Arabic Anaphora Resolution: Corpus of the Holy Qur'an Annotated with Anaphoric Information

Khadiga M. Seddik
Faculty of computers and information
Cairo University
Giza, Egypt

Ali Farghaly
Computational Linguistics
Software Researcher
Networked Insights, Chicago, USA

Aly Aly Fahmy
Faculty of computers and information
Cairo University
Giza, Egypt

ABSTRACT
This paper reports on compiling a large Arabic corpus of the Holy Qur'an script, annotated with anaphoric relation and other anaphoric information, providing multi-dimensional feature vector rich with most of basic anaphoric information needed in statistical anaphora resolution systems. About 24,653 personal pronouns are tagged with their antecedents and other anaphoric information like distance between the anaphor and its antecedent in terms of verses, words, and segments, gender, number, person, and other information which can be used to implement the feature vector of a statistical anaphora resolution system. In addition, it describes the compilation of a bank of sentence patterns consisting of 481 antecedent patterns; each pattern represents particular part-of-speech tag corresponding to its antecedent phrase. The aim is to provide a valuable resource that enables future research in Arabic anaphora resolution, and help in future work in analyzing Quran script. Also, it will be a valuable resource that can be used for training and testing anaphora resolution systems, and evaluating.

General Terms
Natural language processing, Computational linguistics, Anaphora resolution, Corpus development.

Keywords
Anaphora resolution, Arabic language, Corpus, Quran, Pronominal anaphora.

1. INTRODUCTION
Anaphora resolution is one of the challenging tasks of natural language processing. It is specifically concerned with matching up particular entities or pronouns with the nouns or names that they refer to. It is very important since without it a text would not be fully and correctly understood, and without finding the proper antecedent, the meaning and the role of the anaphor cannot be realized.

There are various types of anaphora such as pronominal, verb, comparative, and lexical anaphora. Most widespread types in the Arabic Computational Linguistics Literature are pronominal anaphora, which is realized by anaphoric pronouns [1]. Pronouns form a special class of anaphors because of their empty semantic structure [2], in addition, not all pronouns are anaphoric (e.g. deictic pronouns such as ("أنا" I, "أنت" you, "نحن" we).

Information about anaphoric relations has a significant role in applications such as automatic summarization, machine translation, Opinion mining, Question-Answering, Named Entity Recognition, and others.

Most current machine translation systems dealing with Arabic texts face problems because of the lack of reliable Anaphora Resolution [3]. The main reason for Anaphora resolution errors made by such systems are due to the difference between Arabic and English pronominal systems in syntax, morphology and semantic. The major problem in anaphora resolution is the lack of corpora annotated with anaphoric relations especially for Arabic languages although it is very much needed in most NLP systems. The annotation task of anaphoric relations is very time consuming and requires a significant effort from the human annotator. Arabic annotated resource is much needed to encourage works on Arabic anaphora resolution.

The motivation behind this work is to produce an Arabic corpus annotated with Anaphoric information, which is a valuable resource that enables future research in Arabic anaphora resolution, and help in future work in analyzing Quran script. Also, it will be available resource used for training and testing anaphora resolution systems, and evaluating and optimizing existing approaches of anaphora resolution. My own motivation for choosing Arabic language is that it is a less studied language and work on Arabic anaphora resolution has been very little. Providing a representative corpus will benefit researchers, encourage work in Arabic anaphora resolution, and help in extracting empirical patterns and rules for either building new anaphora resolution approaches or other Arabic NLP applications. Also the provided multi-dimensional feature vector will provide anaphoric information needed to build statistical anaphora resolution systems.

This paper proposes a large Arabic corpus consisting of Holy Qur'an script, annotated with anaphoric relations and other anaphoric information. About 24,653 personal pronouns are tagged with their antecedents and other anaphoric information like distance between the anaphor and its antecedent in terms of verses, words, and segments, gender, number, person, and other information which can be used to implement the feature vector of a statistical anaphora resolution system. In addition, it provides a bank of sentence patterns consisting of 481 part of speech patterns; each pattern represents particular part of speech corresponding to its antecedent phrase to help in the noun-phrase extraction task. The annotation scheme, annotation process, and the final output corpora are described in details.

This paper is structured in 7 sections. In section 2, related works to Arabic anaphora resolution are presented. Section 3 presents a brief overview of available annotated corpora and existing annotation schemes. Section 4 presents the proposed methodology in details. Section 5 discusses results and statistical studies. Section 6 presents the barriers and

challenges faced in Arabic anaphora resolution task. The last section gives a conclusion and future work. The next subsection explains why the Holy Quran text is chosen for this task.

1.1 Why the Holy Quran?

The holy Quran is 1,400 years old and it is the most important religious text of Islam [4]. The Quranic scripture is used and cited by 1.5 billion Arab and non-Arab Muslims all over the world, and it is claimed by [5] to be "the most widely read book in the whole world".

The Holy Quran is organized into 114 chapters. Each chapter (sura in Arabic) has a unique name and number, and it is divided into a sequence of verses (ayat in Arabic). These verses contain the actual words used in the Quran. The total number of verses of all Quran chapters is 6236 verses. Nearly all chapters in the Quran precede their verses with the phrase 'bismillah'.

The main reasons for choosing Quran are: 1. Quran is characterized by a large number of pronouns (over 24,500 pronouns) and very frequent use of anaphors, especially pronominal anaphors. 2. The Quran script is a restricted text with manageable size (127,795 word segments), which is counted as an advantage for the manual annotation task. 3. Resolving the pronouns in Quran is very valuable and useful in understanding the Quranic scripture, Also it will be very helpful in translation process as the Quranic scripture is widely translated into almost all languages in the world [6]. 4. In Quran, anaphora can be span over many verses such as the pronouns in Surah Al-Baqarah (The Cow chapter), there are pronouns in verse 80 that refer to 'Child of Israel' in verse 47 (33 verses distance). 5. Quran is considered to be the highest form of classical Arabic text, Nevertheless its rules and patterns can be applied to Modern Standard Arabic (MSA) and local dialect.

