Abstract

We introduce a list of Arabic multiword expressions (MWE) collected from various dictionaries. The MWEs are grouped based on their syntactic type. Every constituent word in the expressions is manually annotated with its full context-sensitive morphological analysis. Some of the expressions contain semantic variables as place holders for words that play the same semantic role. In addition, we have automatically annotated a large corpus of Arabic text using a pattern-matching algorithm that considers some morpho-syntactic features as expressed by a highly inflected language, such as Arabic. A sample part of the corpus is manually evaluated and the results are reported in this paper.

1 Introduction

A multiword expression (MWE) refers to a multiword unit or a collocation of words that co-occur together statistically more than chance. A MWE is a cover term for different types of collocations, which vary in their transparency and fixedness. MWEs are pervasive in natural language, especially in web based texts and speech genres. Identifying MWEs and understanding their meaning is essential to language understanding, hence they are of crucial importance for any Natural Language Processing (NLP) applications that aim at handling robust language meaning and use. In fact, the seminal paper (Sag et al., 2002) refers to this problem as a key issue for the development of high-quality NLP applications. MWEs are classified based on their syntactic constructions. Among the various classes, one can find the Verb Noun Constructions (VNC), Noun Noun Construction (NNC) and others. A MWE typically has an idiosyncratic meaning that is more or different from the meaning of its component words. In this paper we focus on MWEs in Arabic. Like many other Semitic languages, Arabic is highly inflected; words are derived from a root and a pattern (template), combined with prefixes, suffixes and circumfixes. As opposed to English equivalents, Arabic MWEs can be expressed in a large number of forms, expressing various inflections and derivations of the words while maintaining the exact same meaning, for example, \textggmmD [flAn] Eynyh En [Al>mr] ¹, “[one] disregarded/overlooked/ignored [the issue]”, literally, closed one’s eyes, vs. \textggmmDt [flAnp] EynyhA En [Al>mr], “[one\_fem] disregarded/overlooked/ignored\_fem [the issue]”, where the predicate takes on the feminine inflection. However, in many cases, there are morphological features that cannot be changed in different contexts, for example, \textmkrh \textxAk \textla bTl, “forced with no choice”, in this example, regardless of context, the words of the MWE do not agree in number and gender with the surrounding context. These are considered frozen expressions. One of the challenges in building MWE list for Arabic is to identify those features and document them in every MWE. Our resource is available for download.²

We have manually collected a large number of MWEs from various Arabic dictionaries, which are based on MSA corpora, and then filtered by Arabic

¹ We use the Buckwalter transliteration for rendering Arabic script in Romanization throughout the paper (Buckwalter, 2002).
² To get a direct access, please send a request to one of the authors.
native linguists. We then classified them based on their syntactic constructions, considering the relevant syntactic phenomena expressed in Arabic. The MWEs were manually annotated with the context-sensitive SAMA (Maamouri, 2010) morphological analysis for each word to assist an automated identification of MWEs in a large corpus of text. Part of the Arabic Gigaword 4.0 (Parker, 2009) is processed accordingly and the MWEs are annotated based on a deterministic algorithm considering different variants of every MWE in our list. There are diverse tasks that require a corpus with annotated MWEs, which have not been addressed in Arabic due to the lack of such a resource. However, a lot of attention is put on those tasks when implemented in English and other languages. Among those tasks, classifying MWEs in a running text is the most common one. Diab and Bhutada (2009) applied a supervised learning framework to the problem of classifying token level English MWEs in context. They used the annotated corpus provided by Cook (2008), a resource of almost 3000 English sentences annotated with VNC usage at the token level. Katz and Giesbrecht (2006) carried out a vector similarity comparison between the context of an English MWE and that of the constituent words using Latent Semantic Analysis to determine if the expression is idiomatic or not. In work by Hashimoto and Kawahara (2008), they addressed token classification into idiomatic versus literal for Japanese MWEs of all types. They annotated a corpus of 102K sentences, and used it to train a supervised classifier for MWEs. Using MWEs in machine translation is another application. Carpuat and Diab (2010) studied the effect of integrating English MWEs with a statistical translation system. They used the WordNet 3.0 lexical database (Fellbaum, 1998) as the main source for MWEs. Attia et al., in 2010, extracted Arabic MWEs from various resources. They focused only on nominal MWEs and used diverse techniques for automatic MWE extraction from cross-lingual parallel Wikipedia titles, machine-translated English MWEs taken from the English WordNet and the Arabic Gigaword 4.0 corpus. They found a large number of MWEs, however only a few of them were evaluated.

In this paper, we describe the process of manually creating a relatively comprehensive Arabic MWE list. We use the resulting list to tag MWE occurrences in context in a corpus.

