Int J Speech Technol (2016) 19:259–267
DOI 10.1007/s10772-016-9340-x

Improving Arabic morphological analyzers benchmark

Younes Jaafar[1] · **Karim Bouzoubaa**[1] · **Abdellah Yousfi**[2] · **Rachida Tajmout**[1] · **Hakima Khamar**[3]

Received: 10 November 2015 / Accepted: 2 April 2016 / Published online: 19 April 2016

Abstract The various tools dedicated to Arabic natural language processing have undergone significant development during recent years. Among these tools, Arabic morphological analyzers are of great importance because they are often used within other projects that are more advanced such as syntactic parsers, search engines, machine translation systems, etc. Thus, researchers are forced to make a decision concerning which morphological analyzer to use in their research projects, and this task is very difficult since there are many criteria to take into account. In order to facilitate this choice, we considered the problem of benchmarking morphological analyzers in a previous work by proposing a solution that allows returning a set of metrics of each analyzer that are: accuracy, precision, recall, F-measure and the execution time. In this article, we present two new major improvements to our solution: the establishment of the first version of our corpus that is dedicated to the evaluation of morphological analyzers, as well as the introduction of a new metric, which combines all metrics related to results as well as the execution time of the analyzers.

Keywords Arabic morphological analyzers · Benchmark · Standard corpus

✉ Younes Jaafar
jayounes@yahoo.fr

Karim Bouzoubaa
karim.bouzoubaa@emi.ac.ma

Abdellah Yousfi
yousfi240ma@yahoo.fr

Rachida Tajmout
tajmoutrachida@yahoo.fr

Hakima Khamar
khamarhaki@gmail.com

[1] Mohammadia School of Engineers, Mohammed Vth University, Rabat, Morocco

[2] FSJES, Mohammed Vth University, Rabat, Morocco

[3] Faculty of Letters and Human Sciences, Mohammed Vth University, Rabat, Morocco

1 Introduction

Digital Arabic content has grown increasingly during the last decades (texts, videos, etc). Processing this huge volume of information and taking advantage of it requires the development of tools and programs that are dedicated to Arabic natural language processing (ANLP). Today, several tools for ANLP are already developed such as search engines (Hattab et al. 2009), machine translation systems (Champsaur 2013), opinion mining and sentiment analysis (Pang and Lee 2008), etc. Most of these tools use morphological analyzers in order to analyze the structure of words (Al-Sughaiyer and Al-Kharashi 2004). These analyzers main objective is to decompose words into morphemes and provide several morphological information such as stem, root, pattern, affixes, etc. For example, the word «أكتب» (‘Oktb’) may be analyzed as follows: vowelized = “أَكْتَبَ”, stem = “كتب”, pattern = “فَعَلَ”, root = “كتب”, prefix = “أ: همزة الاستفهام”. Among these morphological analyzers we mention BAMA (Buckwalter 2002a, b), Alkhalil (Alkhalil Morpho Sys 2013; Boudlal et al. 2011), MADAMIRA (Pasha et al. 2014), etc.

Thus, it is important for a researcher in ANLP to make an optimal choice when selecting a morphological analyzer to use in his/her research project. To help researchers making this choice, we have developed a solution (Jaafar and Bouzoubaa 2014) that allows comparing Arabic

morphological analyzers using known metrics such as: accuracy, precision, recall, and F-measure.

Given their importance, researchers have worked over the time to improve the accuracy of morphological analyzers from a low rate to reach nowadays 90–95 % and even more in some cases. Nevertheless, digital content in general, and the Arabic one in particular is increasing to reach a high rate over the last 10 years. The latest statistics[1] confirm this, allowing Arabic Internet users to set the 4th position among all users in the world. To follow this trend, new concepts and fields have emerged such as big data. When processing such large digital content, it becomes very important to take into account not only the accuracy but also the execution time. For example, instant translation of a live speech requires very fast tools for processing and translating texts. We are then shifting from a world where the accuracy of results is the only metric that matters to a world where the combination of accuracy and execution time of results matter.

Moreover, returning separate metrics can generate confusion for researchers when choosing a morphological analyzer. Indeed, the accuracy of the results and the execution time are two metrics that vary in opposite directions. The first should increase while the second should decrease in order to get a good result. This will cause a problem of comparison in the case where we have two analyzers that return such opposite metrics. For example, an analyzer X returns an accuracy rate of 80 % with an execution time of 1960s, and an analyzer Y returns an accuracy rate of 60 % but with an execution time of 8 s, that is to say one of them is more accurate but slower, the other one is less accurate but faster. Is such case, selecting the best analyzer is not obvious since the metrics are disproportionate.