2. RELATED WORKS TO ARABIC ANAPHORA RESOLUTION

There is two major researches work on Arabic anaphora resolution done by [7], and [3]. This section will give a brief description of each one of them.

2.1 Multilingual Robust Anaphora Resolution Approach [7]

To the best of authors knowledge, [7] is the first research work on Arabic anaphora resolution. The approach used in this work consists of a modification of Mitkov's approach [8], and operates on texts pre-processed by a part-of speech tagger.

The system checks input against the number and agreement of the antecedent indicator. A score is set to each candidate according to each indicator, and the candidate with the highest score is returned as the winning antecedent. The approach was initially developed and tested for English, but it has been subjected to some modifications to work with Polish and Arabic. The approach resolves pronoun anaphora coreference without using linguistic or domain knowledge, or even parsing. Instead, it makes use of corpus-based NLP techniques. The robust approach used without any modification scored a success rate of 90.5%, whereas the improved Arabic version scored 95.2%.

However the evaluation of the adapted version for Arabic reported satisfactory results, the evaluation was based on sample text from Arabic technical manual (only 63 examples); also the sample text belongs to a restricted domain.

2.2 Arabic Anaphora Resolution Using the Web as Corpus [3]

The author builds a dynamic statistical anaphora resolution algorithm in Arabic unrestricted text, which uses least possible feature and less human intervention in order to overcome the problem of the absence of enough NLP resources. It made use of the web as a corpus.

The algorithm sets a search scope of -20 window size. The algorithm makes use of collocational evidence, recency and bands features in finding the appropriate antecedent. Collocational evidence depends on finding the collocational relation between candidate antecedents and the pronoun's carrier using conditional probability as the association measure. Recency feature gives more weight to the closer candidate to the pronoun and its carrier than the farther ones. Band feature is used to reduce the search space and in turn, reduce the number of candidate antecedents by dividing the (-20) words into two word bands and the band with the highest score is chosen. The band with the highest score is further divided into smaller bands again. The algorithm reported a precision rate of 78%, F-measured performance of 87.6%, and recall rate of 100%, all measured according to a gold standard set of 5000 pronouns.

3. PREVIOUS WORK ON ANNOTATED CORPORA AND ANNOTATION SCHEME

3.1 Available Annotated Corpora

3.1.1 Arabic Corpora Annotation with Coreferential Links [1]

This work provides an Arabic corpora annotated with anaphoric relation, and in particular, it proposes a tool (AnATAr) designed for annotating Arabic corpus by automatically detecting Arabic pronoun and allows the human annotator to select several anaphoric pronouns related to the same antecedent.

The author mentioned that - to his knowledge- there is no available resources for Arabic before his work. This was his motivation to build his anaphoric annotating tool for Arabic which eases building such resource either by computer scientists or linguists.

The corpus composed of articles in different fields drawn from Arabic newspapers, technical manuals of computer, Tunisian educational book, and a novel. The corpus size was 77,457 words.

The annotation scheme used in this work is XML-based scheme proposed by [9], which is compatible with the MUC annotation scheme with many advantages for Arabic language. The author decided to annotate the identity relation between the anaphors (pronouns, definite descriptions or proper names) and their antecedents (noun phrases), and excluded the demonstrative pronouns.

The developed tool is used to annotate the corpus. The input is encoded with XML and segmented into paragraphs and sentences. The human annotator selects the anaphor manually by using the mouse, and then selects the antecedent. After that, the program adds the tags for both anaphor and its antecedent.

Compared to the proposed corpus, although this corpus is very close to the size of the proposed corpus (77,457 words), the number of tagged pronouns in the proposed corpus are much higher (24,653 vs. 4300). Also this corpus deals with Arabic words, however in Arabic Language, the word consists of root, affixes, and clitics, so it's better to deals with word segments instead of word (like in the proposed corpus).

3.1.2 QurAna Corpus [6]

A large corpus created from the original Quranic text, where personal pronouns are tagged with their antecedents. Over 24,500 pronouns are tagged with their antecedent information. Also the antecedents are maintained as an ontological list of concepts.

The authors mentioned that this corpus is the first of its kind covering Classical Arabic text. QurAna corpus contains 128,000 word segments and 24,679 pronouns. Quranic Arabic Corpus (QAC) [10] is used in counting pronouns in the Quran. QAC is a project where every word of the Quran is tagged with morphological, POS, and syntactic information.

The annotation process is guided by other annotation schemes like [11], [12], and [2]. Also the nature of Arabic Language, Arabic usage of pronouns, and the nature of Quran as a particular domain are considered in the annotation process.

Each Quranic Surah is presented in separate xml file with unique ID. The annotation covers all kinds of personal pronouns: 1st, 2nd, and 3rd person pronouns; singular, dual, and plural; connected and separate stand-alone pronouns. Relative and demonstrative pronouns are left as they are less in number (approx 15%) according to the author's statistics, and often their antecedents are non-anaphoric.

An ontological concept list is maintained out of these antecedents. 1050 concepts are generated during the annotation process. The anaphor is linked to the concept regarding of the presence or absence of the actual antecedent.

The proposed corpus can be compared directly to QurAna corpus as they are both use Quran as the source of the text. The proposed corpus has the advantages over Qurana corpus of a higher accuracy through two levels of semi-automatic and manual verification, and unified way of annotation as the total corpus is annotated by one specialist. Also, QurAna corpus is annotated only by anaphoric links (unlike the proposed corpus) without adding any additional features like person, gender, distance, part of speech, etc.

