server application, management utility for administrators, and the hybrid application for end users. The substantial benefits of this architecture are the following: 1) to dissociate the logic of processing information from the resulting presentation execution; 2) to combine the ability of supporting diverse platforms; 3) to reserve the reusability and efficiency of packages; 4) to provide integration support with previous educational systems; 5) to transfer information over a secure connection (i.e., over the SSL encryption technique); 6) to reserve real time server response; 7) to reserve messaging techniques and track system failures remotely; 8) and to reserve high-velocity of processing Arabic text with rapid responses while presenting consequences.

4.6.3 Server Placement

Multiple options are available to make the system available online for instructors and students, ranging from the free, topical, to the advanced customization provided in charge of commercial companies. We studied the different options to make our system constantly available, from any place and at any time, taking into consideration a suitable frugal budget. These options include: (a) contacting the information technology (IT) support at university to setup a public IP address with a dedicated machine, (b) contacting a telecommunication service provider (i.e., Ooredoo), (b) possessing a hosted storage to place required multimedia content, (c) configuring real-time communication with push notifications [94] (e.g., Google push notification [95]), (d) managing our own dedicated server to place required applications, and others. The IT support at Qatar University informed us such processes may take a long time and may not be carried out due to some security and policy constraints. At Ooredoo, the cost to start the rudimentary phase for our system was fairly expensive.

Therefore, we continued to estimate the cost of the different hosting providers. We noted the price of renting dedicated machinery increases remarkably on the long run if bigger storage is selected. For instance, the buyer needs to pay approximately 30USD per month for 40GB storage in a dedicated server, while 90GB storage costs approximately 60USD per month. Therefore, we opted to have a dedicated server with

40GB storage for 30USD and also 100GB web hosted storage for approximately 100USD per year. The selected option was significantly cheaper in price than requesting a dedicated server that has 90GB. Therefore, we contacted GoDaddy[18] support and asked them different technical questions about the different options. These questions include the accessibility of the server, configuration of the required ports, available storage, and others. This was done to ensure the selected options will work properly before the purchase. We also asked about the possibility of having a discount on the purchase. The company offered a cost of 319.90USD for one year instead of paying 359.88USD.

4.6.4 Server Setup

We configured the remote server with Microsoft Windows Server 2012 R2. It has 2GB RAM with 40GB storage. The server can be accessed with Microsoft remote desktop connection (RDC) as shown in Figure 17. By default, the server does not have installed applications. We installed all the required software and configured them. For instance, we installed Java Runtime Environment (JRE), configured available port numbers (e.g., port 61614 for websocket, port 61613 for stomp socket, etc.), and so on. As depicted in Figure 17, we used the Server Manager to manage roles, features, groups, and others. Therefore, we transferred our files into the remote server. The files include the server application, libraries, and the knowledge base. In order to preserve the server storage for executable applications, we placed the multimedia illustrations on the hosted storage. The content of the hosted storage can be accessed through the hypertext transfer

[18] Godaddy Hosting: www.godaddy.com

protocol (HTTP), a protocol for the collaborative, distributed, hypermedia acquaintance system.

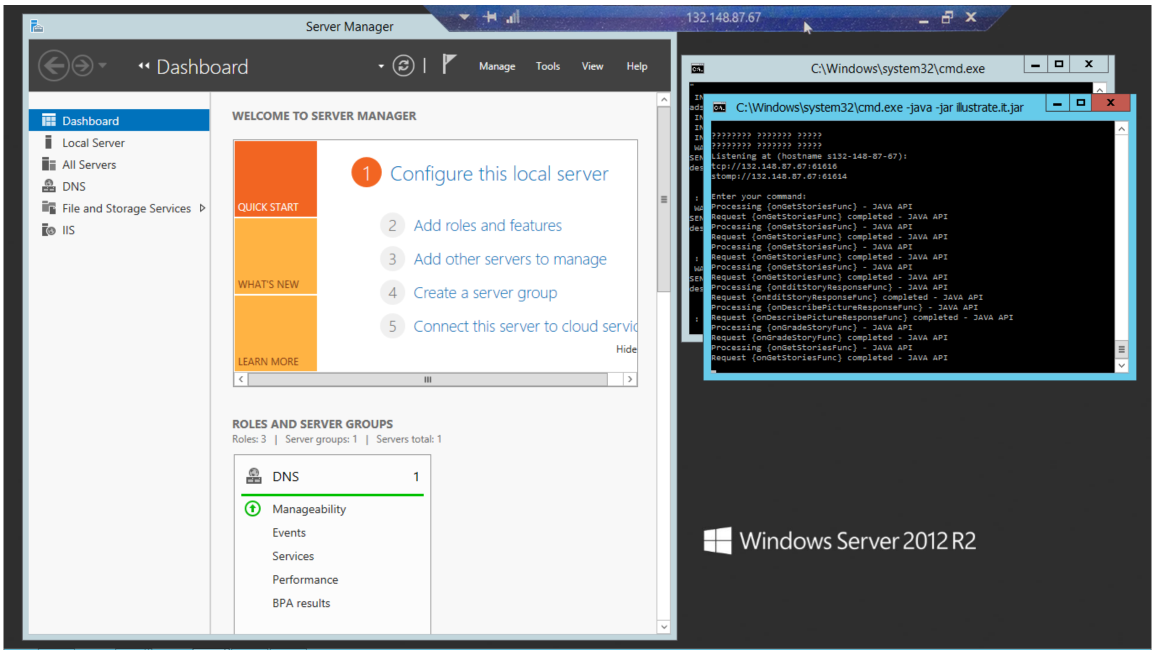


Figure 17. Microsoft Windows Server 2012 R2 snapshot, “Illustrate It!” server is logging information in the console window.

To keep the content of our dedicated server organized, we arranged files in three directories: illustrateit.server.app: contains the released files of the server application, multimedia.repository.data: contains the annotations of pictures, and message.broker: contains the files of the used message broker (i.e., ActiveMQ). In the other hand, we arranged the content on the hosted storage in two directories: 1) illustrateit.client.app: contains the end user hybrid mobile application, and 2) multimedia.repository.illustrations: contains pictures that can be used to depict the textual content. We also placed the mobile application on Google Play. It can be found with the name “Illustrate It!”.

4.6.5 System Operation

Whenever the server is launched, it loads its packages and components as shown in Figure 18. It loads the information from the knowledge base, text parser components, and finally it opens a TCP connection to wait for the users’ requests. Hence, all phases are processed on the server side to decrease power consumption on the client side (i.e., mobile device). After starting the Client end application, it assays to bind the server, which validates and sets up the connection if the provided authentication is veracious.

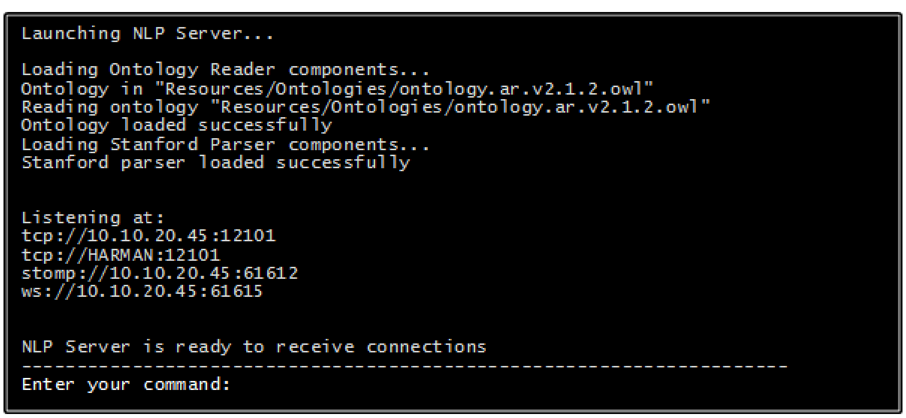


Figure 18. Server startup.

Upon an effective connection trial, the user will be eligible to enter or open existing Arabic stories and process them. When the user selects a story to process it, a client request is composed of, 1) the header "ProcessStory", and 2) the content of the story, placed together in a JSON format. Once the server receives the request from the client, it reads the header of the message and executes the intended function accordingly. Figure 19 shows the workflow to process the paragraph (i.e. educational text) in order to retrieve relevant illustrations.

The execution of commands is done concurrently. The server is capable of executing commands simultaneously in different threads. This is done to get the optimal performance during the processing phase of requests. Figure 20 shows the client-server sequence diagram to establish web socket connection and to execute a command.

During the processing phase, the server logs all performed steps to allow the administrator to be able to track the required information, perform certain analysis, and detect any occurred failure as shown in Figure 21. Moreover, the administrator can configure the server to report by email the following scenarios: 1) critical failures, 2) maximum connected users is reached, or 3) maximum memory capacity is reached.

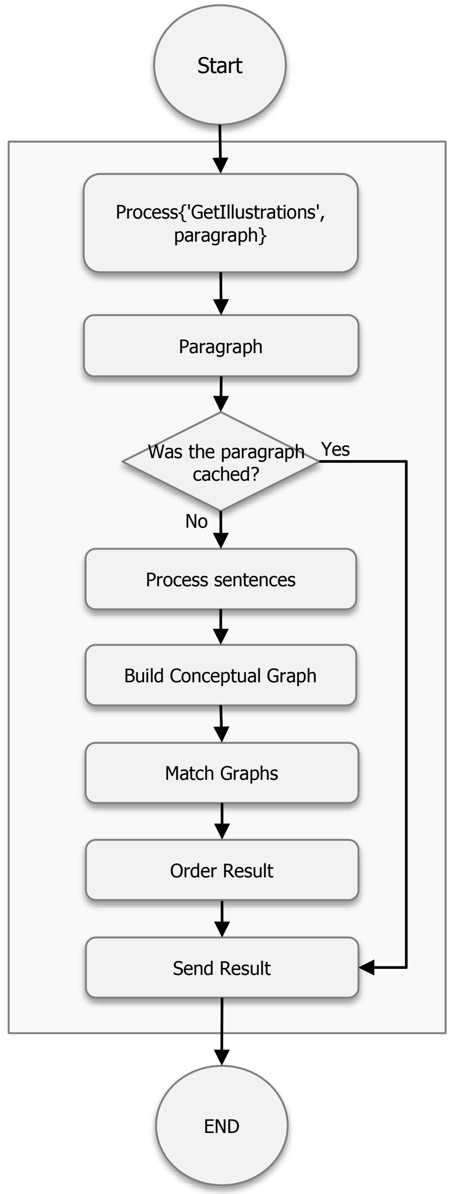


Figure 19. The workflow for illustrating the paragraph.