The paper is organized as follows: In Section 2 we describe the process of creating the Arabic MWE list. Section 3 discusses the algorithm for automatic deterministic tagging of MWEs in running text, based on pattern matching. Sections 4 and 5 summarize the results of applying the pattern-matching algorithm on a corpus. Finally, we conclude in Section 6.

2  Arabic MWE List

Our Arabic MWE list is created based on a collection of about 5,000 expressions, which is manually extracted from various Arabic dictionaries (Abou Saad, 1987; Seeny et al., 1996; Dawod, 2003; Fayed, 2007). Each MWE is preprocessed by the following steps: 1) cleaning punctuations and unnecessary characters, 2) breaking alternative expressions into individual entries, and 3) running MADA (Habash and Rambow, 2005; Roth et al., 2008) on each MWE individually for finding the context-sensitive morphological analysis for every word. Some of the extracted MWEs are originally enriched with placeholder generic words that play the same semantic role in the context of the MWE. That set of generic words is manually normalized and reduced to a group of types, as shown in Table 2.

<table>
<thead>
<tr>
<th>Generic Type</th>
<th>Semantic Role</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>$flAn$</td>
<td>Agent/Patient</td>
<td>$gr , flAn , Ey, nA$ “pleased someone”</td>
</tr>
<tr>
<td>$k*A$</td>
<td>Goal</td>
<td>$ELY , Hs, Ab , k*A$ “at the expense of that/this”</td>
</tr>
<tr>
<td>$&lt;mr$</td>
<td>Source</td>
<td>$&lt;mr , Abn , ywmh$ “something very new”</td>
</tr>
</tbody>
</table>

Table 1 – Generic Types

Generic words are sometimes provided with or without additional clitics. For example, in the MWE $lEt$ [bflAn] $AldnyA$, literally, “the world played-passive with so-and-so”, which could be translated as “life played havoc with so-and-so”, the word $bflan$ “to so-and-so” has the preposition $b$ “with” cliticized to it. Every word that substitutes a generic word (an instantiation) has to comply
with the morphological features of the context surrounding it.

The automatic preprocessing steps we ran on the list are followed by a series of manual ones. We found that the short context we had for every MWE was not sufficient for MADA to return the correct analysis with reasonable precision. Therefore, we had to go over the results and manually select the correct analysis for each word in every MWE. Generic words are also assigned with their correct analysis in context.

The class of each MWE is assigned manually. Arabic is highly inflected; therefore many MWE classes can be identified. However, in this paper, we focus only on the major ones. The following classes are used: Verb-Verb Construction (VVC) as in >xZ [f'A n] w'>ETY “give and take”; Verb-Noun Construction (VNC), for example, md [f'A n] Aljswr “[someone] built bridges” as in extending the arms of peace; Verb-Particle Construction (VPC) as in mDY [f'A n] fy “[someone] continues working on”; Noun-Noun Construction (NNC) as in Enq {lzdAj p “bottleneck”; Adjective Noun Construction (ANC) as in [f'A n] wAsE {ld&f “[someone] broad-minded”.

The final list comprises 4,209 MWE types. Table 2 presents the total number of MWE types for each category.

<table>
<thead>
<tr>
<th>MWE Category Type</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>VVC</td>
<td>41</td>
</tr>
<tr>
<td>VNC</td>
<td>1974</td>
</tr>
<tr>
<td>VPC</td>
<td>670</td>
</tr>
<tr>
<td>NNC</td>
<td>1239</td>
</tr>
<tr>
<td>ANC</td>
<td>285</td>
</tr>
</tbody>
</table>

Table 2 – Arabic MWEs by category types

3 Deterministic Identification of Arabic MWEs

We developed a pattern-matching algorithm for discovering MWEs in Arabic running text. The main goal of this algorithm is to deterministically identify instances of MWEs from the list in a large Arabic corpus, considering some morphological as well as syntactic phenomena. We use the Arabic Gigaword 4.0 (AGW). ³ To capture the large number of morpho-syntactic variations of the MWEs in context, the pattern-matching algorithm is designed to use some of the information available from the selected morphological analysis for every MWE word, as well as shallow-syntactic information that we automatically assigned for every word in the corpus.

One of our immediate intentions is to use the list of MWEs for learning how to statistically classify new ones in running text. Therefore, we begin here with annotating a large part of the AGW corpus with all the occurrences a MWE given in the list. In order to make some shallow-syntactic features available for the pattern-matching algorithm, we pre-processed the AGW with AMIRAN, an updated version of the AMIRA tools (Diab et al., 2004, 2007). AMIRAN is a tool for finding the context-sensitive morpho-syntactic information. AMIRAN combines AMIRA output with morphological analyses provided by SAMA. AMIRAN is also enriched with Named-Entity-Recognition (NER) class tags provided by (Benajiba et al., 2008). For every word, AMIRAN is capable of identifying the clitics, diacritized lemma, stem, full part-of-speech tag excluding case and mood, base-phrase chunks and NER tags. Part of this information was used in previous work for processing English MWEs.