3.2 Annotation Scheme

The first developed annotation scheme is Lancaster and IBM (UCREL) scheme [11]. UCREL scheme is used to annotate Lancaster/IBM anaphoric Treebank. The Treebank contains 100,000 words which is a subsample of The Associated Press Treebank (AP) corpus.

Lancaster/IBM project aimed at annotating all kinds of anaphoric relations including anaphors, cataphors, and bridging anaphors. Under this scheme, the antecedent is surrounded by brackets and given a unique index number; and the anaphor or cataphor is preceded by the symbol 'REF' and the index number. Another symbols are used to indicate the direction of the reference (either cataphor or anaphor); these are '<' for anaphoric relation or '>' for cataphoric relation.

Another annotation scheme is MUC-7 annotation scheme [12]. It uses SGML tagging that is compatible with other several schemes like contemporary XML. The following is an example from this annotated corpus.

```
<COREF ID="100">Lawson Mardon Group Ltd.</COREF> said <COREF
ID="101" TYPE="IDENT" REF="100">it</COREF>
```

The MATE/GNOME scheme [13] was designed in one core scheme and three extensions. The core scheme deals with relations that can be done with MUC (i.e. IDENT). The extended scheme deals with more complex set of relations (like bridges) and extended range of anaphoric expressions.

AQA [2] is a multilingual Anaphora annotation scheme for Question Answering that can be applied in machine learning to improve the question answering systems. It is inspired by the MATE meta-scheme and used to annotate the CLEF 2008 corpus in Spanish, Italian, and English.

4. PROPOSED METHODOLOGY

The aim is to produce a large Arabic corpus with 24,653 personal pronouns drawn from the Holy Qur'an script, with multidimensional features containing: antecedent's starting and ending IDs, distance between the anaphor and its antecedent in terms of verses, words, and segments, part of speech of each segment, gender, number, person, morphological features, and chapter type (Makki/Madani). In addition, it is aimed at proposing a bank of patterns which is a rule set that can be applied to the corpus to extract noun phrases as antecedent candidates.

The following sections describe the process of developing annotated corpus, implementing the multi-dimensional feature vector, and building the bank of patterns.

4.1 System Design and Implementation

Figure 1 illustrates the main components and methodology used to execute this solution, and those components are briefly described next.

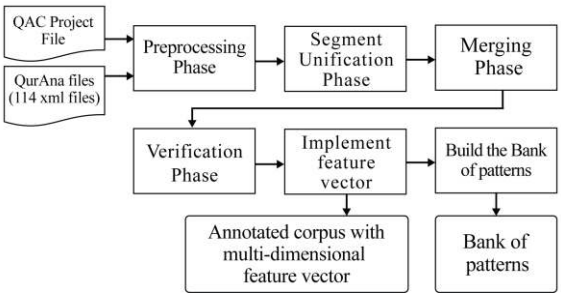


Fig 1: System design

4.1.1 Preprocessing Phase

The proposed corpus consists of 127,795 word segments. A single word (a stem, together with fused suffixes and prefixes) is splitted into prefix, stem, and suffix. Each part is called segment.

A java object model called "segment" is created to represent every segment in the proposed corpus. The model "segment" contains objects for all anaphoric information related to segment like segment ID, segment location, transliteration of the segment, part of speech, distance, antecedent ID, and every other information will be presented in the corpus.

The proposed work makes use of the QurAna annotated corpus [6] to extract the segments of Quran text; also it makes use of their annotation of suggested antecedents of anaphors.

QurAna corpus is made public for download[1] and it is available for Quranic scholar, students, and researchers in the computational linguistics specially those interesting in anaphora resolution.

Another input is QAC project file [10] which is used to extract part of speech and other morphological features of Quranic segments. QAC file is publically available for research purpose[2].

Preprocessing of QurAna and QAC files was needed to prepare the corpus to be suitable for this work.

Preprocessing of QurAna: QurAna corpus consists of 114 xml files; First step was merging all 114 xml files in one file contains all Qur'an's chapters. Merging process consists of reading files one by one using java project, remove comments and headers from each file, and write the chapter in the main file which contains all Quran's chapters.

Next step is to parse the resulted xml file to be read in the java project. SAXParser is used to parse xml file. SAXParser is an API for xml which provides a mechanism for reading data from xml file. During parsing process, fields in java "segment" model which are related to QurAna corpus are filled. Figure 2 illustrates preprocessing of QurAna files.

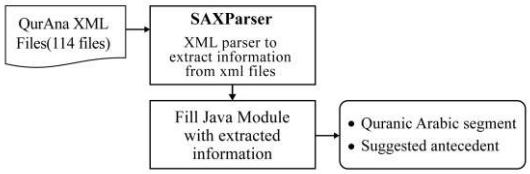


Fig 2: Preprocessing phase of QurAna Files

Preprocessing of QAC: QAC is an on-line annotated linguistic resource with multiple layers of annotation including morphological segmentation, part-of-speech tagging, syntactic analysis using dependency grammar and a semantic ontology.

QAC consists of one plain text file with all Qur'an in one file. Corpus represents Quranic text in records. Each record represents one morphological segment with its location in the Qur'an. Simple parser is created to parse QAC corpus using native java read and write file. Every record in QAC is tokenized and the needed information is extracted. Java "segment" model is filled with this information. Figure3 illustrates preprocessing of QAC files

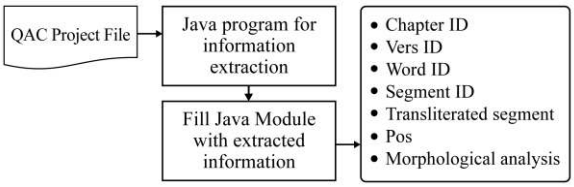


Fig 3: Preprocessing phase of QAC File

4.1.2 Segment Unification Phase

Due to the fact that QAC and QurAna are segmented differently, the number of segments in QurAna corpus was 127,795; on the other hand, the number of segments in QAC corpus was 128,240. Unifying the segmentation is needed in order to match every segment extracted from QurAna with its corresponding information extracted from QAC. Figure 4 shows an imaginary representation of QAC and QurAna list. Note the difference in the segmentation.

