Abstract— Arabic language is a unique language because of its pronunciation of the word written. The formation of more than two Arabic letters will translate to a different meaning. This paper presents an alternative translator from Arabic word to English word by using an equation for the formation of the Arabic letters. The morphological model of Arabic language is used in order to use the equation to put diacritics according to Arabic grammatical rules. A system is then developed based on the morphological model. The input of the system is the Arabic word. The system uses morphological Arabic Natural Language Processing (ANLP) and translates the Arabic word into English word. The output of the system shows percentage of the successful translated words. We tested our system by using 11 Arabic words. The results show that 9 out of 11 Arabic words (81.81%) were successfully translated by conducting the process of diacritics in the morphological model of Arabic language. We also perform a quality assessment to calculate the efficiency and effectiveness metrics based on the ANLP developed. Based on the result, the system can be used as an alternative translator from Arabic word to English word.

Keywords— natural language processing (NLP); morphological model; efficiency and effectiveness metrics;

I. INTRODUCTION

The Arabic language is both challenging and interesting. It is interesting due to its history, the strategic importance of its people and the region they occupy, as well as its cultural and literary heritage. It is also a challenging language because of its complex linguistic structure. Historically, classical Arabic has remained unchanged, clear and functional for more than fifteen centuries [1]. The Classical Arabic represents the language spoken by the Arabs more than fourteen centuries ago, while Modern Standard Arabic is an evolving variety of Arabic with constant borrowings and innovations proving that Arabic reinvents itself to meet the changing needs of its speakers [2]. At the regional level, there are as many Arab dialects as there are members of the Arab league. The diglossic nature of the Arabic language is discussed in [3]. Therefore, the Arabic natural processing language applications must deal with several complex problems pertinent to the nature and structure of the Arabic language. For example: ( عِلم science, عَلَم flag, عَلِمََ taught, عِلَم knew). Arabic is written from right to left. Like Chinese, Japanese, and Korean, there is no capital letter in Arabic. In addition, Arabic letters change shape according to their position in the word. Modern Standard Arabic does not have orthographic representation of short letters which require a high degree of homograph resolution and word sense disambiguation. Like Italian, Spanish, Chinese, and Japanese, Arabic is a pro-drop language, that is, it allows subject pronouns to drop [4]. A language that is subject to recoverability of deletion as a natural language, Arabic has much in common with other languages such as English [5].

II. RELATED WORK

Arabic Natural Language Processing (ANLP) has been an active research for many researchers such as [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12]. Some of the researches are discussed in this section. The summarization of techniques used is shown in Table 1.

<table>
<thead>
<tr>
<th>Author</th>
<th>Technique</th>
<th>Purpose of Technique</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Barlow, 2006) [6]</td>
<td>MonoConc</td>
<td>Translation from Arabic into English using the classical style</td>
</tr>
<tr>
<td>(Badrashiny a et al., 2009) [2]</td>
<td>Concordance</td>
<td>Translation from Arabic into English using the classical style</td>
</tr>
<tr>
<td>(Shaalan et al., 2009) [3]</td>
<td>aConCorde</td>
<td>Translation from Arabic into English using the classical style</td>
</tr>
<tr>
<td>(Attia, 2008) [1]</td>
<td>ANIMA NLP</td>
<td>Add the diacritics to Arabic texts</td>
</tr>
</tbody>
</table>

MonoConc [6] is a concordance programme. This Windows tool is very easy to use as it can initiate concordance searches for words and phrases immediately. MonoConc offers functionality and flexibility through a variety of configurable options. The program used morphological rules for Arabic text analysis in Arabic window, but with one major drawback: the concordance output is presented on the screen.
backwards. In other words, in the middle the KWIC (keyword-in-context) is normal, then the context that is supposed to come after the key word appears before it and that which precedes the key word follows it. The concordance output can be saved to a text-only file, and when opened in a text editor (e.g. MS Arabic Word), the text appears in the right order [6].

aConCorde [3] is originally developed for native Arabic concordance and support right-to-left languages. The program is written in Java and ran on any platform that uses Java Runtime Environment. However, the translator is released early with shortcomings noted by the designer; the results ignore diacritics for Arabic language characters [3].

Nabil [7] discusses the sources of raw Arabic lexical semantics knowledge. The based knowledge is usually organized so that the semantic fields/word senses are the primary keys that recall the terms belonging to them [7].