When looking for Arabic MWEs in the pre-processed corpus, there are two important issues that the pattern-matching algorithm is addressing: morphological variations and gaps. We now elaborate further on each one of them.

Morphological variations: As mentioned above, Arabic is highly inflected; clitics may be attached to reflect definiteness, conjunction, possessive pronouns and prepositions. This fact forces the pattern-matching algorithm to match words on a more abstract level than their surface form. The algorithm considers different levels of representation for each of the words. Those levels are matched based on the information provided by AMIRAN on corpus words, on the one hand, and the morphological analyses that are selected manually for every MWE word on the other hand. In the experiment reported here, we match words on the lemma level. The lemma provided by AMIRAN and the one manually chosen by MADA/SAMA analyses are taken from the same pool, hence matching is enabled. It is worth noting

³ http://www.ldc.upenn.edu/Catalog/catalogEntry.jsp?catalogId=LDC2009T30
that in Arabic, the lemma is a generic name for a
group of words that can be derived from one of its
underlying stems, sharing the same meaning. For
instance, the noun *bnt* “girl” and its plural form
*bnt* “girls” are reduced to the same lemma form
*bnt*. Obviously, perfect and imperfect forms of a
verb are also assigned to the same lemma. A
lemma form does not include the clitics; for every
corpus word, this information is recorded by
AMIRAN. Since clitics are in many cases
important for matching MWES, the pattern-
matching algorithm considers them. For example,
in the MWE: \(<x^* \ [jlan] \ bAlv<e, \ “[so-and-so]
required”, the proclitic *b* “with” expressed in the
last word, is important for matching.

**Gaps:** Sometimes a MWE can be found with
additional words such as modifiers that are not part
of the original MWE expression words. For
instance, the MWE: *wDET AlHrb \(<wzArhA, \ “the
war is over”, is found in the text: *wDET AlHrb
*AIEAlmp AlvAnyp \(<wzArhA, \ “the second world
war is over”. The nominal modifiers *AlEAlmp
AlvAnyp (“second world…”*) are not present in the
original MWE taken from the list, and therefore
considered as gap fillers. To be able to identify
gaps of MWES in context, the pattern-matching
algorithm uses the part-of-speech and base-
phrase tags provided for every word by AMIRAN. In
the reported experiment, we allowed an MWE to be
matched over gaps of noun-phrase
complementing MWE words. In other words, we
allowed every MWE noun to be matched with a
complete non-recursive noun-phrase that appears
in the text. The matching is performed only on the
first noun of the containing noun-phrase, restricting
our approach using only noun-phrases
expressing the head noun in the beginning of a
phrase. For instance, in the previous example
*AlHrb AIEAlmp AlvAnyp*, “the Second World
War”, is a noun-phrase with a first noun word
*AlHrb “the war”. This noun-phrase matches the
word *AlHrb “the war”* from the list MWE *wDET
AlHrb \(<wzArhA “the war is over”, hence allowing
the entire MWE to be found. Obviously, allowing
gaps of any types would have increased the recall
but on the other hand a large number of false
positive MWES would have been identified.
Currently, only noun-phrases are considered as
potential gap fillers. Considering other phrase
types is left for future work. We plan on
identifying the types of potential gap fillers and
correlating them with the various MWE types.

One of the remaining problems with identifying
MWES deterministically in a running text is that
the exact MWE words can be found in a text,
however given the context, in some cases they are
not idiomatic. This is the case for many VNCs for
instance. Hence, they are not a unified concept – a
word with gaps -- as in our definition of a MWE
usage. Accordingly a token MWE classifier is
required to identify such cases, teasing idiomatic
from literal MWES apart.

4 **Building MWE Annotated Corpus**

We ran the pattern-matching algorithm on a large
part of the AGW after we pre-processed the
documents with AMIRAN. Overall, we had 250
million tokens and found 481,131 MWE instances.
Table 3 summarizes the exact number of MWES
that we found, grouped by their class type.

The matching was performed on the lemma
level constraining the search with clitic matching.
Gaps are restricted only to noun-phrases at this
time, as mentioned above. The output of this
process follows the Inside Outside Beginning
(IOB) annotation scheme. In fact, the output files
are based on the same input AMIRAN files,
enriched with O, B/I-MWE tags as found by the
pattern-matching algorithm. Figure 1 shows how a
complete sentence, containing a MWE, is
annotated by the pattern-matching algorithm.