QAC segments	ID
قل	1
يا	2
أيها	3
ال	4
ناس	5
إن	6
ي	7
رسول	8
الله	9
إلي	10
كم	11
جميعا	12

QurAna segments	ID
قل	1
يأيها	2
ال	3
ناس	4
إن	5
ي	6
رسول	7
الله	8
إلي	9
كم	10
جميعا	11

Fig 4: An imaginary representation of QAC and QurAna Arrays.

Figure 5 show the unification process in details.

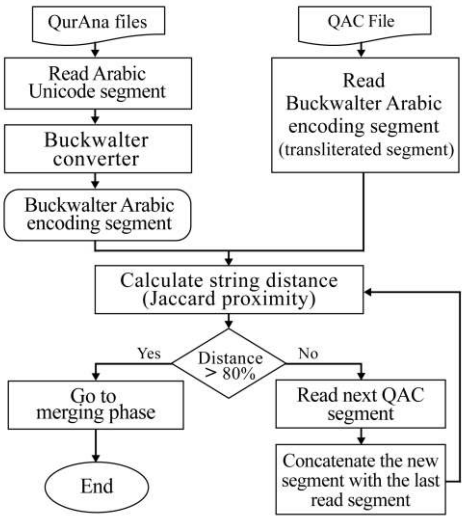


Fig 5: Segment unification process

A program is created to retrieve one by one segment from QAC segment list and its corresponding segment from QurAna corpus using loop.

Because QurAna segments are written in Arabic Unicode Characters, and QAC segments are transliterated in Buckwalter Arabic encodings, so a converter is needed to convert one encoding to the other to be able to compare two segments with each other.

Buckwalter package from Stanford JavaNLP API package[3] is used to convert between Unicode and Buckwalter encodings of Arabic. Some Unicode characters were added to Buckwalter converter to satisfy Quranic Arabic special characters. The modification was guided by Extended Buckwalter encoder and decoder in JQuranTree API which is

1 http://www.textminingthequran.com/wiki/Pronoun_Reference_in_the_Quran#download

2 http://corpus.quran.com/

[3] To download Stanford package and see the documentation, visit: http://nlp.stanford.edu/software/corenlp.shtml

a set of Java APIs for accessing and analyzing the Quran released as an open source project.

Next step, Jaccard Proximity (Jaccard similarity coefficient) is used to measure the similarity between two segments. Jaccard similarity coefficient is a statistic used for comparing the similarity and diversity of sample sets, and is defined as the size of the intersection divided by the size of the union of the sample sets.

Java API from LingPipe[4] for Jaccard similarity is used to compare between two corresponding segments from QAC and QurAna. Similarity coefficient of 80% is chosen after examining the program with many numbers and on 80% the result was 100% correct. If proximity is greater than 80% then it should be the same word segment. So, segment information from QurAna and QAC is merged in one record. If proximity is less than 80% it indicates that the difference in segmentation is found. In this point, the program reads the next QAC segment and concatenates it with its previous segment, then compares the new segment with the same QurAna segment. In 100% of cases the concatenated QAC segment is matched with QurAna segment. As example, see segment 2 and 3 in figure 4.

4.1.3 Merging Phase

Two types of segmentation differences are detected. First one is one token in QurAna is presented in two segments with different Part-of-speech and morphological features for each of them in QAC. Second type is one token in QAC is presented in two segments in QurAna.

$$\text{QurAna segment} = \text{QAC segment}_1 + \text{QAC segment}_2$$

$$\text{QAC segment} = \text{QurAna segment}_1 + \text{QurAna segment}_2$$

Fig 6: Types of segmentation differences between QAC and QurAna

The best choice was not to edit anything in QurAna segments and to modify QAC to meet QurAna segmentation because QurAna is the one where pronouns are annotated with its antecedents using segments ID, so segments ID have not to be changed and all QurAna segments have not to be re-numbered or antecedents' references for all anaphors have not to be changed.

For the first type the two segments of QAC are merged into one segment with unique ID, and their part of speech and morphological features will be concatenated. In the second type, QAC segment is split into two segments with different unique IDs for each one and different part of speech and morphological features.

The online visualization word by word tool [4] that shows the Arabic grammar, syntax and morphology for each word in the Holy Quran is used to obtain the new part-of-speech and morphological analysis for the new generated segments.

4.1.4 Verification Phase

The output of the previous phase is a corpus that9 contains: Arabic Unicode segments of all the Quran, Buckwalter Arabic encoding segments, segment location (chapter ID, verse ID, word ID, and segment ID), part of speech, morphological features, and the antecedent starting and ending IDs that extracted from QurAna corpus.

The output corpus goes through two levels of verification: syntax and semantic verification. Syntax verification is the process of ensuring that the recorded starting and ending IDs refer to the actual antecedent of the anaphor. Semantic verification is the process of ensuring that the referenced antecedent is semantically correct. Figure 7 illustrates the verification process.