Thus, our objective in this paper is to present a new global metric that combines metrics related to the accuracy of results as well as the execution time of each analyzer. This new metric will allow researchers to make the optimal choice in contexts where the execution time is crucial even if the metrics returned by morphological analyzers are disproportionate. Thus, researchers could make their decisions according to single metrics such as accuracy, execution time, etc. or according to one new global metric. We will also present the first version of our corpus dedicated specifically to the evaluation of Arabic morphological analyzers. Indeed, available evaluation corpora are annotated according to word contexts, which is not appropriate for this kind of benchmarking where morphological analyzers return all possible analyses for each word without taking the context into account.

It should be noted that the construction of benchmarks for the different Natural Language Processing (NLP) tools has already been addressed in several works. For example, the benchmark of Arabic stemmers was considered by Sawalha and Atwell (2008) and Al-Kabi et al. (2011). However, they do not offer reusable generic solutions that can be used to benchmark new stemmers, they just provide an evaluation for some specific stemmers. There are also other more advanced benchmark and evaluation solutions such as U-compare (Kano et al. 2010). However, they do not offer tools for benchmarking Arabic morphological analyzers. Therefore, developing this kind of benchmark is useful to the ANLP community.

All tools and resources used in this article are available freely to researchers who would like to test them or who would like to use them in their research projects (http://sibawayh.emi.ac.ma/safar).

The rest of this paper is organized as follows. The next section presents the most used Arabic morphological analyzers within the ANLP community. In Sect. 3, we present the most widely used Arabic annotated corpora, then we present our evaluation corpus dedicated to the benchmark of Arabic morphological analyzers. In Sect. 4, we present some usual metrics used for the evaluation of results, then we present in Sect. 5 our new global metric that combines metrics related to results as well as the execution time of each morphological analyzer. In Sect. 6, we present experiments and results of the benchmark of the three most used Arabic morphological analyzers. Finally, we present the conclusion and our plans for future work in Sect. 7.

2 Arabic morphological analyzers

Arabic Morphological analyzers identify the structure of a given word and other linguistic units. Among these analyzers we find the following most widely used ones:

BAMA: Is a morphological analyzer for Arabic, written in Perl by Buckwalter (2002b). BAMA uses three components in order to analyze a text: the lexicon, the compatibility tables and the analysis engine. AraMorph (Brihaye 2003) is a Java version of BAMA.

Alkhalil: Is a morphological analyzer for Arabic written in Java (Boudlal et al. 2011). This analyzer identifies all possible solutions of a word and establishes a list of morphological features of these solutions (type, pattern, root, POS…). The output can be in either HTML or CSV format.

MADA + TOKAN: is a system of morphological analysis and disambiguation for Arabic , written in Perl for UNIX systems only. Its main objective is to return as much linguistic information as possible about each word in an Arabic text, thereby, reducing or eliminating any ambiguity surrounding the words. It also provides tokenization with several schemas. The output is a simple formatted text file.

[1] http://www.internetworldstats.com/stats7.htm.

MADAMIRA: Is a system for morphological analysis and disambiguation of Arabic (Pasha et al. 2014), written in Java. MADAMIRA combines two previously used systems for Arabic processing, MADA and AMIRA (Diab 2009). Input and output texts can be supplied as plain text or in XML.

There are also other morphological analyzers such as ElixirFM (Smrž 2007), Sebawai (Darwish 2002), etc.

We selected three among the most widely used morphological analyzers within the ANLP community, namely: BAMA, Alkhalil and MADAMIRA to serve as example for our tests and experiments with the benchmark solution throughout this article. Due to licensing restrictions, we used the version of MADAMIRA that is packaged with Aramorph (Brihaye 2003) as database. It should be noted that the results of MADAMIRA could be enhanced by the use of SAMA database (Graff et al. 2009) instead of Aramorph. These three analyzers were used in many other projects. For example, BAMA was largely used in several other projects either as morphological analyzer or as database. It is used as a lexicon resource by MADA +TOKAN (Habash et al. 2009), it was also used to annotate the International Corpus of Arabic (Alansary et al. 2007). Concerning Alkhalil, it was selected as the best morphological analyzer in the competition that was organized by the Arab League Educational, Cultural Scientific Organization (ALECSO) in 2010. It is also used in other projects such as (Chennoufi and Mazroui 2014; Koulali and Meziane 2013; Wali et al. 2014). MADAMIRA was also used by many other projects such as (Hassan et al. 2014).