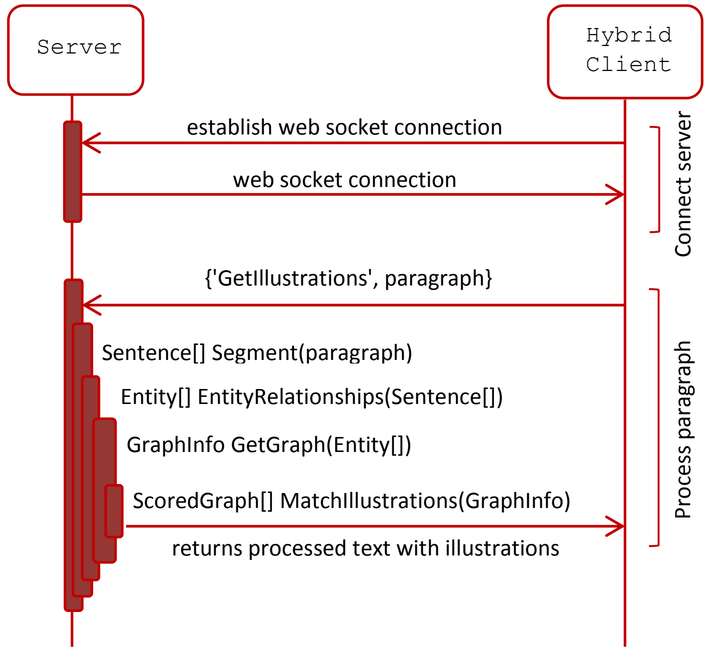


Figure 20. A client-server sequence diagram.

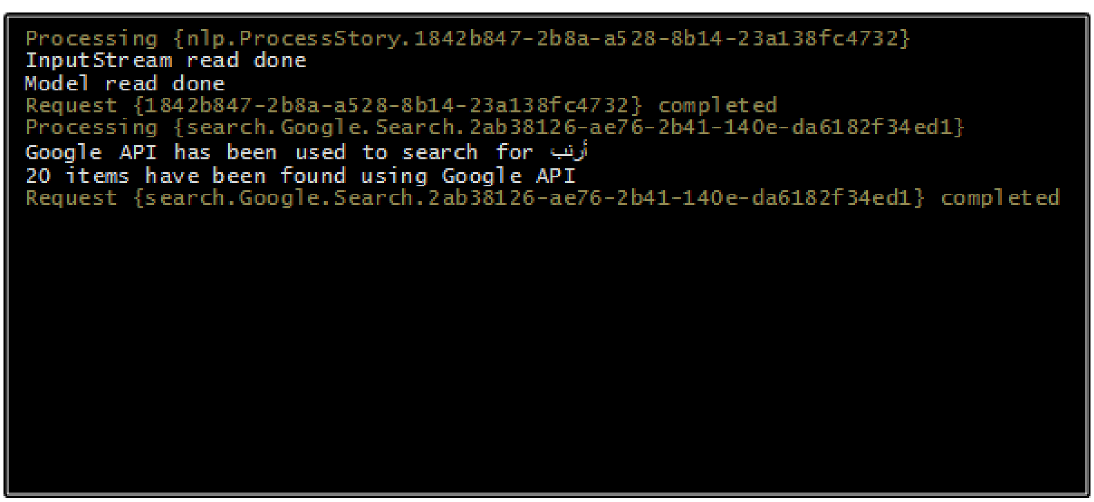


Figure 21. Processing the user requests.

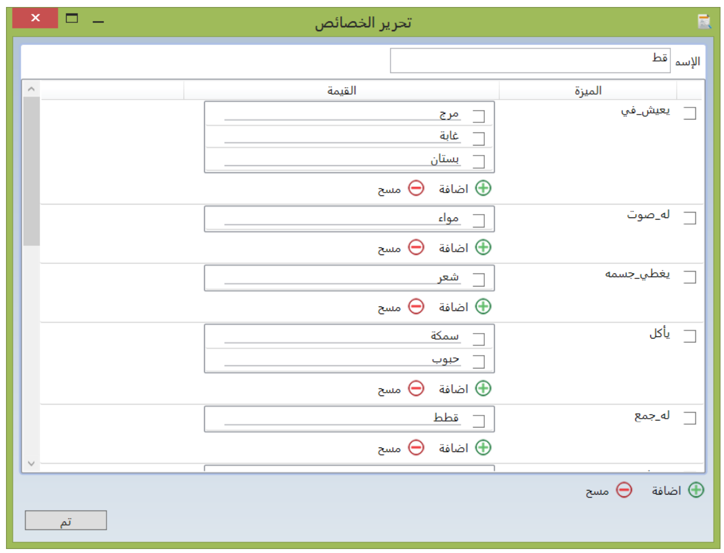
4.6.6 Knowledge Base Management Utility

The ontology management utility has been created with Windows Presentation Foundation (WPF) framework [96], a graphical rendering user interface by Microsoft[19]. It enables a complex binding of structured data automatically inside customized controls. The binding process does not require program processing for each structured instance separately. For instance, a defined control can be assigned directly to a collection with ListViewOntologyItems.ItemsSource = collectionOfItems. We used the C Sharp (C#) programming language to write the logic of the application. We created the required user controls to avoid the repetitive implementation of user interface components. Figure 22 shows different snapshots for the management utility applications.

[19] Microsoft: https://www.microsoft.com

(a)

(b)



(c)

Figure 22. Management utility application. It shows three screenshots: (a) The main page, (b) Ontology items list, and (c) the editing page of ontological entities.

4.6.7 Hybrid Application

The end client user application is a hybrid application created with HTML5. The notion of hybrid application refers to the ability to access embedded native functionalities of an end system. The implementation of such an application enables it to operate on the diverse kind of platforms. These platforms can be smart phone based systems (e.g., Android, iOS, etc.), computer based systems (e.g., Windows, Linux, etc.), or directly based on web browsers (e.g., Google Chrome, Safari, etc.).

We have used web socket programming in order to establish the fully-duplex connection with the server. We have employed Streaming Text Oriented Message Protocol[20] (STOMP) API as a communication protocol to exchange messages. Cordova library has been used to build the application for smart handled devices. Treant library[21] has been employed to draw the conceptual graph that is associated with its picture. Moreover, we have designed pages that are flexible with the different sizes of screens. Additionally, we focused on the look and feel of the application to ensure user enjoyment while using the application. Figure 23 shows various screens from the mobile application.

[20] Stomp Protocol: http://jmesnil.net/stomp-websocket/doc/
[21] Treant: http://fperucic.github.io/treant-js/

Figure 23. Illustrate It! hybrid application snapshots: (a) home page, (b) reading a story, (c) editing story, and (d) automatic grading assessment.

4.6.8 System Availability

We examined different software markets to make the application widely available for users around the world. These markets include, Windows App Store[22], Google Play Store[23], and others. We considered making the application available on Google Play Store. We investigated the distribution of educational applications over the different versions of android systems as shown in Figure 24. Based on the study, we decided to make our android application compatible at least with Android 4.0.

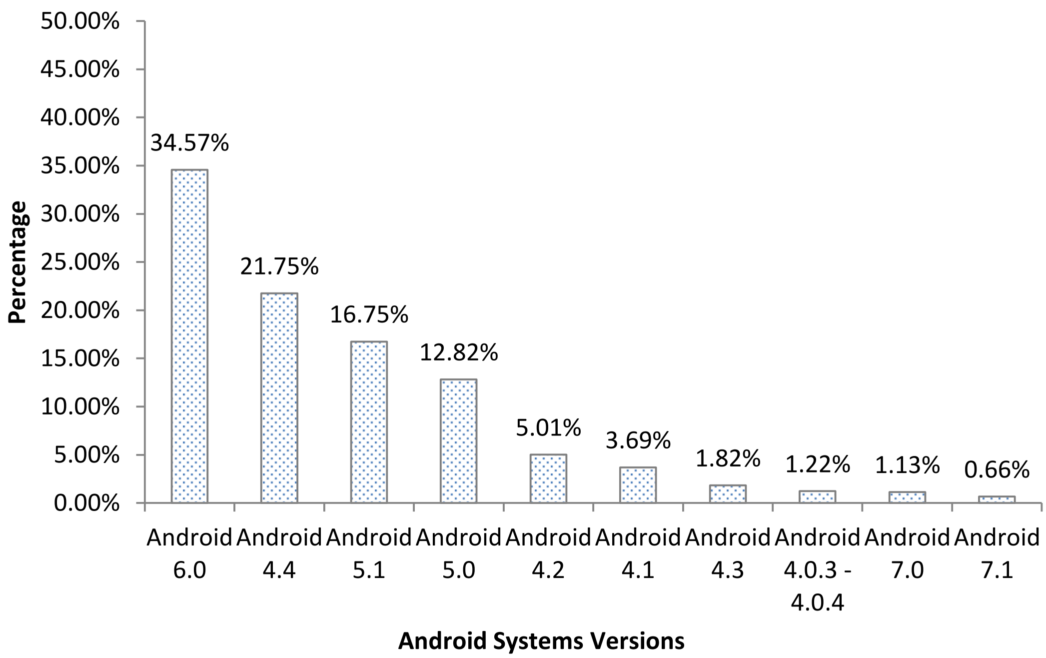


Figure 24. Distribution of educational application for top 10 Android versions.

[22] Windows App Store: https://www.microsoft.com/en-US/store/apps/windows
[23] Google Play Store https://play.google.com/store

We created a Google Developer account, signed the application, and prepared required items for submissions (e.g., registration, description forms, screenshots, etc.). We have submitted the application to get official content review from the International Age Rating Coalition (IARC) and participating rating agencies. This to verify the application is reputable and meets the minimum publication standards. The application has been accepted and received a global certificate record. The obtained content rating results are: the Brazilian Advisory Rating System (ClassInd), the Entertainment Software Rating Board (ESRB), and the German Entertainment Software Self-Regulation Body (USK) rated the application to be usable by everyone, International Age Rating Coalition (IARC) and Google Play South Korea rated it to be usable by the ages 3 and above. Figure 25 shows the rating of the different software rating organizations including the IARC certificate identifier.

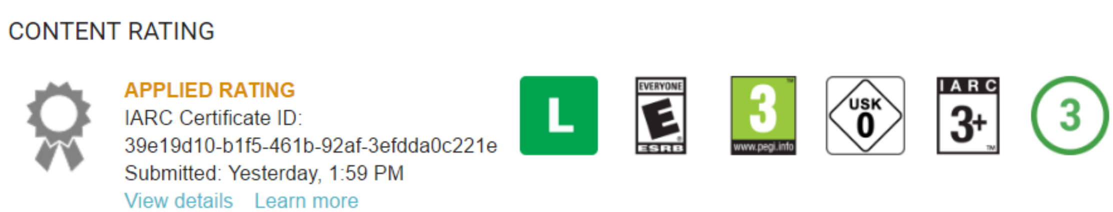


Figure 25. IARC content rating global certificate.

4.7 Summary

In this chapter, we presented the different components of our proposed multimedia TTP system. We showed the process of collecting educational illustrations for the multimedia repository and the elements of the semantic knowledge base. Consequently, we presented our proposed approaches including the conceptual graph, semantic weight, and the integration with existing approaches used in previous TTP systems based on the best sentences selection and best keywords selection. Moreover, we presented two novel features encompassing the conceptual graph visualization and visual illustrative assessment. The conceptual graph visualization is based on the content of the picture. It shows how words are linked between each other. And the visual illustrative assessment enables the user to describe the picture using modern language and obtain the score automatically. Eventually, we detailed the technical development details of the system and the required settings to make the system accessible. Moreover, we made the end-user mobile application available on Google Play Store.