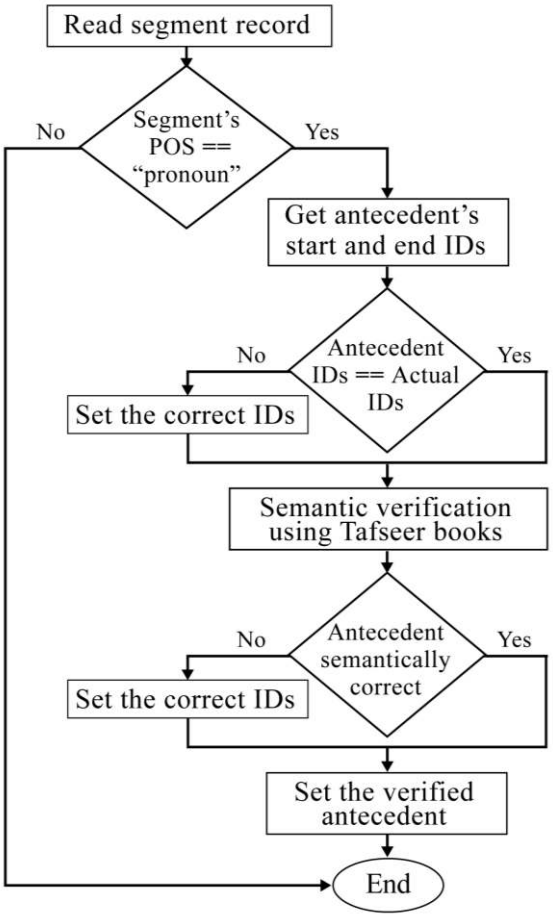


Fig 7: Verification process

Regarding the fact that the starting and ending indexes that represent antecedents for each pronoun are recorded manually, there are lots of annotation mistakes. Many pronouns refer to incorrect antecedents. Figure 8 shows an example of annotation mistake for the pronoun (ون/they) in the Quranic verse:

ولو شاء الله لجعل الناس$_1$ أمه واحده ولا يزالون$_1$ مختلفين (هود-118)[5]

And if your Lord had so willed, He could surely have made mankind one Ummah$_1$ [nation or community (following one religion only i.e. Islam)], but they$_1$ will not cease to disagree (Hud – 118).

[4] To download LingPipe package and read the documentation, visit: http://alias-i.com/lingpipe-3.9.3/demos/tutorial/read-me.html

[5] The Holy Quran has been translated into the modern English Language by Dr. Muhammad Taqi-ud-Din Al-Hilali, Ph.D. & Dr. Muhammad Muhsin Khan (http://www.dar-us-salam.com/TheNobleQuran)

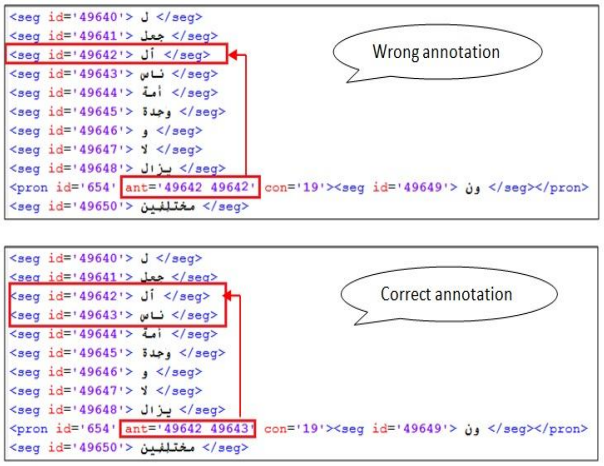


Fig 8: An example of annotation mistakes

Another kind of mistakes arises because the annotation process doesn't follow the same rules as more than one annotator contributed QurAna corpus. Every contributor annotates segments from his point of view and both are likely right. Figure 9 shows this kind of mistake for the pronoun (هم/they) in the Quranic verse:

صراط الذين أنعمت عليهم، غير المغضوب عليهم ولا الضالين (الفاتحة – 7)

The Way of those on whom, you have bestowed Your Grace, not (the way) of those who earned Your Anger nor of those who went astray.

7	6	5	4	3	2	1	ID
هم	علي	ت	أنعم	لذين	ال	صراط	Segment

	Antecedent's starting ID	Antecedent's ending ID
Annotator A	2	7
Annotator B	0	0

Fig 9: An example of annotation mistake

After finishing the first level of verification, all annotation mistakes should be corrected, and the second level of verification will take place.

The Qur'an is classified as Classical Arabic text, beside that fact, the Holy Qur'an is not easily understood by unspecialized people. For that reason, the semantic verification is important to be done. Tafseer books (interpretation of the Quran) are consulted to understand the meaning of the verses to ensure that the annotated antecedent of every anaphor is semantically correct. The consulted books used in semantic verification are:

Tafsir Ibn Kathir (تفسير بن كثير): It is the most popular interpretation of the Qur'an in the Arabic language. It is written by Hafiz Ibn Kathir (701:774 AH; 1300:1373 AD), student of famous scholar of Islam, Ibn Taymiyyah Al Harraanee.

Tafsir Al-Tabari (تفسير الطبري): Jami al-bayan fi ta'wil ay al Qur'an (جامع البيان في تأويل آي القرآن), popularly called Tafsir al Tabari. It is a classic Sunni tafsir by the Persian scholar Muhammad ibn Jarir al-Tabari (224–310 AH; 838–923 AD). The work is a collective tafsir, very rich in content, and a major source for academic research and historical inquiry.

Tafsir Al-Baghawi (تفسير البغوي): ma'alim al-tanzil (معالم التنزيل) popularly called Tafsir al-Baghawi. It is written by Abu Muhammad Al-Husayn Ibn Mas'ud Al-Baghawi (433~436-516 AH), a Persian Muslim Mufassir, hadith scholar and a Shafi'i faqih. The book is a classical Sunni tafsir written as an abridgement of Tafsir al-Tha'labi.