3 Evaluation corpora for Arabic morphological analyzers

3.1 Presentation of some evaluation corpora

In order to perform the benchmark process, the results returned by morphological analyzers must be compared to results of an annotated evaluation corpus. This corpus should be verified manually by linguists to maximize its precision and provide confidence in its data. It should be also annotated without taking the context of the words into account in order to have all possible analyses of each word as do morphological analyzers. In addition, this corpus should contain a maximum amount of morphological information such as root, pattern, stem, POS, prefixes, suffixes, etc.

In general, there is a huge lack of these kinds of corpora within the ANLP community. The available ones are either not free, don't have a significant amount of morphological information (tags), are not checked manually by linguists, or are annotated according to the context of words, etc. Thus, these corpora are not suitable for benchmarking morphological analyzers. Among these corpora, we find the following ones:

« Gold Standard of Arabic » (Sawalha, Gold Standard of Arabic, n.d.): is a free evaluation corpus. It is considered by its authors as a standard for the evaluation of Arabic morphological analyzers, because it contains an important amount of information relevant to the morphology such as stem, root, affixes, POS, etc. It consists of the chapter 29 of the holy Qur'an, « sourhat Al-ankaboot ». This corpus contains 976 words (575 unique words without diacritics) which are analyzed according to their context, annotated and checked by Arabic linguists.

« Quranic Arabic Corpus » (Dukes 2010; Dukes and Habash 2010): is an online annotated linguistic resource with multiple layers of annotation including morphological segmentation, part-of-speech tagging, syntactic analysis using dependency grammar. It consists of the holy Qur'an annotated according to the context of words. The main morphological information returned by this corpus is the root, lemma, part-of-speech, prefixes and suffixes.

« International Corpus of Arabic » (Alansary et al. 2007): a corpus that is planned to contain 100 million words of Modern Standard Arabic. The collection of samples is selected from a wide range of sources. This corpus is analyzed by BAMA (Buckwalter 2002b), the suitable analysis for each word is then chosen according to its context.

3.2 Towards a new corpus for Arabic morphological analyzers evaluation

Given the lack of suitable corpora dedicated to the evaluation of Arabic morphological analyzers, it was necessary to set up a new corpus in order to address this need.

Our corpus consists of 100 words carefully chosen from the holy Qur'an to represent a set of several possible cases of morphological analysis of words (according to prefix/suffix combinations). Each word has several morphological analyses (1628 analyses in total). It is annotated, manually checked by linguists and available for the general public.[2] The words of our corpus are distributed as shown in Table 1.

We have annotated this corpus in two steps:

Automatic step Since we deal with a huge number of analyses of each word, it was difficult to arrange for linguists to annotate all the words manually. To remedy this, we have used Arabic morphological analyzers as an intermediate step to produce all eventual possible analyses of each word.

[2] http://sibawayh.emi.ac.ma/safar/resources/100words_corpus.xml.

Table 1 Distribution of 100 words of our corpus dedicated to the evaluation of arabic morphological analyzers

Words	Verbs	Nouns	Particles
Without affixes	5	5	15
1 prefix + 0 suffix	5	5	9
2 prefix + 0 suffix	5	5	3
3 prefix + 0 suffix	1	1	0
0 prefix + 1 suffix	4	5	5
0 prefix + 2 suffix	4	2	0
1 prefix + 1 suffix	6	10	0
2 prefix + 2 suffix	4	1	0

Manual step In order to correct the results of the automatic step, two Arabic linguists have checked all analyses of each word in order to validate the analyses, correct, delete or add new ones.

This corpus is structured according to the standard format proposed by ALECSO that is considered to be more compatible with the nature of the Arabic language. In addition, our corpus covers a large number of morphological features: diacrization, stem, type, part-of-speech, prefixes, suffixes, pattern, root, case, mood, number, gender, definiteness, transitivity, augmented and unaugmented. Moreover, the corpus is in XML format, which allows more flexibility while exploiting it compared to other corpora that are in simple text format. Table 2 gives an overview.

For example, the word « **أحسب** » (‘OHsb’) has 47 manually checked analyses, each one of these analyses has several tags. For example, the analysis with the id = “1” for the word « **أحسب** » has the following morphological information: vowelized = “**أَحسَبُ**” (which represents the word with diacritics), stem = “**حسب**”, pattern = “**فَعَلُ**”, root = “**حسب**”, etc.