CHAPTER 5: EXPERIMENTS AND STATISTICAL EVALUATIONS

The proposed multimedia TTP system named as "Illustrate It!" has been implemented as discussed in the previous sections [20]. The system is compared with the *Best Sentence Selection* and the *Best Keyword Selection* based systems. Also, the combinations utilizing the conceptual graph approach in both methods are implemented and compared. The comparisons are performed for general retrieval, order of relevance, immediate illustration, and mining performance in terms of execution time. In addition to that, usability assessments and learning improvement comparisons in children with special needs is also performed. The performance evaluation of the proposed system depicted significant improvements in retrieving images illustrating the text. In order to conduct evaluations, we compared the different retrieval methods. These methods include:

1) The proposed Conceptual Graph (CG): The conceptual graph is constructed for the entered text as described in previous sections.
2) Best Sentence Selection (BSS): The text is split into several sentences. The sentence with the highest weight is selected. The Jaro-winkler [37] string similarity is used during the retrieval process.
3) Best Sentence Selection using Semantic Weight (BSSW): The text is split into several sentences as in BSS. The sentence with the highest semantic weight is selected.
4) Best Keywords Selection (BKS): Keywords of every paragraph are weighted using TF-IDF method. Top five keywords are selected and used for the

selection of illustrations.

5) Best Keywords Selection (BKSW): Keywords of the entered text are weighted are weighted using the proposed semantic weight method. Top five keywords are selected as in BKS.

Consequently, a conceptual graph (CG) is constructed for the selected sentences and keywords. We conduct the same evaluations using the integrated CG approach for the different methods as follows:

6) Best Sentence - Conceptual Graph (BS-CG): The best sentence is selected as described in BSS method. A conceptual graph is constructed for the best selected sentence and used during the retrieval process.
7) Best Sentence - Semantic Weighting (BSW-CG): The best sentence is selected as described in BSSW. A conceptual graph is constructed for the selected best sentence.
8) Best keywords - Conceptual Graph (BK-CG): A conceptual graph is constructed for the selected best keywords obtained in BKS. The conceptual graph is employed to retrieve the relevant pictures.
9) Best keywords - Semantic Weighting (BKW-CG): A conceptual graph is built for the semantically chosen keywords in BKSW. The conceptual graph is then used to obtain the required pictures.

In some cases, the same picture can be used appropriately to describe several paragraphs. In order to clarify the evaluation process, we exclude such pictures from the test dataset. This is to make sure the evaluation is considering only the intended picture that is associated with its accompanied paragraph. In addition to that, a retrieval failure is counted if the index of the required image is greater than the specified rank β. We use five β values to assess different retrieval methods, β=500 or top five hundred images, β =100 or top hundred images, β=50 or top fifty images, β=10 or top ten images, and β =1 for immediate retrieval.

5.1 Retrieval Comparisons Using CG Methods

Figure 26 demonstrates the results of the general retrieval experiment for the different methods before and after the employment of CG method. In this study, the order of images is not considered. Only the retrieved images are considered. The proposed method, CG, achieved the highest score compared to the other methods. Moreover, the retrieval using BSS and BKS improved significantly when the CG is integrated with these methods. This shows the effectiveness of the proposed CG approach.

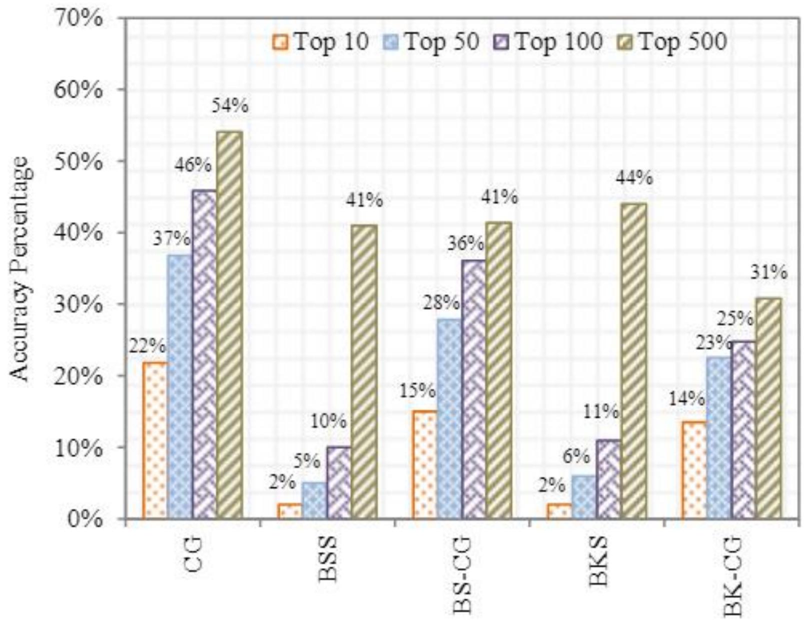


Figure 26. Comparison for the general retrieval between retrieval methods.

5.2 Order Of Relevance Comparisons Using CG Methods

In the experiment depicted in Figure 27, the impact of different methods is considered when sorting the required images in the retrieval process. The retrieval score S_m is used to rank the retrieval methods. Thus, if a retrieval method sorted, the required image with an index i lower than all other indices i_o of other retrieval methods, then, the score S_m will be incremented by 1. If more than one indices i_s of retrieval methods are equal and lower than other indices i_o of other retrieval methods, then the scores of all retrieval methods S_k with low indices will be incremented by 1. The order of relevance conditions can be summarized as follows:

$$\begin{cases} S_m \leftarrow S_m + 1, & if\ (i < i_o) < \beta \\ \forall (S_i \in S_k): S_i \leftarrow S_i + 1, & if\ (i_s < i_o) < \beta \\ skip, & otherwise \end{cases}$$

The retrieval methods using CG got the highest percentage in comparison with other retrieval methods. Moreover, our proposed approach has shown significant improvement when integrated with previous methods. However, it decreased from 22% to 13% of top ten required images when the order of relevance is considered in the selection. This can also be seen for the retrieval of the best sentences integrated with the conceptual graph, and best keywords integrated with the conceptual graph, where the percentage decreased from 15% to 5% and from 14% to 6% respectively.

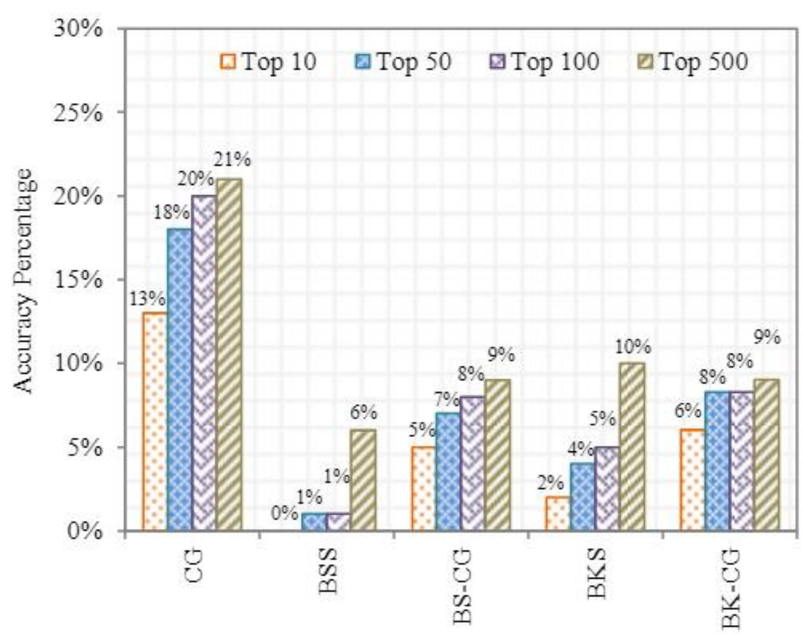


Figure 27. Order of relevance comparison between retrieval methods.

5.3 Immediate Illustration Comparisons With CG Methods

In Figure 28, only the desired illustrations that should be selected for the paragraph are considered. Our proposed approach achieved the highest percentage (6% of retrieved images) in comparison with BSS (1% of retrieved images) and BKS (2% of retrieved images). In addition, the retrieval of images is improved when the CG is integrated with the BSS and BKS. Thus, after integrating BSS with BS-CG the accuracy of retrieval for the first correct picture is improved by 3% of retrieved images, and after integrating the BKS with BK-CG it is improved by 1% of retrieved images.

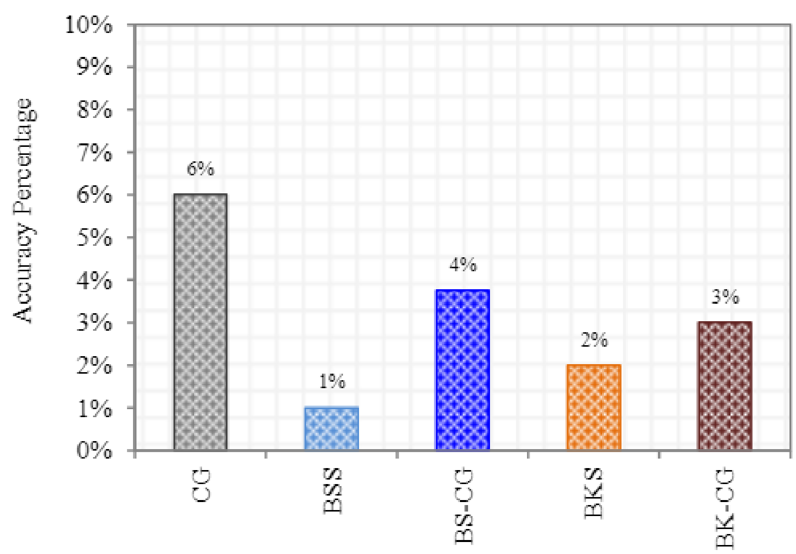


Figure 28. Immediate illustration comparison between retrieval methods.

5.4 Mining Performance Comparisons Using CG methods

In Figure 29 and Figure 30, the mining performance of the different retrieval methods is demonstrated. Results depict that processing performance is correlated with the number of words or number of characters. Thus, a text with a higher number of words or characters requires more processing time. The proposed CG method achieved better performance in comparison with other methods. Moreover, the performance has been significantly improved after the integration of the proposed approach with existing retrieval methods. The selection using BSS required the highest computational processing, but it did not give the most optimal results.

The time complexity of the developed CG algorithms is $O(n^2)$ where n is the number of times statements will be executed. The number of nodes (concepts and entities) in the knowledge base notably affects the matching between terms and their nodes in the graph. Based on the experiments conducted, we are using a total number of 90285 nodes with a hierarchical depth of 17 in Arabic WordNet; along with 1000 nodes and a hierarchical depth of 7 in our Taxonomy ontology. Therefore, in order to reduce the graph construction complexity, we are mapping matched terms with their concepts in hash tables. Thus, the time complexity of graph construction becomes $O(n \log(n))$.