<table>
<thead>
<tr>
<th>MWE Category Type</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>VVC</td>
<td>576</td>
</tr>
<tr>
<td>VNC</td>
<td>64,504</td>
</tr>
<tr>
<td>VPC</td>
<td>75,844</td>
</tr>
<tr>
<td>NNC</td>
<td>316,393</td>
</tr>
<tr>
<td>ANC</td>
<td>23,814</td>
</tr>
</tbody>
</table>

Table 3 – Annotated MWES by class

5 **Evaluation**

The annotations are manually evaluated by a native
speaker of Arabic. We sub sampled the corpus and
examined each MWE instance that is identified by
the pattern-matching algorithm. Table 4 shows our
findings. Each row represents one category type.
The middle column shows the number of instances
evaluated, followed by the number of unique
MWE types. In the last column, the number of
correct instances as it was examined in context, is
reported. The correctness of an instance is determined by its context. Remember that MWEs are not only matched statically; generic words, gaps and inflections may cause the pattern-matching algorithm to annotate expressions with an MWE type, incorrectly.

<table>
<thead>
<tr>
<th>Word</th>
<th>Lemma</th>
<th>POS</th>
<th>NER</th>
<th>MWE</th>
</tr>
</thead>
<tbody>
<tr>
<td>swlAnA</td>
<td>swlAnA</td>
<td>NN</td>
<td>I-OR</td>
<td>O</td>
</tr>
<tr>
<td>:)</td>
<td>:)</td>
<td>PUNC</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>AlAmHAd</td>
<td>&lt;it-iHAd</td>
<td>NN</td>
<td>B-GP</td>
<td>O</td>
</tr>
<tr>
<td>AlAwrbwy</td>
<td>Awrbwy-iy</td>
<td>NN</td>
<td>I-GP</td>
<td>O</td>
</tr>
<tr>
<td>w+</td>
<td>wa+</td>
<td>CC</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>wASnTn</td>
<td>wASnTnuN</td>
<td>NNP</td>
<td>B-GP</td>
<td>O</td>
</tr>
<tr>
<td>ysEyAn</td>
<td>saEY-a</td>
<td>VBPMD3</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>l+</td>
<td>li+</td>
<td>IN</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>AyjAd</td>
<td>AyjAd</td>
<td>NN</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>Alyp</td>
<td>li-y-ap</td>
<td>NNFS</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>l+</td>
<td>li+</td>
<td>IN</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>wqf</td>
<td>waqof</td>
<td>NN</td>
<td>O</td>
<td>O</td>
</tr>
<tr>
<td>ATIAq</td>
<td>Talaq</td>
<td>NN</td>
<td>O</td>
<td>B-MW</td>
</tr>
<tr>
<td>Al+</td>
<td>Al+</td>
<td>DET</td>
<td>O</td>
<td>I-MW</td>
</tr>
<tr>
<td>nAr</td>
<td>nAr</td>
<td>NN</td>
<td>O</td>
<td>I-MW</td>
</tr>
</tbody>
</table>

Figure 1 – Annotated sentence example

<table>
<thead>
<tr>
<th>MWE Type</th>
<th>Evaluated Instances</th>
<th>Correct Instances</th>
</tr>
</thead>
<tbody>
<tr>
<td>VVC</td>
<td>111 (2 types)</td>
<td>2</td>
</tr>
<tr>
<td>VNC</td>
<td>157 (34 types)</td>
<td>154</td>
</tr>
<tr>
<td>VPC</td>
<td>161 (32 types)</td>
<td>125</td>
</tr>
<tr>
<td>NNC</td>
<td>155 (26 types)</td>
<td>154</td>
</tr>
</tbody>
</table>

Table 4 – Evaluation Results

The evaluation set is relatively small. Nevertheless, one can see that in most cases the annotations are correct. For the VNC, the pattern matching algorithm achieves an accuracy of 98%, for VPC, we get an accuracy of 77.6%, and NNC we achieve an accuracy of 99%. It is worth noting that NNCs are the only category that employs the gapping. The VVC category contains only a few MWE types, in the sampled set we evaluated 111 instances of merely two different types from which, one was constantly identified incorrectly by the algorithm and it constitutes the majority of the instances (109 instances).

6 Conclusions

In this paper we have introduced a list of MWEs in Arabic. The MWEs are enriched with morphological information that was carefully assigned to every word. A large part of the Arabic Gigaword 4.0 was deterministically annotated using a pattern-matching algorithm, considering morphological variations as expressed by Arabic and some potential gaps. A sample of the corpus was manually evaluated with encouraging results. Building both resources is a first step toward our research in the field of Arabic MWEs. Classifying the level of idiomaticity of the part of the MWE classes is one direction we are currently exploring.

Acknowledgment

This paper is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR0011-12-C-0014. Any opinions, findings and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of DARPA.

References