Tafsir Al- Qurtubi (تفسير القرطبي): Also known as Al-Jami' li Ahkam al-Qur'an (الجامع لأحكام القرآن) or Tafsir al-Jami' (تفسير الجامع).Tafseer Al-Qurtubi is a 12-volume of preeminent classical works of Qur'an exegesis. It is written by the famous mufassir, muhaddith and faqih scholar Imam Abu Abdullah Al-Qurtubi (1214:1273).

4.1.5 Implementing the Feature Vector

There are several factors (also called features) that facilitate identifying the antecedent of an anaphoric expression in statistical anaphora resolution systems. These are: the distance between an anaphoric expression and its antecedent; lexical constraints such as gender and number agreement that are used to eliminate some antecedent candidates; etc. [14].

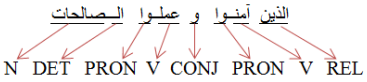
The availability of those features in an annotated corpus is considered as a major contribution in this work as it will help in building statistical anaphora resolution systems without consuming lots of processing and time.

The following section explains the features provided in the developed corpus.

Gender agreement is the requirement that the anaphor and its antecedent must agree in gender. In the developed corpus gender annotation for each segment is provided and it is tagged with 'M ' for male, 'F ' for female, and '-' for null (such as verb segment). **Number agreement** requires that the anaphor and its antecedent must agree in number. It is tagged with 'S' for singular, 'D' for dual, 'P' for plural, and '-' for null. **Person agreement** requires that the anaphor and its antecedent must agree in person. It is tagged with '1' for first person, '2'for second person and'3' for third person, and '-' for null. **Distance in terms of verses** is used as the number of Quranic verse boundaries between an anaphor and its antecedent. If the anaphor and its antecedent are in the same verse, the value is 0. If they are one verse apart, the value is 1; if they are two verses apart, the value is 2; and so on. **Distance in terms of words** is used as the number of word boundaries between an anaphor and its antecedent. If the antecedent is in the last preceding word of the anaphor, the value is 0. If they are one word apart, the value is 1; and so on**. Distance in terms of segments** is used as the number of segment boundaries between an anaphor and its antecedent. The case of having an anaphor and its antecedent in the same segment is not possible. If the antecedent is the last preceding segment of the anaphor, the value is 0. If they are one segment apart, the value is 1; if they are two segments apart, the value is 2; and so on**. Chapter type** The Quranic Chapters or Surahs were revealed to the Islamic prophet Muhammad in the city of Mekka and the city of Medina. The Meccan chapters (Makki) are revealed any time before the migration of the prophet Muhammad and his followers from Mecca to Medina. The Medinan surahs (Madani) are revealed after the move to the city of Medina. Each set of Makki and Madani chapters has its own stylistic, content, and subject characteristics, way of discourse, usage of language components and rhetoric. The identification of this feature gives a good domain and linguistic knowledge which contributes in identifying the antecedent of an anaphoric expression.

4.1.6 Implementing the bank of patterns

Due to the nature of Arabic language especially Quranic text, the determination of the noun phrases that can be antecedent candidates is not an easy process. For that reason, the authors are motivated to propose a bank of sentence patterns that contains the part-of-speech tags for each noun phrase that represents the antecedent candidate. It is a rule set that can be applied to the corpus to extract noun phrases. The following is an example of the pattern "REL V PRON CONJ V PRON DET N":



The implementation of the bank goes through the following process: A program is created to iterate over the corpus, and the segments that are tagged as pronoun are retrieved. As explained above, each anaphor segment carries information about the starting and ending IDs of its antecedent, so the antecedent phrase can be easily retrieved. The word segments that compose the antecedent phrase are printed and its corresponding part of speech tags are captured.

For the first occurrence of each pattern, a record is created in the bank of patterns with the new pattern. If the pattern is repeated again, the count of the recorded pattern is incremented. Following the process described above, a bank of patterns is created and filled with 481 patterns which represent 24,653 antecedent phrases.

Patterns	frequency	length of pattern
REL V PRON CONJ V PRON P N PN CONJ V PRON DET N	3	14
REL P DET N N	4	5
DET N P REL V PRON DET N P N PRON	2	11
COND V PRON V	1	4
DET PN V DET PN	2	5
REL V PRON DET N SUB V PRON PRON CONJ V PRON P PN	1	14

Fig 10: sample of the implemented bank of patterns

5. CORPUS ANALYSIS AND RESULTS

Following the proposed methodology, the entire Quran was annotated. Each segment in the corpus has a very specific location, Arabic Unicode representation, Buckwalter encoding representation, and tagged with part of speech, morphological features, person, number, gender, chapter type. Moreover, the pronoun segments are annotated with additional features: antecedent starting and ending IDs, distance in terms of verse, distance in terms of word, and distance in terms of segment. Additionally, the bank of patterns was implemented and filled with 481 different patterns.

The main advantages of the developed corpus are i) combining the basic features and advantages of QAC and QurAna corpus in addition to the new valuable features and patterns. ii) Avoiding the shortcoming of Qurana corpus by providing a unified way of annotation as the whole corpus is annotated by only one researcher. iii) Providing two levels of semi-automatic and manual verification.

Table 1 gives a quantitative measure of this corpus. Note that the personal pronouns are tagged and relative and demonstrative pronouns are excluded because they represent only about 16% of the total number of pronouns and the majority of them are non-anaphoric. Although there are some cataphoric cases in the corpus, but they are only 65 (0.26%) of the antecedents. Among the total number of pronouns, there are 55.6% pronouns which have antecedent, while 44% their antecedent is not available in the text (hidden antecedent).

Table 2. Quantitative measure from the developed corpus

Measure	Count	%
Personal pronouns	24653	84.18%
Relative pronouns	3573	12.20%
Demonstrative pronouns	1061	3.62%
Total	**29287**	
Anaphor with existing antecedent	13713	55.60%
Anaphor with hidden antecedent	10875	44.10%
Cataphors	65	0.26%
Total	**24653**	

Table 2 gives a statistics of this corpus and describes the distribution of the pronouns in the Quran.