III. THE PROPOSED ALGORITHM

The proposed algorithm is formed based on the algorithm proposed in [11]. Table 2 shows the differences of the proposed algorithm with the old algorithm proposed in [11]. The algorithm uses a word to build a natural word in the Arabic language. When choosing the word, the system separates each Arabic character and gives Diacritics (Fatha, Kasra, and Damma) for each character. A word from the dictionary is entered by the user and the letters were examined after separating each character to duplicate them in the base of female.

If it duplicates other word, and the word was not necessary added with diacritical marks and did not exist, diacritical marks were added and collected in one word, using the ASCII code as shown in Figure 1. The Fatha, Kasra, and Damma would become three words and then, it would test one’s consent if the words were incorrect. However, if it is compatible with the words and the translation is correct, it provides the meaning of its own.

Table 2: The Different Steps for the proposed New Algorithm

<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Build the m-grams $W$</td>
<td>Build the all Word</td>
</tr>
<tr>
<td>Sort (S)</td>
<td>Separation of letters</td>
</tr>
<tr>
<td>Compute $\alpha$ Alpha</td>
<td>Add ASCII code</td>
</tr>
<tr>
<td>Discard $W^m_1$ and corresponding $\alpha$</td>
<td>Sum the letters</td>
</tr>
<tr>
<td>Text with Diacritics</td>
<td>Word with Diacritics</td>
</tr>
<tr>
<td></td>
<td>Translate</td>
</tr>
</tbody>
</table>

Based from Figure 1, the pos tagging of word uses morphology adopted from [1] and is shown in Table 3.

Table 3: Pos Tagging Adopted from [1]

<table>
<thead>
<tr>
<th>Cat.</th>
<th>Mnemonic</th>
<th>Meaning in English</th>
<th>Meaning in Arabic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start of word marker</td>
<td>SOW</td>
<td>Start-Of-Word marker</td>
<td>بداية الكلمة</td>
</tr>
<tr>
<td>Padding string</td>
<td>Padding</td>
<td>Padding string</td>
<td>حشو</td>
</tr>
<tr>
<td>Features of noun and verb prefixes</td>
<td>NullPrefix</td>
<td>Null prefix</td>
<td>لا سابق</td>
</tr>
<tr>
<td></td>
<td>Conj</td>
<td>Conjunctive</td>
<td>عطف</td>
</tr>
<tr>
<td></td>
<td>Confirm</td>
<td>Confirmation by Laam</td>
<td>لام التوكيد</td>
</tr>
<tr>
<td></td>
<td>Interrog</td>
<td>Interrogation by Hamza</td>
<td>همزة الاستفهام</td>
</tr>
</tbody>
</table>

The overall process of the translation is shown in Figure 2.
Based from Figure 2, the usability metrics are also used to analyze the efficiency of the proposed ANLP.

Table 2 shows the different steps between the old and new algorithms. They share the same process; build, compute, and sort, but the old one uses m-grama, secma, and pot table, to add all the diacritical marks for Arabic letters. As for the new algorithm, simple equation is added to the three signs to form Arabic language characters, collect letters, and check all the letters and compare them to the database to prevent repetition.

1. Initialize by creating a stack SS holding nodes
2. Pop up the surface node.
3. If length of path in W is D, exit with the most likely path dw.
4. Expand W to nodes of next column with scores calculated from equations (4.1), (4.2), (4.3), and (4.4), and push them into SS.
5. Go to step 2.

In Figure 3, where the path algorithm and sequences work by creating road constructing “Stack Let SS” and when conducting a call “pop up” to be examined along the track in the word, and if that process diacritics was correct by calling a set of equations that had been working in every one on the basis of their work and then return to the second step if there was a second call.

1. SELECT ID, Arabic word, etranslation1, etranslation2
2. FROM Table1 (Dictionary)
3. IF length of path in W is D, exit with the most likely path dw
4. WHERE (Arabic word like ‘value1%’ +? ‘value2%’ +? ‘value3 %’) OR
5. (Arabic word like ‘value1%’ + mid (?, 1,1)+ value2%’+mid(?1,1)+? ’value3’)
6. Put them into SS.

The prototype for ANLP was successful developed. This section discusses the experimental results based on the ANLP and its usability assessment metrics.