Table 2 Xml format of our evaluation corpus

```
<?xml version="1.0" encoding="UTF-8"?>
<morphology_analysis total_words="100">
  <word id="1" value="أحسب" total_analysis="47">
    <analysis
        id="1" vowled="أَحسَبُ" stem="حسب"
        pattern="فَعَلُ" root="حسب"
        pos="مفرد مذكر مرفوع في حالة الاضافة"
        number="مفرد" gender="مذكر" mood="مرفوع"
        caze="في حالة الاضافة" type="اسم جامد"
        prefix="[أ: همزة الاستفهام]" suffix="#"
    />
    <!--- 46 other analyses for the word أحسب… -->
  </word>
</morphology_analysis>
```

4 Metrics of performance used to evaluate morphological analyzers

The Arabic morphological analyzers benchmark process consists of returning a list of metrics on which researchers can rely to measure the performance of a given morphological analyzer. To measure this performance, we use the usual evaluation metrics[3]: the precision, recall, accuracy and F-measure. These metrics are calculated for each word returned by the morphological analyzer using the parameters presented in Table 3 and which are adjusted to the context of morphological analyzers:

For morphological analyzers, the parameters presented above are calculated as follows for each word “W”:

$$TP_W = |X_W \cap Y_W|$$

where

X_W: Analyses of a morphological analyzer for the word “W”. It is the set of all analyses returned by the morphological analyzer for a given word “W”. This set may contain correct analyses as well as eventual incorrect ones.

Y_W:: Analyses of the evaluation corpus for the word “W”. It is the set of all possible correct analyses for the word “W” that must be returned by a morphological analyzer after analyzing that word. This set do not contain any incorrect analyses since it is manually checked by linguists.

The intersection of X_W and Y_W gives TP_W (True Positive) which is the total number of correct analyses returned by the morphological analyzer for the word “W”. For example, if a morphological analyzer returns 10 analyses for a word “W” and 3 of them are incorrect, then the TP_W would be equal to 7. That is to say, TP_W represents correct analyses returned by the morphological analyzer for the word “W”.

$$FP_W = |X_W - (X_W \cap Y_W)|$$

The subtraction of TP_W from X_W gives FP_W (False Positive) which is the total number of incorrect analyses returned by the morphological analyzer for the word “W”. In contrast to TP_W, the FP_W represents incorrect analyses. For example, if a morphological analyzer returns 10 analyses for a word “W” and 3 of them are incorrect, then the FP_W would be equal to 3.

$$TN_W = 0$$

For morphological analyzers, $TN_W = 0$ (True Negative) because they are expected to return only the correct

[3] http://en.wikipedia.org/wiki/Precision_and_recall.

Table 3 Parameters used to calculate the benchmark metrics

	Positive (P)	Negative (N)
True (T)	Total of correct analyses returned by the morphological analyzer	Total of incorrect analyses identified by the morphological analyzer
False (F)	Total of incorrect analyses returned by the morphological analyzer	Total of correct analyses not returned by the morphological analyzer

analyses. In other context where other types of tools should classify elements into categories, the TN_W may be different to zero. However, in the context of morphological analyzers, only correct analyses are supposed to be returned, there is no classification of elements. That is to say, the total number of incorrect analyses identified by the morphological analyzer for a word "W" is equal to zero.

$$FN_w = |Y_w - (X_w \cap Y_w)|$$

The subtraction of TP_W from Y_W gives FN_W (False Negative) which is the total number of correct analyses but not returned by the morphological analyzer for the word "W". For example, if a morphological analyzer returns 10 analyses for a word "W" assuming that 3 of them are incorrect, and the word "W" has in reality 20 possible correct analyses in the evaluation corpus, then the FN_W would be equal to 13. That is to say, besides the 7 correct analyses returned by the morphological analyzer, there are 13 others that are not returned.

Using the parameters TP_W, FP_W, TN_W and FN_W we can calculate our main metrics as follows:

$$Precision = \frac{\sum TP_w}{\sum TP_w + \sum FP_w}$$

The precision of analyses returned by a morphological analyzer expresses the total number of correct analyses compared to the total number of all analyses returned by the morphological analyzer. The precision can be less than 100 % even if the analyzer returns all possible correct analyses of all words, which means that, in addition to the correct analyses, it returns additional analyses that are incorrect ($\sum FP_w > 0$), and can be equal to 100 % even if it does not return all possible correct analyses for that word, which means that all its results are correct ($\sum FP_w = 0$). For example, if a morphological analyzer returns 10 analyses for a word "W" assuming that 3 of them are incorrect, then the precision would be equal to 7/10 (70 %).