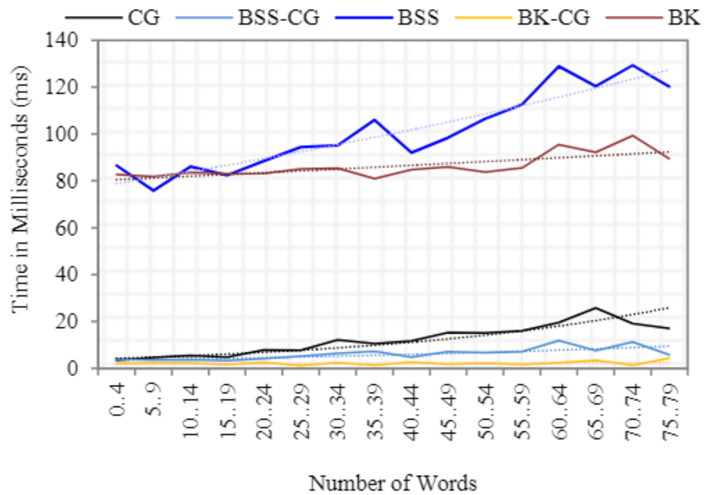


Figure 29. Mining performance of the different retrieval methods with number of words representation.

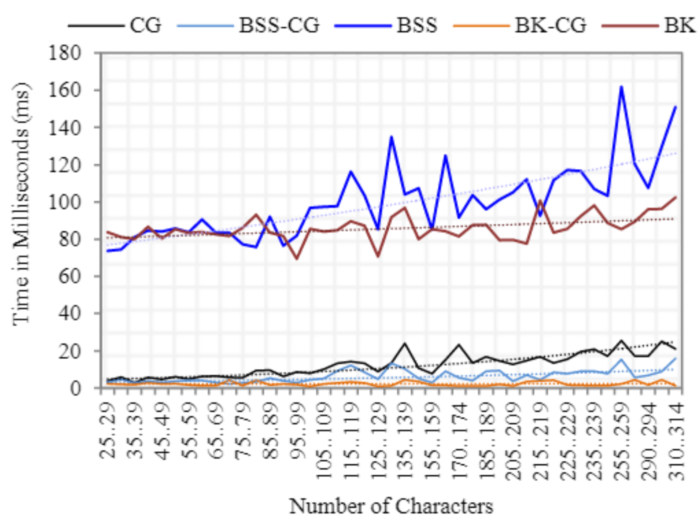


Figure 30. Mining performance of the different retrieval methods with number of characters representation.

Figure 31 summarizes the overall average of the processing performance in milliseconds (ms). Mining using CG achieved the most optimal performance (7.61 ms) in comparison with other tested retrieval methods including BSS (77.76 ms) and BK (71.34 ms). In addition, the performance significantly improved after the integration of CG with BSS and CG with BKS. The processing time after integrating BSS with CG (BSS-CG) decreased from 77.76 to 4.20 ms, and it decreased from 71.34 to 1.71 ms after integrating BKS with CG (BK-CG). This is because during the process of matching terms with the BSS and BKS methods, the distance of all possible terms must be examined with the provided text. However, the process of matching two graphs requires checking only if edges are equal.

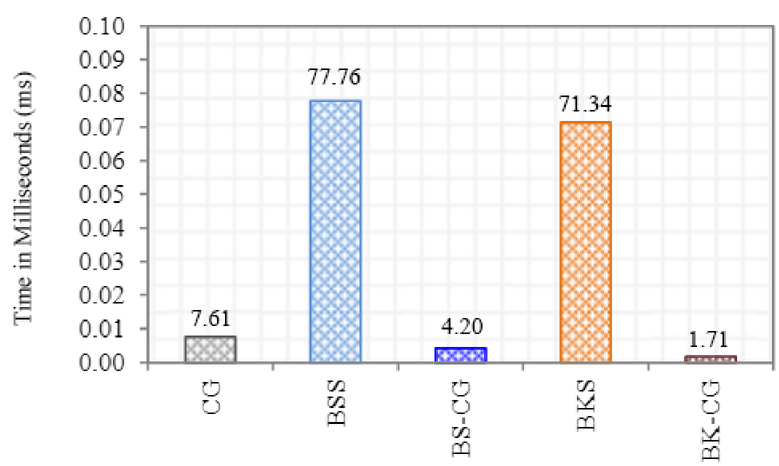


Figure 31. Average execution time of the various retrieval methods.

5.5 Retrieval Comparisons Using Semantic Weight And CG Methods

Figure 32 and Figure 33 show the obtained results of the retrieval experiments for the different methods. In this study, the order of images is not considered. Only retrieved images are considered. The method using semantic weights improved the retrieval result of relevant images. Moreover, the employment of semantic weight greatly optimized the results. For the top 10 images, after integrating the semantic weight in BSS (BSSW) the retrieval accuracy improved by 2%, and after the integration of semantic weight with BKS (BKSW) the retrieval accuracy also improved by 2%. In addition, the employment of semantic weight improved the results following its integration with BS-CG and BK-CG. Thus, for the top 10 images, after the integration of semantic weight with BS-CG (BSW-CG) the retrieval results improved by 3%, and after the integration of semantic weight with BK-CG (BKW-CG), the retrieval results improved by 12%.

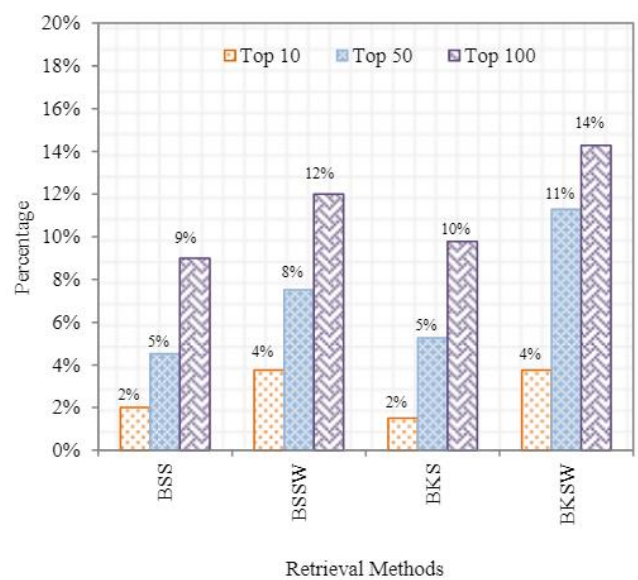


Figure 32. Comparison in the retrieval methods before and after using semantic weight.

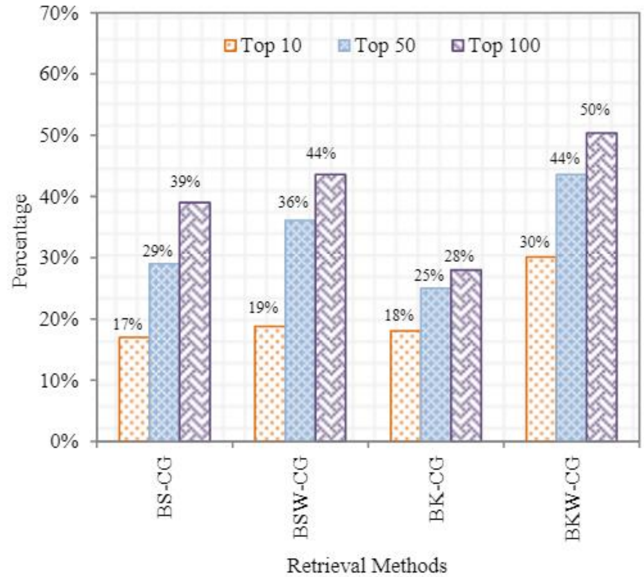


Figure 33. Comparison between the retrieval methods using conceptual graph and semantic weight.

In Table 2 the weight using TF-IDF and the semantic method are presented. The TF-IDF weight for the first sentence is 73.04 and the weight of the third sentence is 69.23. For this reason, in this example, the first sentence will be selected as the best sentence. However, after employing the semantic weighting method, the weight of the first sentence became 121.64 and the weight of the third sentence became 214.06. Therefore, the third sentence will be selected as best sentence.

Table 2: *Sample weights for different sentences*

No	Sample Sentence	TF-IDF Weight	Semantic Weight
1	Therefore, Jeha opened the door and said for the man:	73.04	121.64
2	Who are you and what do you want from me?	34.61	87.03
3	The man said, I am proud in your intelligence and knowledge, and I hope you will accept this gift from me to you.	69.23	214.06

5.6 Mining Performance Using Semantic Weight And CG Methods

In Figure 34 and Figure 35, the mining performance using the best sentence selection approach is presented. Again, the obtained results show the processing performance is correlated with the number of words or number of characters. Thus, a text with a higher number of words or characters requires more processing time.

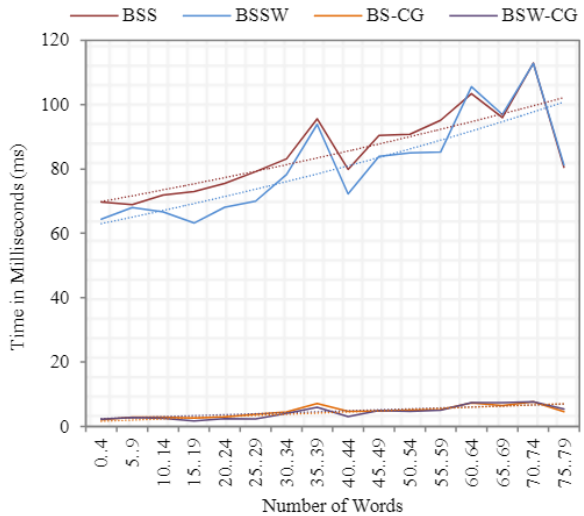


Figure 34. Mining performance of the different retrieval methods with number of words representation using best sentences selection.

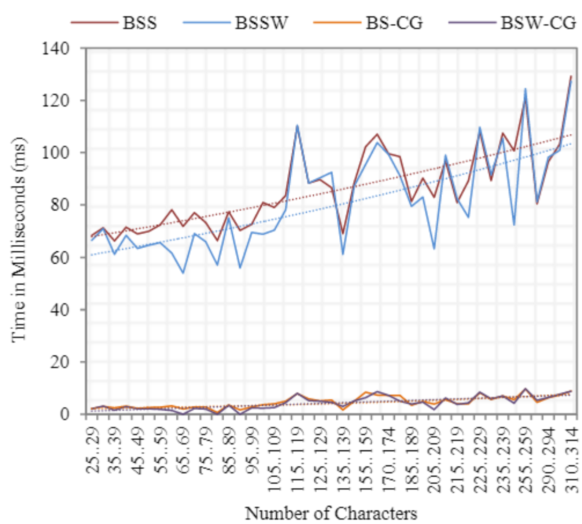


Figure 35. Mining performance of the different retrieval methods with number of characters representation using best sentence selection.