IV. RESULTS AND DISCUSSION

Table 4 shows the process of inserting the words taken from the Qur’an and the literature in Arabic by conducting the process of diacritics, translation and watching the proportion of words that were successfully translated with high accuracy, where it was in the Qur’an 69.23%, 11 words and 9 with high accuracy, and Arab literature used 7 words and 4 of which had been translated with high accuracy; 30.77%, and while everything between the percentages was a success of the program, which compiled 100%.

Figure 5 shows the final results for ANLP, whereby, when writing the word and pressing the direct translation, the words appeared with the diacritics selected and translated each word by its meaning.

Figure 2: The framework for the proposed ANLP

Figure 3: Path Algorithm for ANLP

Figure 4: Search Algorithm for ANLP

Figure 5: The results for the ANLP
B. Usability Metric Analysis

After the analysis result was obtained from the ANLP to English, the usability metric using QUIM technique [13] was used to find the result of the analysis. The result provided in the main program interface as part of the program helped the user to draw conclusions, and identify relevant factors and usability criteria and standards, and then, the completion of the process of ANLP.

QUIM is hierarchical and it decomposes usability into factors and finally into specific metric [13]. In this study, 2 factors were used to test the ANLP for the usability. These 2 factors are efficiency and effectiveness.

1. **Efficiency** is the relationship between the inputs of the production process and the outputs resulting from this process.

\[
EE = \frac{S_{\text{Essential}} - \text{RWT}}{S_{\text{Essential}}} \times 100
\]

where:
- EE : Essential Efficiency
- \(S_{\text{Essential}}\) : Number word use in programs
- RWT : rate word translation

Equation 1 was used for analyzing the efficiency of ANLP. The result was 95% based on Figure 6. The efficiency was good, however if it falls below 95%, the efficiency would be bad.

![Figure 6: The Efficiency of ANLP](image)

Figure 6 shows the effort of ANLP where \(S_{\text{Essential}}\) was calculated. A number of words used in the system were subtracted from the number of correct results.

2. **Effectiveness** is the system’s ability to achieve the goals. It calculated cost increase as an indicator of the efficiency of the system. Quantity production of the modules was with less effort.

Productivity Evaluation:

The process of evaluation of productivity is an important part of a program to improve productivity since they give indications of the performance of the current, and reveal weaknesses and areas of improvement required, as demonstrated by the results achieved by the program.

The assessment tools were:
- Value-added
- Rates of productivity
- Productivity standards

Assume that we used 5000 words, including 420 words correct and over 3 years of age to measure the efficiency of the system, the system’s effectiveness would be as follows:

\[
\text{Effectiveness} = \frac{\text{RWT} \times \text{Years of use of the system}}{\text{number all word use in system}} \times 100\%
\]

![Figure 7: The Effectiveness of ANLP](image)

Based on Figure 7 the assumption of the words uses are 5000 words, including 420 words correct and over 3 years of age to measure the efficiency of the system. The system outputs successfully where the proportion was 60%, which demonstrated the need for the user of the system.
V. CONCLUSION

The method used statistical approaches have eased the process to solve the problems of Arabic language processing although it required the construction of a large database to be interconnected for simultaneous solving of the problem of diacritics language when added to the Arabic word. In this paper, it is shown that the solution to this dilemma is not difficult, but there is a need to use tactics statistical morphological to resolve how to add Arabic diacritical marks and put in a word and then translated by the type of diacritical marks. Besides, research and algorithms are new kinds of explanations and new types of data and new algorithms for dealing with such data, and the complex tasks of the Arabic language natural can be processed and resolved with new artistic style. A set of data from the Quran and Arabic Language Literature gave much contribution in completing the project.

REFERENCES


BIOGRAPHY

Abdulrahman Ahmed Al Zend was born in Baghdad in 1979 and worked in the Ministry of Science and Technology as a programmer for the design of e-government sites as well as a lecturer at the Iraqi University to teach computer science. He completed his master study at the Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia (UTHM) under the supervision of Rosziati Ibrahim.

Rosziati Ibrahim is with the Software Engineering Department, Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia (UTHM). She obtained her PhD in Software Specification from the Queensland University of Technology (QUT), Brisbane and her MSc and BSc (Hons) in Computer Science and Mathematics from the University of Adelaide, Australia. Her research area is in Software Engineering that covers Software Specification, Software Testing, Operational Semantics, Formal Methods, Data Mining and Object-Oriented Technology.