Table 2. Statistics of the various types of pronouns in the Quran

Pronoun types	Count	%
Person		
1st person	3908	15.85%
2nd person	6846	27.77%
3rd person	13899	56.38%
Gender		
Masculine	18889	76.62%
Feminine	1633	6.62%
Others	4131	16.76%
Number		
Singular	8007	32.48%
Dual	372	1.51%
Plural	16274	66.01%

Table 3 shows the distances between the anaphor and its antecedent in terms of verses, words, and segments. Among the pronouns which have antecedents, the majority of them are within the same verse as anaphor 55.7%. In the second place, 19.5% of the antecedents are found within 9 verses from the anaphor. In terms of words distance, 51.16% of the antecedents are found within 9 words away from its anaphor, and only about 1% antecedents are found in the last preceding word of the anaphor.

Table 3. Distances between anaphor and antecedent

Distance	Occurrence		
	Verse	Word	Segment
0	7639	111	73
1 to 9	5652	7015	5471
10 to 19	356	2602	2396

20 to 29	61	1267	1431
30 to 49	5	1231	1586
50 to 100	0	1001	1588
100 to 200	0	325	815
200 to the end (1060)	0	161	353

Considering the statistical study made above, table 4 suggests the search scope that should be identified to look for the antecedents for anaphora resolution systems that will use the developed corpus. According in the study and the previous tables, about 91% antecedents are matched correctly within the search scope of 5 verses (the current and 4 preceding verses with anaphor). About 96% antecedents are matched correctly within the search scope of 100 words preceding an anaphor. In terms of segments, 91% antecedents are found in search scope of 100 segments.

Table 4. Recommended search scope

Recommended search scope	Distance	Frequency	Percentage
In terms of verses	0 to 4	12458	90.85%
In terms of words	1 to 99	13116	95.65%
In terms of segments	1 to 99	12472	90.95%

Table 5 and 6 gives brief information about the bank of patterns and show the most frequent patterns in the developed list. Note that the pattern (PN) such as "محمد/Muhammad" is the most frequent pattern in the corpus (10.5%). Another (10%) of the antecedents are matched with the pattern (DET N) such as "السماء/ the sky", and "الكتاب/the book". 6% antecedents are matched with pattern (REL V PRON) such as "الذين آمنوا/those who believe". 5% antecedents are matched with (N) pattern such as "جنات/gardens".

Table 5. Summary about bank of patterns

Bank of patterns	Count
Total number of antecedent phrases	24653
Total number of generated patterns	481
Length of the longest pattern	83
Length of the shortest pattern	1

Table 6. Most frequently patterns in the bank

Most frequently patterns	Frequency
PN	2575
DET N	2451
REL V PRON	1459
N	1150

6. CHALLENGES AND BARRIERS

In addition to the traditional challenges of natural language processing for English, there are unique complexities for the Arabic language, which make the work on Arabic anaphora resolution more difficult than other languages. In [15], the authors stated in their work the morphological and syntactic challenges in the Arabic language in general. Here, some difficulties encountered in the Arabic anaphora resolution task are presented.

Grammatical Shifts in Quran: The Quran makes grammatical shifts deliberately, as a result the number or person agreement between the pronouns and its antecedent is violated. There are many types of shifts in the Quran such as: changes in person, between 1st, 2nd and 3rd person, changes in the number, between singular, dual and plural, and changes in the tense of the verb. The next example shows a sudden shift in the pronoun from 2nd person singular (النَّبِيُّ/ Prophet) to 2nd person plural pronoun "تم/you" in (طَلَّقْتُمُ/divorce).

يَا أَيُّهَا النَّبِيُّ إِذَا طَلَّقْتُمُ النِّسَاء فَطَلِّقُوهُنَّ لِعِدَّتِهِنَّ وَأَحْصُوا الْعِدَّةَ (الطلاق -1).

O Prophet! When you divorce women, divorce them at their 'Iddah (prescribed periods), and count (accurately) their 'Iddah (periods) (Al-Talaq-1).

Hidden Antecedents in Quran: In some cases, especially in Qur'anic texts, the pronoun may refer to something which is not presented in the text, such as the example below. The pronoun (هو) refers to "Allah", which isn't presented in the text. The human mind can determine the hidden antecedent by using the knowledge that Allah is the only one who knows the unseen. But for the anaphora resolution algorithm, it may fail.

وَعِندَهُ$_1$ مَفَاتِحُ الْغَيْبِ لاَ يَعْلَمُهَا إِلاَّ هُوَ$_1$ (الانعام -59).

And with Him[1] are the keys of the Ghaib (all that is hidden), none knows them but He[1] (Al An'am -59).

Ambiguity of the antecedent: In some cases, the anaphora resolution's algorithm fails to identify the correct antecedent because of the ambiguity of the antecedent. In these cases, external knowledge about the context is required to identify the correct antecedent [16] such as the example:

اكلت الطفلة$_1$ الموزه$_2$ لانها$_1$ جائعة بالرغم انها[2] لم تنضج بعد

The girl$_1$ ate the banana$_2$ because she$_1$ is hungry although it isn't ripped yet.

In each case the pronoun (ها) refers to something different, in 1 الطفلة/the girl, in 2 الموزة/the banana. Each "ها" is going to be interpreted correctly if the knowledge is used that the girl, being human, are likely to be hungry and bananas, being fruit, are likely to be ripe [17][18].

Complex Sentence Form Structure: Unlike other languages, Arabic has a unique structure form, which may combine the verb, subject, and object in one word such (انجيناهم) which means "we saved them". The phrase has to be broken up, and the subject has to be recognized before starting the process of anaphora resolution.