Consequently, in Figure 36 and Figure 37, the mining performance of best keywords approach is also demonstrated. In general, the processing time is roughly constant. This is because it is always the same number of best keywords being considered. In fact, the integration of the CG method improved the processing performance. The employment of semantic weight did not make considerable changes in processing the performance. The selection using BSS and BSSW needed the highest computational processing. The BSS method did not produce the most optimal results. However, the selection using BSSW improved the depiction of required illustrations by 2% in comparison with BSS.

The time complexity of the developed semantic weighting algorithms is $O(n \times m)$ where n is the number of terms (or entities) and m is the number of associated hierarchical nodes. The maximum value of m is 17 because the maximum hierarchical depth in Arabic WordNet is 17. Therefore, the complexity of the weighting algorithm becomes $O(n)$. Using the notion of $P_t(r_m)$ where P_t is the processing time and r_m is the retrieval method, we denote that $P_t(\text{BSS}) \approx P_t(\text{BSW})$, $P_t(\text{BS-CG}) \approx P_t(\text{BSW-CG})$, $P_t(\text{BKS}) \approx P_t(\text{BKSW})$, and $P_t(\text{BK-CG}) \approx P_t(\text{BKW-CG})$.

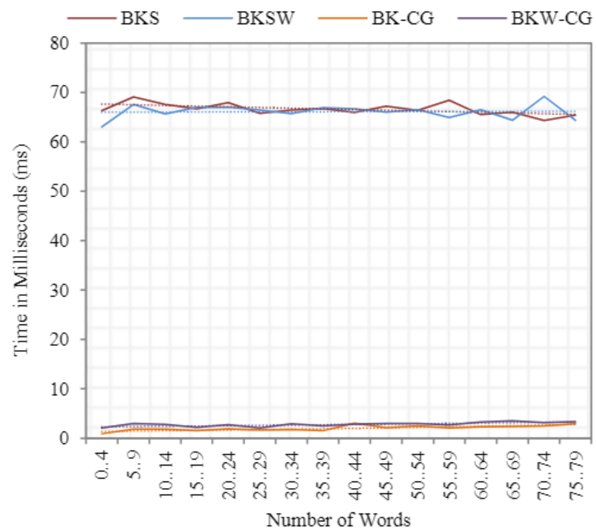


Figure 36. Mining performance of the different retrieval methods with number of words representation using best keywords selection.

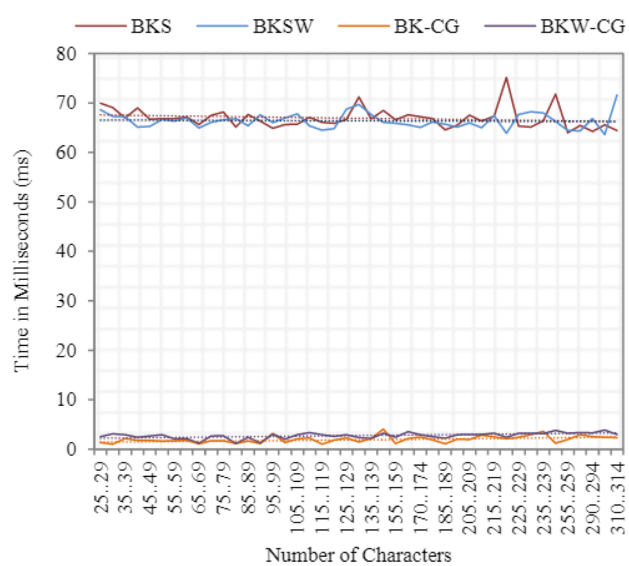


Figure 37. Mining performance of the different retrieval methods with number of characters representation using best keywords selection.

Figure 38 and Figure 39 summarize the overall average of the processing performance. In fact, the mining using CG achieved the best performance in comparison with other tested retrieval methods. After integrating the CG with BSS, BSSW, BKS and BKSW, the processing time decreased from 80.30 to 3.94 ms for BS-CG, decreased from 74.31 to 3.36 ms for BSW-CG, decreased from 66.93 to 2.33 ms for BK-CG, and decreased from 66.37 to 2.65 ms for BKS-CG. This is because the process of matching two graphs requires checking only if edges are equal. However, during the process of matching terms in other methods, the distance of all possible terms within the provided text must be examined.

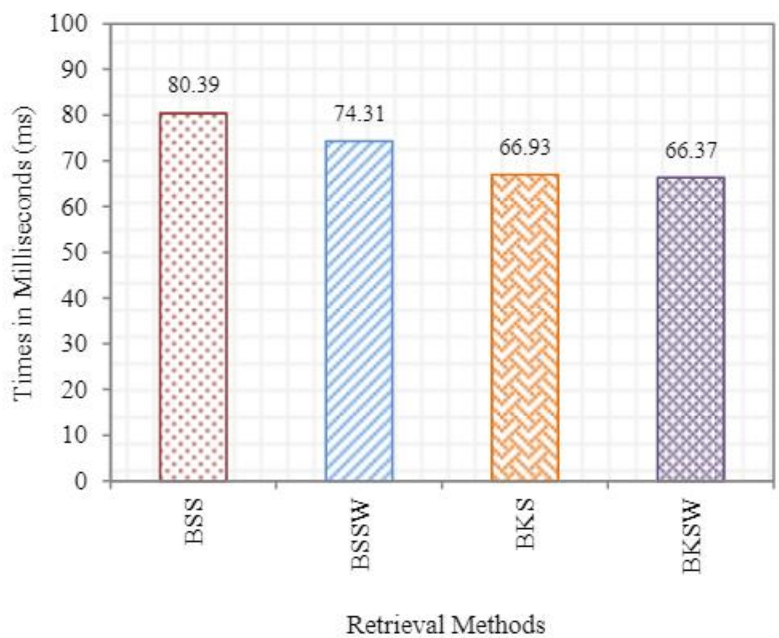


Figure 38. Average execution time of methods before and after using semantic weight.

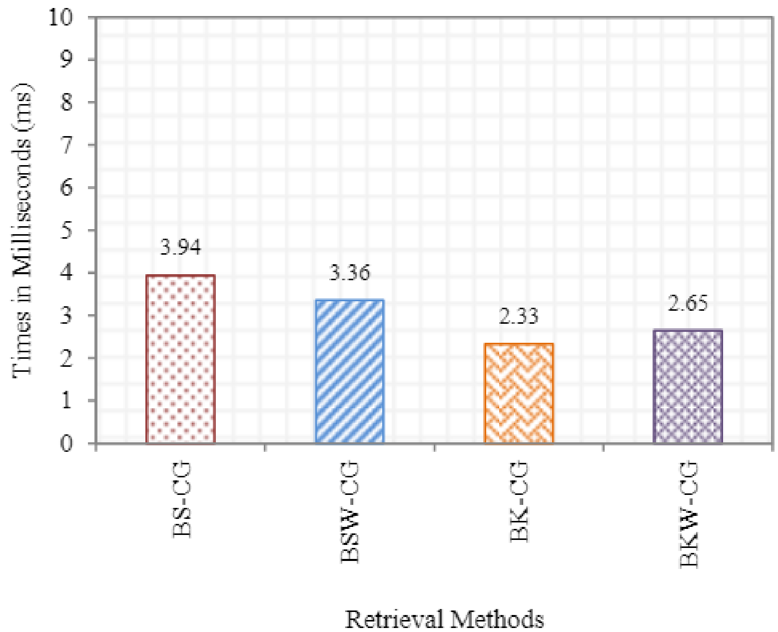


Figure 39. Average execution time of the various retrieval methods using semantic weight and conceptual graph.

5.7 Usability Assessment Comparison

We considered a usability study to assess the serviceability of our proposed system. We employed several principles of ethical considerations while making analysis with participants [97]. These principles includes: (a) privacy protection of participants and sufficient level of confidentiality; (b) full respect for the dignity of participants; (c) anonymity of participants; (d) complete consent of participants; and (e) to not harm participants in any ways. We took permission from private home instructors and parents of children to make the assessment with children. We visited schools, special-needs centers, and private tutoring centers to select instructors. We interviewed fifteen instructors teaching in different places including five elementary school instructors, five instructors for children with special needs, and five private-home teachers. We considered this diversification of selections to provide reliable usability study. We assumed that instructors that teach at home and at school, are only instructors at school. This is to distinguish the obtained results of school instructors from home instructors. We asked each instructor to fill a survey after using the system. The answer in the survey considers five points rating scale; strongly agree, agree, undecided, disagree, and strongly disagree. Table 3 demonstrates the obtained average score of all instructors.

Table 3: *Average scores of usability assessment*

Usability Statement	Average
The system is useful for education	4.47
The system was easy to use	3.93
You fairly got pictures that describe my entered text	2.87
Obtained Illustrations were clear	3.93
The conceptual graph visualization feature helped students to memorize more words	4.13
The conceptual graph feature helped students to understand more the relations between words	3.87
The conceptual graph feature assisted students to increase their ability while describing pictures	4.27
The system raised the motivation of your students	4.07
The automatic visual assessment was useful	4.73
The automatic visual assessment gave the right score for students	3.47

Moreover, we asked them to select the appropriate place where the application can be used. We asked them to choose between “Schools”, “Centers for people with special needs”, “Home”, and “All the above”. Figure 40 illustrates the obtained results.

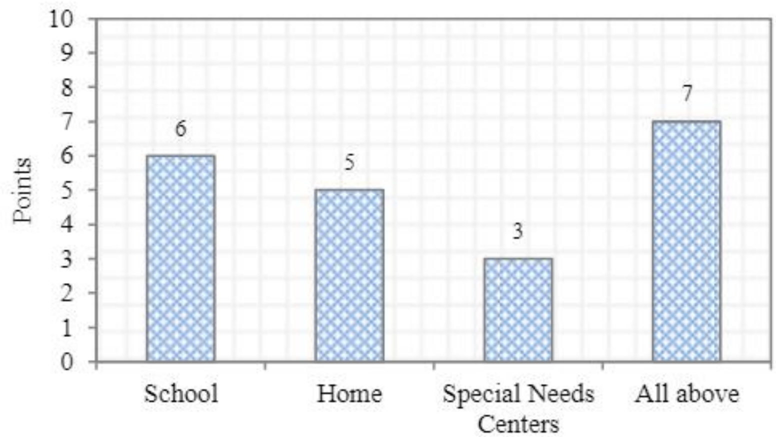


Figure 40. Appropriate 'place of use' assessment of the proposed system.

Private tutors were asked to use the system with their students. A total number of 25 students, aged 7-9 years old, were asked to describe the content of the pictures. We selected this number of children to provide assessment results with high accuracy. We considered the base line of the assessment is when reading and describing pictures without presenting their associated CGs. In the first round of a trial, the instructor will select randomly five students to read five stories that do not contain the presentation of CGs. Therefore, students must describe the content of selected pictures in natural text to obtain their grade. In the second round, the instructor will ask students to read five different stories that present CGs with their associated pictures. And then, students should write again about the content of the newly selected pictures in natural text. Figure 41 summarizes the workflow of the assessment performed with students.