$$Recall = \frac{\sum TP_w}{\sum TP_w + \sum FN_w}$$

The recall of analyses expresses the total number of correct analyses returned by a morphological analyzer for this word compared to the total number of all analyses (in the evaluation corpus) that should be returned normally by the morphological analyzer. In contrast to the precision, the recall may be equal to 100 % even if the analyzer returns additional incorrect analyses (since it does not take into account the FP_W parameter). For example, if a morphological analyzer returns 13 analyses for a word "W" assuming that 3 of them are incorrect, and the word "W" has in reality 10 possible correct analyses in the evaluation corpus, then the recall would be equal to 10/10 (100 %).

$$Accuracy = \frac{\sum TP_w + \sum TN_w}{\sum TP_w + \sum TN_w + \sum FP_w + \sum FN_w}$$

Since we have $TN_w = 0$ in the case of morphological analyzers, the accuracy is then reduced to:

$$Accuracy = \frac{\sum TP_w}{\sum TP_w + \sum FP_w + \sum FN_w}$$

The accuracy of analyses returned by a morphological analyzer expresses the proportion of the analyses that are false. In contrast to the precision and recall, the accuracy is equal to 100 % only if the morphological analyzer returns all possible correct analyses for all words ($\sum FN_w = 0$), and in addition to that, there are no additional analyses that are incorrect within its results ($\sum FP_w = 0$). If the accuracy is equal to 100 %, this means that the morphological analyzer is perfect.

$$F - measure = 2 \times \frac{precision \times recall}{precision + recall}$$

The F-measure combines both precision and recall in one single metric. It can be interpreted as a weighted average of the precision and recall.

When evaluating an NLP tool, researchers often use these usual metrics that are related to the accuracy of results. However, Arabic data in the digital world has become so large that it becomes impossible to neglect the execution time of tools. Thus, our benchmark solution returns the metrics related to the accuracy of results (precision, recall, etc.) as well as the execution time of each analyzer.

However, returning separate metrics makes the selection of the best analyzer more difficult for researchers. Indeed, the accuracy of the results and the execution time are two metrics that vary in opposite directions. This causes a

problem of comparison in the case where we have two analyzers that return such disproportionate metrics.

5 New metric to improve our benchmark for morphological analyzers

To remedy the problem of disproportionate metrics, we propose a new global metric called G_M-score (for $Global_{Morphology}$-score) that combines the accuracy, the execution time as well as the morphological information returned by morphological analyzers.

It should be noted also that there are no standards concerning tags returned by morphological analyzers. For example, some analyzers may return the pattern of words but some other analyzers may not. Nevertheless, some common tags appear in most analyzers results. Thus, we organized the morphological tags into two categories:

- Required tags: the common tags that the morphological analyzer should take into account, otherwise it will be penalized while benchmarking. These tags are: diacritized word, stem, type (verb, noun or particle), root, pattern, prefixes and suffixes.
- Additional tags: These are all the other tags returned by the morphological analyzer, such as gender, number, mood, case, etc.

Thus, the G_M-score (Global Score for Morphology) is calculated as follows:

$$G_M - score = \frac{\sum T_w}{Accuracy + Tags_i + \alpha Tags_a}$$

where:

- T_w: The time taken by the morphological analyzer to analyze the word "W".
- $Accuracy_w$: The accuracy of analyses returned by the morphological analyzer for the word "W".
- $Tags_i$: The number of required morphological tags that are taken into account by the analyzer.
- $Tags_a$: The number of additional morphological tags that are taken into account by the analyzer. The α is used to increase or decrease the weight of the parameter $Tags_a$, this is because $Tags_a$ can be considered less important than $Tags_i$ by most researchers. The value of α can be set to a given value before starting the benchmark.

For this new metric, we chose three types of parameters that represent the different sides of strength or weakness of a morphological analyzer, that are: the execution time, the accuracy of results and the number of morphological tags returned. We have grouped these parameters into two categories: parameters whose values must decrease in order to be better, such as the execution time, and parameters whose values must increase in order to be better, such as the accuracy. Our idea is to put all parameters of the first group in the numerator and all parameters of the second group in the enumerator. This justifies the presence of the parameter T_W alone in the numerator.