Free Word Order: The Arabic sentence constituents can be swapped without affecting structure or meaning such as the following example. This adds more syntactic and semantic ambiguity, and requires more complex analysis.

وَإِذِ اسْتَسْقَى مُوسَى لِقَوْمِهِ (VSO)

And (remember) when Musa (Moses) asked for water for his people

وَإِذِ ابْتَلَى إِبْرَاهِيمَ رَبُّهُ بِكَلِمَاتٍ (VOS)

And (remember) when the Lord of Ibrahim (Abraham) [i.e., Allah] tried him with (certain) Commands

The Lack of Arabic Corpora Annotated with Anaphoric Links: Still the major problem is the lack of Arabic Corpora annotated with anaphoric relations; however, it is needed in most NLP systems. The annotation task of anaphoric relations is very time consuming and requires a huge effort from the human annotator. In addition, the annotation of written texts in Arabic is more difficult and time consuming than that in other language such as French or English because of the fact that the density of pronouns in Arabic texts is higher than that in other languages [9].

7. CONCLUSION AND FUTURE WORK

This paper introduced a solution for one of the major anaphora resolution problems which is the scarcity of annotated corpora in anaphora resolution. It proposed a large language resource for researchers and linguists who are investigating anaphora resolution systems. The developed corpus has the advantage of combining the basic features of two of the rich language resources; QAC and QurAna corpus. In addition, a set of anaphoric features are defined to fill the feature vectors in the AR systems which uses vector based machine learning techniques. Also a bank of patterns has been developed to ease the process of noun phrase extraction which is an important step in extracting antecedent candidates in AR systems.

The authors intend to expand the developed corpus to cover other kinds of classical texts such as Al-Hadith Al-Nabawi books (e.g. Sahih Bukhari). Also the authors consider adding more useful and convenient set of features to the feature vector to improve the performance of anaphora resolution systems.

8. REFERENCES

[1] Hammami, S., Belguith, L., and Ben Hamadou, A. (2009). "Arabic Anaphora Resolution: Corpora Annotation with Coreferential Links". The International Arab Journal of Information Technology, (volume 6), pp. 480-488

[2] Boldrini, E., Puchol-Blasco, M. Navarro, B., Martínez-Barco, P., Vargas-Sierra, C. (2009). "AQA: a multilingual Anaphora annotation scheme for Question Answering". Procesamiento del Lenguaje Natural, Revista. N. 42 (marzo 2009). ISSN 1135-5948, pp. 97-104

[3] Elghamry, K., Al-Sabbagh, R., El-Zeiny, N. (2007). "Arabic Anaphora Resolution Using Web as Corpus", Proceedings of the seventh conference on language engineering, Cairo, Egypt.

[4] Dukes, K., and Habash, N. (2010). "Morphological Annotation of Quranic Arabic". Language Resources and Evaluation Conference (LREC). Valletta, Malta.

[5] Ali, M., and Fish, D. (2011). "The Holy Quran English Translation and commentary" [e-book]. USA: Ahmadiyya Islamic Society. Available at: Google Books [Accessed 18 April 2015].

[6] Sharaf A., and Atwell E. S. (2012). "QurAna: Corpus of the Quran annotated with Pronominal Anaphora". Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12). European Language Resources Association (ELRA), Istanbul, Turkey

[7] Mitkov, R., Belguith, L., (1998). "Multilingual robust anaphora resolution", In Proceedings of the 3rd Conference on Empirical Methods in Natural Language Processing, Granada, Spain, pp. 7-16

[8] Mitkov, R., 1998a. "Evaluating anaphora resolution approaches". In Proceedings of the Discourse Anaphora and Anaphora Resolution Colloquium (DAARC'2)

[9] Tutin A., Trouilleux F., Clouzot C., Gaussier E., Zaenen A., Rayot S., and Antoniadis G., (2000). "Annotating a Large Corpus with Anaphoric Links". in Proceedings of the Discourse Anaphora and Reference Resolution Conference, pp. 134-137, UK.

[10] Dukes, K., Atwell, E., and Sharaf, A. (2010). "Syntactic Annotation Guidelines for the Quranic Arabic Dependency Treebank". Proc LREC'2010, Valetta, Malta

[11] Garside R., Fligestone S. and Botley S. (1997). "Discourse annotation : anaphoric relations in corpora", in R.Garside, G. Leech & A. McEnery (eds), Corpusannotation : Linguistic Information from Text Corpora, London, Longman

[12] Hirschman, L. and Chinchor, N. (1997). "MUC-7 coreference task definition".In MUC-7 Proceedings. Science Applications International Corporation

[13] Davies S., Poesio M., Bruneseaux F., and Romary L. (1998). "Annotating Coreference in Dialogues: Proposal for a Scheme for MATE"

[14] Seddik, KH., and Farghaly A., (2014). "Anaphora/Coreference Resolution", to appear in Zitouni I., "Natural Language Processing Approaches to Semitic Languages (Theory and Applications of Natural Language Processing)". Springer-Verlag Berlin Heidelberg, p. 247-277

[15] Seddik M.K., Farghaly A, Fahmy A. (2011) "Arabic anaphora resolution in Holy Qur'an text". In: Proceedings of ALTIC 2011 conference on Arabic language technology, Alexandria, pp 21–28

[16] Baker, K., Brunner, A., Mitamura, T., Nyberg, E., Svoboda, D., Torrejon, E. (2002). "Pronominal Anaphora Resolution in the KANTOO Multilingual Machine Translation System", Language Technologies Institute, Carnegie Mellon University

[17] Wilks, Y., (1973). "Preference semantics". Stanford AI Laboratory memo AIM-206. Stanford University.

[18] Wilks, Y., (1975). "Preference semantics". The formal semantics of natural language ed. by E. Keenan, Cambridge University Press.