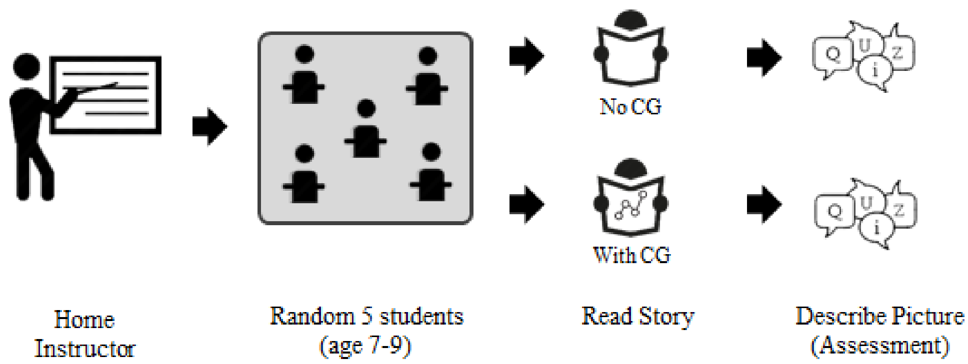


Figure 41. Assessment workflow with students

Learner-centered pedagogy [98] approach is used during the process of the experiments. We selected this approach to encourage students to use their empirical knowledge for self-learning and to increase their motivations. Each student was given 7 minutes to read the story alone and then describe the picture. Since the current version of the system does not provide automatic generation of quizzes, the instructor must select the best pictures that can be used to assess the learning of the student. Using the written text of the student, the system provides automatic grading. The automatic grading is done by matching the CG of the student input with the CG of the picture as discussed in Chapter 4. After reading stories, and viewing illustrated pictures with associated CGs, students who were seeing CGs used more words in their descriptions. We used descriptive statistic in order to analyze the obtained results. Figure 42 demonstrates the obtained results before and after using the system. Thus before using the system, the average grade of students was 6.24 ± 0.597, while after using the system the grade changed to 8.48 ± 0.770. The corresponding p-value is $p = 0.00112$, that shows the hypothesis that the proposed system will improve learning may be incorrect in only 1 in 893 cases. That shows the results of the experiment are statistically significant. In conclusion, the presentation of CGs is beneficial for learners since it facilitates their memorization of words. Some of these words were dedicatedly presented in CGs such as the type of animals, tools, and so.

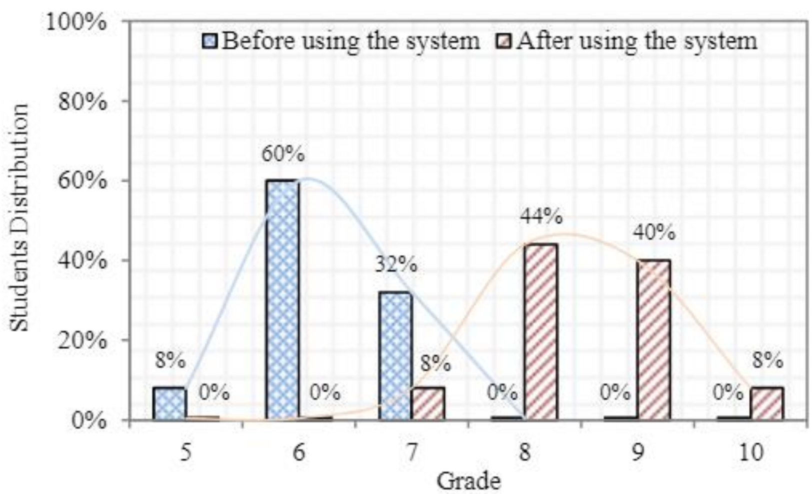


Figure 42. Improvement in student scores after using the proposed system. The overall scores of 25 students in the experiment have shifted towards higher side.

5.8 Summary

In this chapter, we presented different experiments to compare the obtained results using conceptual graph and semantic weight methods. The experiments were conducted over two approaches used in existing TTP systems including the best sentence selection and best keywords selection. In fact, the conceptual graph and semantic weight methods significantly improved the results of former approaches. Moreover, the conducted usability study demonstrated the efficiency of the proposed multimedia TTP educational system.

CHAPTER 6: CONCLUSION, CONTRIBUTIONS, AND FUTURE WORK

In this chapter, we conclude the done research in this dissertation and we list our contributions. Consequently, we present some potential future work in this area.

6.1 Conclusion

In this dissertation, we presented different multimedia TTP systems that automatically illustrate given textual content. We highlighted the limitations of some of the existing TTP systems and proposed several approaches to overcome some of these limitations. These limitations include consideration of illustration of pictures that are appropriate for required learning tasks, providing complete pedagogic information, consideration of semantic relations between words during the illustration process. We built a multimedia repository in order to solve the limitation of obtaining pedagogic illustrations. Therefore, several approaches based on the employment of conceptual graphs are explored to improve the performance of some TTP systems. Using a knowledge base, a conceptual graph is built from the text accompanied with the pictures in the multimedia repository and the text entered by the user. Consequently, the distribution of semantic weights is anticipated to consider concepts that are semantically linked hierarchically. Thus, the semantic weight retains scoring abstractive concepts. Eventually, the proposed conceptual graph and semantic weight methods are integrated with former TTP approaches based on the best sentences selection [5, 8] and best keywords selection methods [11, 15]. On the end user application level, a multimedia TTP m-learning system called "Illustrate It!" is designed and implemented. It overcomes the limitations of former TTP systems by providing illustrations for educational use. In addition, to facilitate educational needs, conceptual graph visualization and visual illustrative assessment modules are developed. The conceptual graph visualization enables learners to discover relationships between words presented in pictures. The visual illustrative assessment performs automatic evaluation of user learning by matching the semantic graph of the text accompanied with pictures and the semantic graph of the textual user input. The different conceptual graph based approaches were compared with two former approaches used in previous TTP systems including selection of the best sentence, and selection of the best keywords. In conclusion, evaluation tests demonstrate that the proposed system has significantly outperformed the best keyword selection and best sentence selection based systems. In addition, the conducted usability study demonstrates the efficiency of the proposed multimedia TTP system.

6.2 Contributions

The contributions of this research show the consequences of integrating semantic associations to illustrate textual content. The contributions can be listed as follows:

(1) A novel TTP techniques stress the usage of semantic conceptual graph to consider the semantic associations of terms to select optimal illustrations.

(2) A conceptual graph based method to distribute the score (semantic weight) of a term over its semantically associated terms. This is in order to consider concepts on the higher level of abstraction.

(3) An approach to mine pictures that considers all important terms in a paragraph instead of the selection of best sentences or best keywords.

(4) Performance improvement through the integration of the conceptual graph and semantic weight with existing mining approaches.

(5) A novel multimedia TTP educational system that can mine pedagogic illustrations for educational use.

(6) An Arabic based TTP educational system that can be used on mobile smart devices.

(7) A novel conceptual graph visualization for pictures feature.

(8) A visual illustrative assessment scoring feature.

(9) Student performance improvement while accessing the proposed visualization feature.

6.3 Future Work

Future research work in this direction may be to explore further enhancements to obtain useful illustrations without the restriction of multimedia repositories. This is because constructed repositories may not store all required illustrations. Employment of large scale knowledge bases during the process of obtaining factual information about concepts. These concepts are not necessary formally defined in local knowledge bases. Consideration of deeper knowledge extraction from natural text to improve the accuracy while constructing conceptual graphs. Usage of diverse online educational libraries to fetch automatically various types of educational illustrations. The employment of bigger number of online libraries will decrease the limitation of getting required pedagogic illustrations that are not available in the multimedia repository. Introducing diverse multimedia contents, such as animated scenes, 3D views, and Virtual Reality (VR) environments to improve the user experience and the accessibility of the proposed TTP system. Eventually, we look forward to consider the automatic adaptation of illustrated images according to the preferences of the user. These preferences include age, race, interest, and so. The automatic adaptation of displaying results will enhance the satisfaction of user.

REFERENCES

[1] R. E. Mayer and R. Moreno, "Aids to computer-based multimedia learning," *Learning and instruction,* vol. 12, pp. 107-119, 2002.

[2] R. N. Carney and J. R. Levin, "Pictorial illustrations still improve students' learning from text," *Educational psychology review,* vol. 14, pp. 5-26, 2002.

[3] W. H. Levie and R. Lentz, "Effects of text illustrations: A review of research," *Educational Communication and Technology,* vol. 30, pp. 195-232, 1982.

[4] R. E. Mayer, "Multimedia learning," *Psychology of learning and motivation,* vol. 41, pp. 85-139, 2002.

[5] X. Zhu, A. B. Goldberg, M. Eldawy, C. R. Dyer, and B. Strock, "A text-to-picture synthesis system for augmenting communication," in *Advancement of Artificial Intelligence (AAAI)*, 2007, pp. 1590-1595.

[6] C.-T. Li, C.-J. Huang, and M.-K. Shan, "Automatic generation of visual story for fairy tales with digital narrative," in *Web Intelligence*, 2015, pp. 115-122.

[7] R. Agrawal, S. Gollapudi, A. Kannan, and K. Kenthapadi, "Enriching textbooks with images," in *Proceedings of the 20th ACM international conference on Information and knowledge management*, 2011, pp. 1847-1856.

[8] A. B. Goldberg, J. Rosin, X. Zhu, and C. R. Dyer, "Toward text-to-picture synthesis," in *Neural Information Processing Systems (NIPS) 2009 Mini-Symposia on Assistive Machine Learning for People with Disabilities*, 2009, pp. 1-3.

[9] R. Mihalcea and C. W. Leong, "Toward communicating simple sentences using pictorial representations," *Machine Translation,* vol. 22, pp. 153-173, 2008.

[10] H. Li, J. Tang, G. Li, and T.-S. Chua, "Word2image: towards visual interpreting of words," in *Proceedings of the 16th ACM international conference on Multimedia*, 2008, pp. 813-816.

[11] S. A. Aramini, E. Ardizzone, and G. Mazzola, "Automatic Illustration of Short Texts via Web Images," presented at the International Conference on Information Visualization Theory and Applications (IVAPP), 2015.

[12] P. Jain, H. Darbari, and V. C. Bhavsar, "Vishit: A Visualizer for Hindi Text," in *Fourth International Conference on Communication Systems and Network Technologies (CSNT)*, 2014, pp. 886-890.

[13] D. Bui, C. Nakamura, B. E. Bray, and Q. Zeng-Treitler, "Automated illustration of patients instructions," in *American Medical Informatics Association (AMIA) Annual Symposium Proceedings*, 2012, p. 1158.

[14] N. Aletras and M. Stevenson, "Representing Topics Using Images," in *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, 2013, pp. 158-167.