We used accuracy instead of precision or recall because these latter do not take into consideration all the results returned (or that must be returned) by the morphological analyzer. On one side, the precision focuses on the results returned by the analyzer by neglecting those that are not returned while they must be. On the other side, the recall focuses on the correct set of results that must be returned by the analyzer by neglecting the incorrect results of the analyzer. The accuracy takes into account correct results of the analyzer, its incorrect results and correct results that are not returned by the analyzer. This means that the accuracy is the best metric to use in order to reflect the overall relevance of results of an analyzer. In addition to that, we should also take into account the richness of the morphological information (tags) returned by analyzers because; an analyzer that returns more morphological information will logically take more execution time than an analyzer with less information. It will not be fair to favor one over the other depending only on its execution time or only on its richness, but both at once. Thus, our new metric combines all the other different metrics and will allow researchers to make the optimal choice even if the metrics returned by morphological analyzers are disproportionate.

It should be noted that our G_M-score metric is considered better when its value tends to zero and worse when it tends to a big number. It can be applicable to all other languages. Moreover, researchers can also rely on this metric even if the execution time does not matter for them. In such case, they just have to fix the execution time to a constant, for example '1'.

6 Experiments and results

In order to give concrete examples of using our solution of the benchmark, we have selected three morphological analyzers, namely: BAMA, Alkhalil and MADAMIRA that are among the most used within the ANLP community. These three morphological analyzers have been compared using our corpus of evaluation. These experiments were performed on a computer having the following characteristics: CPU = Core 2 Duo @2.13 GHZ, RAM = 4GO, Operating System = Win7, 32bits.

Table 4 presents the global metrics for the morphological analyzers after the benchmark, namely: the number of analyzed words, the number of words not analyzed, the execution time, the number of required tags, the number of

Table 4 Results of comparing Bama, Alkhalil and Madamira using our evaluation corpus

Metrics	BAMA	Alkhalil	MADAMIRA
Total analyzed words	96	100	96
Total words not analyzed	4	0	4
T1: Execution time (including time of loading files)	6.82 s	16.63 s	81 s
T2: Execution time (excluding time of loading files)	5.6 s	10.07 s	0.2 s
Precision (%)	31.83	78.41	20.13
Recall (%)	6.5	79.84	10.08
Accuracy (%)	5.71	65.45	7.2
F-measure (%)	10.8	79.12	13.43
Required tags ($Tags_i$)	4	7	5
Additional tags ($Tags_a$)	2	7	12
G_M-score (using T1)	0.63	0.21	4.45
G_M-score (using T2)	0.52	0.13	0.01

Table 5 List of tags returned by each analyzer

Analyzers	Required tags	Additional tags
Alkhalil	Vowelized | stem | pattern | root | type | prefix | suffix	POS | number | gender | mood | caze | transitive | impartial
BAMA	Vowelized | stem | type | prefix | suffix	Lemma | gloss
MADAMIRA	Vowelized | stem | type | prefix | suffix	POS | number | gender | mood | case | lemma | BW | gloss | person | aspect | voice | state

additional tags, precision, recall, accuracy, F-measure and finally the new G_M-score metric.

As indicated in Table 4, the three analyzers analyze all or most of words: 100 analyzed words by Alkhalil and 96 analyzed words by both BAMA and MADAMIRA. However, they have a large difference in terms of execution time. If we include the loading files time, BAMA completes the analysis in 6.82 s; Alkhalil takes 16.63 s while MADAMIRA takes 81 s. However, if we exclude it, MADAMIRA jumps to the first place with 0.2 s, followed by BAMA with 5.6 s and finally Alkhalil with 10.07 s.

The results show also that Alkhalil achieves 65.45 % accuracy, followed by MADAMIRA with 7.2 % and finally BAMA with 5.71 %. This large difference between Alkhalil and the two other analyzers is mainly due to the number of analyses returned by each one of them. Alkhalil returns much more analyses and matches most of the morphological analyses present in the evaluation corpus. For example, for the word "أنزل" ('Onzl') which have 50 manually checked morphological analyses in the evaluation corpus, Alkhalil returns 57 possible analyses while BAMA and MADAMIRA return only 7 analyses each.

Concerning the morphological information returned by the analyzers, Alkhalil returns all required tags (7 tags), followed by MADAMIRA and BAMA (5 tags). For the additional tags, MADAMIRA returns more information with 12 tags, followed by Alkhalil with 7 tags and finally BAMA with 2 tags. Table 5 gives an overview of these tags:

Concerning our new global metric G_M-score that combines all the above metrics, Alkhalil gets 0.21, followed by BAMA with 0.63 of G_M-score and MADAMIRA with 4.45. These results of GM-score include the time of loading files of each analyzer. However, if we exclude it, MADAMRA gets the best rate with 0.01, followed by Alkhalil with 0.13 and finally BAMA with 0.52. It should be reminded that the G_M-score metric is considered better when its value tends to zero and worse when it tends to a big number.