[15] Y. Jiang, J. Liu, and H. Lu, "Chat with illustration," *Multimedia Systems*, vol. 22, pp. 5-16, 2016.

[16] J. Zhong, H. Zhu, J. Li, and Y. Yu, "Conceptual graph matching for semantic search," *Conceptual structures: Integration and interfaces,* pp. 92-106, 2002.

[17] A. G. Karkar and J. Ja'am, "A native Arabic eLearning mobile application to transpose Arabic text to illustrations," in *International Conference on Information and Communication Technology Research (ICTRC)*, 2015, pp. 72-75.

[18] A. G. Karkar, J. Ja'am, and S. Foufou, "A Survey on Educational Ontologies and

their Development Cycle," in *The IFIP WG5.1 12th International Conference on Product Lifecycle Management (PLM)*, 2015, pp. 649-658.

[19] C. Wen, Y. Liu, J. Gu, J. Chen, and Y. Zhang, "E-SKB: A Semantic Knowledge Base for Emergency," in *Knowledge Graph and Semantic Computing: Semantic, Knowledge, and Linked Big Data*, ed: Springer, 2016, pp. 196-202.

[20] A. Karkar, J. AlJa'am, and A. Mahmood, "Illustrate It! An Arabic Multimedia Text-to-Picture m-Learning System," *IEEE Access,* vol. 5, pp. 12777 - 12787, 2017.

[21] A. Karkar, J. M. Alja'am, M. Eid, and A. Sleptchenko, "e-Learning Mobile Application for Arabic Learners," *Journal of Educational & Instructional Studies in the World,* vol. 5, 2015.

[22] A. Karkar, J. M. Al Ja'am, S. Foufou, and A. Sleptchenko, "An e-learning mobile system to generate illustrations for Arabic text," in *Global Engineering Education Conference (EDUCON)*, 2016, pp. 184-191.

[23] A. Karkar and J. M. Al Ja'am, "An Arabic mobile educational system," in *12th Learning and Technology Conference*, 2015, pp. 20-25.

[24] A. G. Karkar, A. Ja'am, and J. Mohamad, "An Educational Ontology-based M-Learning System," *International Journal of Interactive Mobile Technologies,* vol. 10, 2016.

[25] D. J. Schultz, "IEEE standard for developing software life cycle processes," *IEEE Std,* pp. 1074-1997, 1997.

[26] S. N. Dessai and R. Dhanaraj, "A Different Approach to Text to 3D Generation," *Artificial Intelligent Systems and Machine Learning,* vol. 8, pp. 104-107, 2016.

[27] D. Ustalov and R. Kudryavtsev, "An Ontology Based Approach to Text to Picture Synthesis Systems," in *Proceedings of the 2nd International Workshop on Concept Discovery in Unstructured Data (CDUD 2012).* 2012, pp. 94-101.

[28] F. M. Wahl, K. Y. Wong, and R. G. Casey, "Block segmentation and text extraction in mixed text/image documents," *Computer graphics and image processing,* vol. 20, pp. 375-390, 1982.

[29] S. Kumar, R. Gupta, N. Khanna, S. Chaudhury, and S. D. Joshi, "Text extraction and document image segmentation using matched wavelets and MRF model," *IEEE Transactions on Image Processing,* vol. 16, pp. 2117-2128, 2007.

[30] S. Mori, H. Nishida, and H. Yamada, *Optical character recognition*. John Wiley & Sons, Inc., 1999.

[31] M. A. Hasnat, M. R. Chowdhury, and M. Khan, "An open source tesseract based optical character recognizer for bangla script," in *10th International Conference on Document Analysis and Recognition (ICDAR'09)*, 2009, pp. 671-675.

[32] H. Noh, Y. Jo, and S. Lee, "Keyword selection and processing strategy for applying text mining to patent analysis," *Expert Systems with Applications,* vol. 42, pp. 4348-4360, 2015.

[33] C. Baral, G. Gonzalez, and T. Son, "Conceptual modeling and querying in multimedia databases," *Multimedia Tools and Applications,* vol. 7, pp. 37-66, 1998.

[34] S. Robertson, "Understanding inverse document frequency: on theoretical arguments for IDF," *Journal of documentation,* vol. 60, pp. 503-520, 2004.

[35] J. Goldstein, M. Kantrowitz, V. Mittal, and J. Carbonell, "Summarizing text

documents: sentence selection and evaluation metrics," in *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, 1999, pp. 121-128.

[36] L. I. Cheng, A. Basu, and R. Goebel, "Interactive Multimedia for Adaptive Online Education," *IEEE MultiMedia,* vol. 16, pp. 16-25, 2009.

[37] W. Cohen, P. Ravikumar, and S. Fienberg, "A comparison of string metrics for matching names and records," in *Kdd workshop on data cleaning and object consolidation*, 2003, pp. 73-78.

[38] P. N. Pirnay-Dummer, "Ontology and semantic web," in *Encyclopedia of the sciences of learning*, ed: Springer, 2012, pp. 2507-2510.

[39] T. Berners-Lee, J. Hendler, and O. Lassila, "The semantic web," *Scientific american,* vol. 284, pp. 28-37, 2001.

[40] N. F. NOY, "Ontology Development 101: A Guide to Creating Your First Ontology: Knowldege Systems Laboratory, Stanford University," *Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880,* 2001.

[41] *Semantic Web Ontology*, 2003. Online. Available: http://wiki.dbpedia.org/services-resources/interlinking. Accessed: June 2012.

[42] S. Bechhofer, "OWL: Web ontology language," in *Encyclopedia of Database Systems*, ed: Springer, 2009, pp. 2008-2009.

[43] A. Mateescu and A. Salomaa, "Formal languages: an introduction and a synopsis," *Handbook of formal languages,* vol. 1, pp. 1-40, 1997.

[44] S. BuvaE and I. A. Mason, "Propositional logic of context," in *Proceedings of the*

eleventh national conference on artificial intelligence, 1993.

[45] A. De Morgan, *Formal logic: the calculus of inference, necessary and probable*. Taylor and Walton, 1847.

[46] A. Pnueli, "The temporal logic of programs," in *18th Annual Symposium on Foundations of Computer Science*, 1977, pp. 46-57.

[47] J. Cheng, "Strong relevant logic as the universal basis of various applied logics for knowledge representation and reasoning," *Frontiers In Artificial Intelligence and Applications,* vol. 136, p. 310, 2006.

[48] U. Bojars and J. Breslin, "Semantically-interlinked online communities ontology," *National University of Ireland (NUI) Galway Std*, 2010.

[49] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives, *Dbpedia: A nucleus for a web of open data*. Springer, 2007.

[50] G. A. Miller, "WordNet: a lexical database for English," *Communications of the ACM,* vol. 38, pp. 39-41, 1995.

[51] S. Elkateb, W. Black, H. Rodríguez, M. Alkhalifa, P. Vossen, A. Pease, and C. Fellbaum, "Building a wordnet for arabic," in *Proceedings of The fifth international conference on Language Resources and Evaluation (LREC 2006)*, 2006.

[52] H. Ishkewy, H. Harb, and H. Farahat, "Azhary: An Arabic Lexical Ontology," *arXiv preprint arXiv:1411.1999,* 2014.

[53] C. Bizer, T. Heath, K. Idehen, and T. Berners-Lee, "Linked data on the web (LDOW2008)," in *Proceedings of the 17th international conference on World Wide Web*, 2008, pp. 1265-1266.

[54] J. Hanna, E. Joseph, M. Brochhausen, and W. R. Hogan, "Building a drug ontology based on RxNorm and other sources," *J. Biomedical Semantics*, vol. 4, p. 44, 2013.

[55] C. Bizer, R. Cyganiak, and T. Gauß, "The RDF book mashup: from web APIs to a web of data," in *Proceedings*, 2007, pp. 1-7.

[56] V. De Boer, J. Wielemaker, J. Van Gent, M. Hildebrand, A. Isaac, J. Van Ossenbruggen, and G. Schreiber, "Supporting linked data production for cultural heritage institutes: the amsterdam museum case study," in *The Semantic Web: Research and Applications*, ed: Springer, 2012, pp. 733-747.

[57] G. O. Consortium, "The gene ontology project in 2008," *Nucleic acids research*, vol. 36, pp. D440-D444, 2008.

[58] *Dbpedia Interlinking*, 2007. Online. Available: http://wiki.dbpedia.org/services-resources/interlinking. Accessed: April 19, 2016.

[59] W. C. S. W. Group, "SPARQL 1.1 Overview," *W3C Recommendation. W3C*, 2013.

[60] M. Chein and M.-L. Mugnier, "Conceptual graphs are also graphs," in *International Conference on Conceptual Structures*, 2014, pp. 1-18.

[61] B. D. McKay and A. Piperno, "Practical graph isomorphism, II," *Journal of Symbolic Computation*, vol. 60, pp. 94-112, 2014.

[62] M. A. Musen, "Protégé ontology editor," *Encyclopedia of Systems Biology*, pp. 1763-1765, 2013.

[63] J. Guerson, T. P. Sales, G. Guizzardi, and J. P. A. Almeida, "OntoUML Lightweight Editor: A Model-Based Environment to Build, Evaluate and

Implement Reference Ontologies," in *IEEE 19th International on Enterprise Distributed Object Computing Workshop (EDOCW),*, 2015, pp. 144-147.

[64] *MarkLogic*, 2015. Online. Available: http://www.marklogic.com/. Accessed: May 2017.

[65] M. Horridge and S. Bechhofer, "The owl api: A java api for owl ontologies," *Semantic Web*, vol. 2, pp. 11-21, 2011.

[66] J. J. Carroll, I. Dickinson, C. Dollin, D. Reynolds, A. Seaborne, and K. Wilkinson, "Jena: implementing the semantic web recommendations," in *Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters*, 2004, pp. 74-83.

[67] B. Glimm, I. Horrocks, B. Motik, G. Stoilos, and Z. Wang, "HermiT: an OWL 2 reasoner," *Journal of Automated Reasoning,* vol. 53, pp. 245-269, 2014.

[68] G. Guizzardi, *Ontological foundations for structural conceptual models*. Centre for Telematics and Information Technology (CTIT), 2005.

[69] *OWL 2 Web Ontology Language Document Overview (Second Edition)*, 2012. Online. Available: https://www.w3.org/TR/owl2-overview/. Accessed: December 2012.

[70] D. Tsarkov and I. Horrocks, "FaCT++ description logic reasoner: System description," *Automated reasoning,* pp. 292-297, 2006.

[71] *SmartLogic*, 2015. Online. Available: http://www.smartlogic.com/. Accessed: March 2017.

[72] *Synaptica*, 2015. Online. Available: http://www.synaptica.com/. Accessed: February 2017.

[73] *Linked Data*, *"Linked data-connect distributed data across the web"*, 2010. Online. Available: http://linkeddata.org/. Accessed: February 2017.

[74] T. Kleinberger, L. Schrepfer, A. Holzinger, and P. Müller, "A multimedia repository for online educational content," 2001.

[75] T. Kleinberger and P. Müller, "Content management for web based education," 2000.

[76] A. Borman, R. Mihalcea, and P. Tarau, "PicNet: Augmenting Semantic Resources with Pictorial Representations," in *Advancement of Artificial Intelligence (AAAI) Spring Symposium: Knowledge Collection from Volunteer Contributors*, 2005, pp. 1-7.

[77] D. Ustalov, "A text-to-picture system for russian language," in *Proceedings of the Sixth Russian Young Scientists Conference in Information Retrieval, (RuSSIR'2012)*, 2012.

[78] K.-P. Yee, K. Swearingen, K. Li, and M. Hearst, "Faceted metadata for image search and browsing," in *Proceedings of the SIGCHI conference on Human factors in computing systems*, 2003, pp. 401-408.

[79] P. D. Turney, "Learning to extract keyphrases from text," *arXiv preprint cs/0212013,* 2002.

[80] R. Mihalcea and P. Tarau, "TextRank: Bringing order into texts," in *Proceedings of the 2004 conference on empirical methods in natural language processing*, 2004, pp. 1-8.

[81] E. Brill, "A simple rule-based part of speech tagger," in *Proceedings of the workshop on Speech and Natural Language*, 1992, pp. 112-116.

[82] T. Marchant, "Valued relations aggregation with the Borda method," *Journal of Multi‐Criteria Decision Analysis,* vol. 5, pp. 127-132, 1996.

[83] A. C. Browne, G. Divita, A. R. Aronson, and A. T. McCray, "Umls language and vocabulary tools: Amia 2003 open source expo," in *American Medical Informatics Association (AMIA) Annual Symposium Proceedings*, 2003, p. 798.

[84] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision,* vol. 60, pp. 91-110, 2004.

[85] P. G. Ipeirotis, "Analyzing the amazon mechanical turk marketplace," *XRDS: Crossroads, The ACM Magazine for Students,* vol. 17, pp. 16-21, 2010.

[86] A. Abbache, F. Meziane, G. Belalem, and F. Z. Belkredim, "Arabic query expansion using wordnet and association rules," *International Journal of Intelligent Information Technologies,* vol. 12, pp. 51-64, 2016.

[87] A. C. Erdal, *Binary tree structures for matrix multiplication*. Michigan State University, 1983.

[88] A. Karkar, A. Dandashi, and J. M. Al Ja'am, "A Text-to-Picture m-Learning System in a Private Wireless Mesh Network," in *19th Conference on Innovations in Clouds, Internet and Networks*, 2016, pp. 212-218.

[89] C. Zhang, L. Zhang, C.-J. Wang, and J.-Y. Xie, "Text Summarization Based on Sentence Selection with Semantic Representation," in *2014 IEEE 26th International Conference on Tools with Artificial Intelligence (ICTAI)*, 2014, pp. 584-590.

[90] W. Guo, "Flexible Selection of Output Format for Sets in Java Collections:

Algorithms and Their Complexity and Reusability," *WSEAS TRANSACTIONS ON MATHEMATICS*, vol. 6, p. 309, 2007.

[91] R. S. Pressman, *Software engineering: a practitioner's approach*. Palgrave Macmillan, 2005.

[92] K. Arnold, J. Gosling, and D. Holmes, *The Java programming language*. Addison Wesley Professional, 2005.

[93] B. Snyder, D. Bosnanac, and R. Davies, *ActiveMQ in action*. Manning Publications, 2011.

[94] I. Warren, A. Meads, S. Srirama, T. Weerasinghe, and C. Paniagua, "Push notification mechanisms for pervasive smartphone applications," *IEEE Pervasive Computing*, vol. 13, pp. 61-71, 2014.

[95] K. M. Bell, D. N. Bleau, and J. T. Davey, "Push notification service," ed: Google Patents, 2011.

[96] A. Nathan, *Windows Presentation Foundation Unleashed (WPF)*. Sams, 2006.

[97] A. Bryman and E. Bell, *Business research methods*. Oxford University Press, USA, 2015.

[98] S. Dole, L. Bloom, and K. Kowalske, "Transforming pedagogy: Changing perspectives from teacher-centered to learner-centered," *Interdisciplinary Journal of Problem-Based Learning*, vol. 10, pp. 1-14, 2016.

APPENDICES

Appendix A: Questionnaire

Illustrate It! A Text-to-Picture multimedia learning system

Questionnaire

Dear respected instructor,

We are pleased to ask you to fill this questionnaire based on your experience with the system. The questionnaire focuses on pros and cons of our proposed system. It is required to further develop and improve the quality we are providing. Your opinion is very valuable for us. We do highly appreciate your honesty in providing accurate answers.

<u>Section I: General Information</u>

Please, answer the following questions:

Full Name (optional):			
Country of Origin:			
Age			
Gender	☐ Male	☐ Female	
Academic Level	☐ Preparatory	☐ Secondary	☐ University
Teaching Center	☐ School	☐ Home	☐ Special Needs
Teaching Center Name			
Computer Skills	☐ Good	☐ Moderate	☐ Low

Section II. Appropriate places to use the system

What is the best place to use the System? and Why?

☐ School ☐ Home ☐ Special Needs Centers ☐ All above

Please explain:

Section III. Usability of the system

No.	Statement	Strongly Agree	Agree	Undecided	Disagree	Strongly Disagree
1	The system was easy to use.					
2	You fairly got pictures that describe the entered text.					
3	Illustrations were clear					
4	The conceptual graph helped students to memorize more words					
5	The conceptual graph feature helped students to understand the relations between words					
6	The conceptual graph feature assisted students to increase their ability to describe pictures					

No.	Statement	Strongly Agree	Agree	Undecided	Disagree	Strongly Disagree
7	The system raised the motivation of your student(s)					
8	The automatic visual assessment was useful.					
9	The automatic visual assessment gave the right score for students.					

Section III. Potential features.

No.	Statement	Strongly Agree	Agree	Undecided	Disagree	Strongly Disagree
1	It is good to use the system to read and follow with your students					
2	It is good to use the system as part of the educational material					
3	It is good to see how much students are focused while studying					

Please, write your own futures that should be considered as part or improvements to the proposed educational system.

Appendix B: Dissertation Publications

Journal Papers:

1. **A. Karkar**, and J. M. AlJa'am, "A Semantic Based Text-to-Picture System", (under preparation for journal submission)
2. **A. Karkar**, and J. M. AlJa'am, A. Mahmood, "Illustrate It! An Arabic Multimedia Text-to-Picture m-Learning System," *IEEE Access*, vol. 5, no. 1, pp. 12777-12787, 2017. **(ISI, IF 3.2)**
3. **A. Karkar**, and J. M. AlJa'am, "An Educational Ontology based m-Learning System," International Journal of Interactive Mobile Technologies, vol. 10, no. 4, pp. 48-56, 2016.
4. **A. Karkar**, J.M. JA'AM, M. Eid, and A. Sleptchenko, "e-Learning Mobile Application for Arabic Learners," Journal of Educational Instructional Studies in the World, vol. 5, no. 2, pp. 45-54, 2015.

Conference Papers:

5. **A. Karkar**, A. Dandashi, and J. M. JA'AM, "A Text-to-Picture m-Learning System in a Private Wireless Mesh Network," *Innovations in Clouds, Internet and Networks*, 2016, pp. 212-218.

6. **A. Karkar**, J. M. JA’AM, S. Foufou, and A. Sleptchenko, "An e-Learning Mobile System to Generate Illustrations for Arabic Text," *Global Engineering Education Conference*, 2016, pp.184-191.

7. **A. Karkar**, J. M. Ja’am, and S. Foufou, "An Arabic Text-to-Picture Mobile Learning System," *Qatar Foundation Annual Research Conference*, 2016, no. 1, p. ICTPP3331.

8. **A. Karkar**, J. M. JA’AM, and S. Foufou, "A Survey on Educational Ontologies and their Development Cycle," *The IFIP WG5.1 12th International Conference on Product Lifecycle Management*, 2015, pp. 649-658.

9. **A. Karkar**, and J. M. JA’AM, "A Native Arabic eLearning Mobile application to Transpose Arabic Text to Illustrations," *International Conference on Information and Communication Technology Research*, 2015, pp. 69-72.

10. **A. Karkar**, and J. M. JA’AM, "An Arabic Mobile Educational System," *The 12th Learning and Technology International Conference*, 2015, pp. 20-25.

11. **A. Karkar**, J. M. JA’AM, and S. Foufou, "Illustration Generation based on Arabic Ontology for Children," *Arab Computing Society International Conference on Computer Systems and Applications*, 2014.

Appendix C: Other Related Publications

Journal Papers:

1. A. Dandashi, **A. Karkar**, S. Saad, Z. Barhoumi, J. Al-Jaam, and A. El Seddik, "Enhancing the Cognitive and Learning Skills of Children with Intellectual Disability through Physical Activity and Edutainment Games," *International Journal of Distributed Sensor Networks*, SAGE, vol. 2015, 2015. **(ISI, IF 1.23)**
2. S. El-Seoud, **A. Karkar**, J.M. JA'AM, and O. Karam, "A Pictorial Mobile Application for Improving Communication Skills in Non-Verbal Autism," *International Journal of Interactive Mobile Technologies*, vol. 9, no. 4, pp. 49-55, 2015.
3. S. El-Seoud, **A. Karkar**, A. Dandashi, J.M. JA'AM, and I. Taj-Eddin, "Using Handled Mobile System to Address Illiteracy," *International Journal of Computer Science and Information Security*, vol. 13, no. 6, pp. 77-84, 2015.

Conference Papers:

4. A. Dandashi, **A. Karkar**, and J.M. JA'AM, "Cognitive Technology for Children with Hearing," *Global Engineering Education Conference*, 2016, pp. 660-667.
5. S. El-Seoud, H. El-Sofany, **A. Karkar**, A. Dandashi, I. Taj-Eddin, and J.M. JA'AM, "Semantic-Web Automated Course Management and Evaluation System using Mobile Applications," *Interactive Collaborative Learning*, 2015, pp. 271-282.

6. A. Dandashi, **A. Karkar**, and J.M. JA’AM, S. El-Seoud, "Framework for Development of Cognitive Technology for Children with Hearing Impairments", International *Conference on Interactive Collaborative Learning*, 2015, pp. 515-522.

7. **A. Karkar**, M. S. Saleh, S. Saad, and J. M. JA’AM, "An Arabic Ontology-based Learning System for Children with Intellectual Challenges," *Global Engineering Education Conference*, 2014, pp. 670-675.

8. **A. Karkar**, A. Dandashi, J. M. JA’AM, and S. Foufou, "Illustration Generation based on Arabic Ontology for Children with HI," *Qatar Foundation Annual Research Conference*, 2014, p. ITPP0337.

9. A. Dandashi, **A. Karkar**, and J.M. JA’AM, "Arabic Natural Language Processing: Framework for Translative Technology for Children with Hearing Impairments," *Qatar Foundation Annual Research Conference*, no. 1, p. ITPP0396, 2014.