7 Conclusion

In this article, we presented the benchmark of Arabic morphological analyzers. We described the two major improvements we have made to our previous work on benchmarking, namely: the establishment of the first version of our annotated corpus which is verified by linguists and dedicated to the evaluation of morphological analyzers. Our corpus consists of 100 words carefully chosen from the holy Qur'an to represent several possible cases of morphological analysis of words. Each word in the corpus has all its possible morphological analyses (1628 analyses in total) since it is annotated out of its context. The establishment of this corpus was required given the huge lack in such free evaluation corpora dedicated to the benchmark of morphological analyzers. The available ones

are either not free, don't have a significant amount of morphological information (tags), are not checked manually by linguists, or are annotated according to the context of words, etc. Consequently, these corpora are not suitable for benchmarking morphological analyzers.

The second improvement consists of the introduction of a new evaluation metric called G_M-score that combines metrics related to the accuracy of analyzers as well as the execution time. It should be noted that many researchers in ANLP community dot not take execution time into account while developing their tools. Indeed, the execution time is an important element for many other researchers, it affects the decision of using a tool in their projects or not. Moreover, the accuracy of the results and the execution time are two metrics that vary in opposite directions for better results. This causes a problem of comparison in the case of two analyzers that return a disproportionate metrics: good accuracy with worse processing time or inferior accuracy with better execution time. Our proposed G_M-score allows researchers to make the best possible choice of the morphological analyzer to use in their projects that consider execution time as an important metric. It should be noted that our G_M-score metric is considered better when its value tends to zero and worse when it tends to a big number.

We have chosen the three most used Arabic morphological analyzers, namely: BAMA, Alkhalil and MADAMIRA in order to compare their results and give a concrete example of our solution. The three analyzers have been evaluated and compared using our evaluation corpus. Results show that Alkhalil reaches respectively 78.41, 79.84 and 65.45 % of precision, recall and accuracy; BAMA achieves respectively 31.83, 6.5 and 5.71 %, while MADAMIRA reaches respectively 20.13, 10.08 and 7.2 %. Concerning the G_M-score rates, results show that Alkhalil, BAMA and MADAMIRA reach respectively 0.21, 0.63 and 4.45 when taking the time of loading files into account (which makes Alkhalil in the first position). However, if the time of loading files is not taken into account, they reach respectively 0.13, 0.52 and 0.01 (which makes MADAMIRA in the first position).

In the future, we plan to deal with new and more advanced metrics such as the complexity of analyzer algorithms. We also intend to further enrich our evaluation corpus by adding new words.

References

Alansary, S., Nagi, M., & Adly, N. (2007). Building an international corpus of Arabic. *7th international conference on language engineering*, (p. np.). Cairo.

ALECSO. (n.d.). Retrieved December 23, 2014, from مواصفات نظام التحليل الصرفي في اللغة العربية: http://www.alecso.org.tn/images/stories/OULOUM/MOHALLILAT%20SARFIA_DAMAS_2009/022%202%20SPECIFICATIONS.pdf.

Al-Kabi, M., Al-Radaideh, Q., & Akkawi, K. (2011). Benchmarking and assessing the performance of Arabic stemmers. *Journal of Information Science, 37*(2), 111–119.

Alkhalil Morpho Sys. (2013). Retrieved April 23, 2015, from Alkhalil Morpho Sys: http://sourceforge.net/projects/alkhalil/.

Al-Sughaiyer, I. A., & Al-Kharashi, I. A. (2004). Arabic morphological analysis techniques: A comprehensive survey. *Journal of the American Society For Information Science and Technology, 55* (3), 189–213. Retrieved from Imad Al-Sughayer and Ibrahim Al-Kharashi. "Arabic morphological Analysis Techniques: a comprehensive Survey". Computer and Electronics.

Boudlal, A., Lakhouaja, A., Mazroui, A., Meziane, A., Ould Abdallahi, O. B., & Shoul, M. (2011). Alkhalil Morpho Sys: A morphosyntactic analysis system for Arabic texts. *Proceedings of ACIT'2010.*

Brihaye, P. (2003). *AraMorph*. Retrieved April 23, 2015, from AraMorph: http://www.nongnu.org/aramorph/english/index.html.

Buckwalter, T. (2002a). *Arabic morphology analysis*. Retrieved April 23, 2015, from QAMUS: http://www.qamus.org/morphology.htm.

Buckwalter, T. (2002b). Buckwalter Arabic morphological analyzer version 1.0.

Champsaur, C. (2013, January). La traduction automatique : Un outil pour les traducteurs? *The Journal of Specialised Translation, 19*, pp. 19–28.

Chennoufi, A., & Mazroui, A. (2014). Apport de la deuxième version de l'analyseur Alkhalil Morpho Sys dans la voyellation automatique des textes Arabes. *5th international conference on Arabic language processing (CITALA* 2014*)*, (pp. 223–230). Oujda.

Darwish, K. (2002). Building a shallow Arabic morphological analyzer in one day. *Proceedings of the ACL-2002 workshop on computational approaches to semitic languages*, (pp. 47–54). Retrieved from https://aclweb.org/anthology.

Diab, M. (2009). Second generation tools (AMIRA 2.0): Fast and robust tokenization, POS tagging, and base phrase chunking. *Second international conference on Arabic language resources and tools*, (pp. 285–288). Cairo.

Dukes, K. (2010). *The Quranic Arabic corpus*. Retrieved April 23, 2015, from Quranic Arabic Corpus. http://corpus.quran.com.

Dukes, K., & Habash, N. (2010). Morphological annotation of Quranic Arabic. *Language resources and evaluation conference (LREC)*. Malta. Retrieved from https://aclweb.org/anthology.

Graff, D., Maamouri, M., Bouziri, B., Krouna, S., Kulick, S., & Buckwalter, T. (2009). *Standard Arabic morphological analyzer (SAMA) version 3.1*. Linguistic Data Consortium LDC2009E73.

Habash, N., Rambow, O., & Roth, R. (n.d.). *MADA + TOKAN software suite*. Retrieved April 23, 2015, from MADA + TOKAN: http://www1.cs.columbia.edu/rambow/software-downloads/MADA_Distribution.html.

Habash, N., Rambow, O., & Roth, R. (2009). Mada + tokan: A toolkit for Arabic tokenization, diacritization, morphological disambiguation, pos tagging, stemming and lemmatization. *Proceedings of the 2nd international conference on Arabic language resources and Tools (MEDAR)*, (pp. 102–109). Cairo.

Hassan, Y., Aly, M., & Atiya, A. (2014). Arabic spelling correction using supervised learning. *Proceedings of the EMNLP 2014 workshop on Arabic*, (pp. 121–126). Doha.

Hattab, M., Haddad, B., Yaseen, M., Duraidi, A., & Shmais, A. A. (2009). Addaall Arabic search engine: Improving search based on combination of morphological analysis and generation considering semantic patterns. *The 2nd international conference on Arabic language resources & tools*, (pp. 159–162).

Jaafar, Y., & Bouzoubaa, K. (2014). Benchmark of Arabic morphological analyzers: Challenges and solutions. *Intelligent systems: Theories and applications (SITA-14)*, (pp. 1–6). Rabat.

Kano, Y., Dorado, R., McCrohon, L., Ananiadou, S., & Tsujii, J. (2010). U-Compare: An integrated language resource evaluation platform including a comprehensive UIMA resource library. *Proceedings of the seventh international conference on language resources and evaluation (LREC 2010)*, (pp. 428–434).

Koulali, R., & Meziane, A. (2013). Experiments with Arabic topic detection. *Journal of Theoretical and Applied Information Technology, 50*(1), 28–32.

Pang, B., & Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval, 2*(1–2), 1–135.

Pasha, A., Al-Badrashiny, M., Diab, M., El Kholy, A., Eskander, R., Habash, N., & Roth, R. M. (2014). MADAMIRA: A fast, comprehensive tool for morphological analysis and disambiguation of Arabic. *LREC'14*, (pp. 1094–1101). Reykjavik.

Sawalha, M., & Atwell, E. (2008). Comparative evaluation of Arabic language morphological analysers and stemmers. *International conference on computational linguistics—COLING*, (pp. 107–110). Retrieved from https://aclweb.org/anthology.

Sawalha, M. (n.d.). *Gold Standard of Arabic*. Gold standard for evaluating Arabic morphological analyzers. Retrieved April 23, 2015, from http://www.comp.leeds.ac.uk/sawalha/goldstandard.html.

Smrž, O. (2007). ElixirFM: Implementation of functional Arabic morphology. *Proceedings of the 2007 workshop on computational approaches to Semitic languages: common issues and resources* (pp. 1–8). Stroudsburg: Association for Computational Linguistics.

Wali, W., Gargouri, B., & Ben Hamadou, A. (2014). A system for evaluating the content of LMF Arabic dictionaries. *5th international conference on Arabic language processing (CITALA 2014)*, (pp. 159–167). Oujda.