Journal of Theoretical and Applied Information Technology
31st March 2019. Vol.97. No 6

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

ENHANCING ARABIC NAMED ENTITY RECOGNITION USING PARALLEL TECHNIQUES

[1]ZIADOON A. OTAIWI, [2]MOHAMMED A. OTAIR, [3]KHALDOON A. AOTAIWE, [4]AHMAD ODAT, AND [5]FERAS ALMASHAKBAH

[1]Atilim University, Software Engineering Department
[2]Amman Arab University, Faculty of Computer Science and Informatics
[3]Iraqi Ministry of Education, General Directorate of Anbar Education - Department of Education Heet
[4]Irbid National University, Faculty of Science and Information Technology
[5]Zarqa University, Information Technology Faculty
[1]ziadoonotaiwi@gmail.com ,[2]otair@aau.edu.jo , [3]khaldooncom@gmail.com , [4]aodat@yahoo.com ,
[5]fmashakbah@zu.edu.jo

ABSTRACT

Named entities recognition systems (Proper Names) are used in the development of many natural language processing applications. There is a paucity of published research in the field of identifying the named entities from texts written in Arabic. This is due to the fact that the Arabic language has a specificity regarding the complexity of spelling and morphology, which is an obstacle to the development of a technique to identify the names of the Arabic entities or the so-called Arabic Named Entity Recognition system (ANER). This paper presented the experiments conducted to identify the appropriate technique to design a robust and reliable system for identifying Arabic entities. For this purpose, this study focuses on the most common state-of-art in the field of identification of Arabic named entities, then a comparison was made between five of the most famous tools that interested in identifying the Arab entities, after that, integrated each of two tools together to get 10 different parallel techniques. The results of the comparison between the tools showed that Rosette achieved the best results followed by Madamira, while it was the worst performance results in the gate tool and for hybrid systems, the R-F (combining Rosette and Farasa) achieved the best performance with better accuracy than individual tools.

Keywords: *Parallel Techniques, Arabic Named Entity, Named Entity Recognition, Tool*

1. INTRODUCTION

Named entity recognition have been studied over the latest few years, especially in names that refer to the field of natural language processing, which has a lot of related topics such as information extraction, question answering systems, information retrieval techniques, text categorization, and summarization. Unfortunately, the Arabic language has not received much attention in this field, unlike other languages, in which many researchers expanded their studies. A computer system for grammatical analysis was applied in Arabic sentences, so they relied on the Definite Clause Grammar (DCG) in Prolog mainly in this system.

The subject of morphology was introduced in the Arabic language, with a general framework for dealing with it was found to solve some of the problems related to names [1]. To limit the costs of the process, many researchers have united the affixes, which had a positive effect on solving the problems of manipulating [2]. Some others have created a robust system, where it can only save the roots of the verbs without their derivatives, then once stored these verbs; the system automatically finds all the words related to this verb (derivations).

One of the problems facing the process of getting the proper names is to find the vagueness in these names and their classification, whether it is ambiguous in structure or in the semantics. Some have worked on a particular system, which can elicit the names of people and other nouns based on the Wall Street Journal [3]. NOMEN is one of the systems used to find generic names in a text which is based on bootstrapping to extract a variety of textual sequences and to identify their patterns [4].

This paper tries to enhance the quality of extracting Arabic Named Entity from Arabic text files by using common tools (Farasa, API Entity Extraction, Madamira, Gate, Rosette) in Arabic Named Entity Recognition. Most of the problems that facing (ANER) in the common tools most notably not returning all real names or retrieving some words as entities although their true classification is not entities. There are some problems in some of the tools that used to recognize the named entities in the text files, for example in Farasa tool, there is a robust in not retrieving errors, but there is a weakness in retrieving the named entities from the text files.

This paper has organized as follows: Section 1: Starts this study with a general introduction, which contains the research problem. Section 2: Describes the recent related work. It reviews a comprehensive survey of the institutes of modern studies. Section 3: Gives a detailed explanation of five of the most famous tools available online. Section 4: Provides a description of the proposed approach. Section 5: Presents a set of test results that have applied in used technique as well as the results of each tool individually. Finally, Section 6: Concludes the paper with final remarks, in addition to many open issues for future work.

2. RELATED WORK

Many studies have been dealt with the identification of Arabic named entities, some of these studies have relied on the open-source tools, while some researchers have designed their own tool. But in general, all techniques were used to extract named entities from the text. This section introduces the mostly related works of Arabic named entity recognition.

The research in [5] addressed a topic at the core of Named Entity Recognition (NER). The purpose of their research was to defeat the problem of the rules of NER systems. So that, the researchers tried to solve them to improve the efficiency of such systems and develop them, then, they can be updated continuously, using the parallel NER software to extract recognition decisions to find the flaws for exploring new grammar rules, which in turn increased the opportunity to give better more precise results.

Drawing on ACE Newswire data as a source for the extraction and analysis of new language rules for the place, person and the naming organization, the researchers reconstructed each new rule to categorize them into two groups: the geographical index for every named entity, where Part-of-Speech labels either common or proper nouns. Their studies resulted in the development of 14 new patterns, which were expressed in the form of grammar rules and assessed based on the coverage area. Their experiments were carried out using one version of the NER system called POS. Based on their studies and experiences; it was found that the NERA 2.0 system helped greatly overcome the problem as it was developed in coverage of people names by 69.93%, places by 57.09% and finally the names of organizations by 54.28%.

The researchers in [6] have dealt with Named Entity Recognition (NER) in Arabic, in particular, Modern Standard Arabic (MSA). This study aimed to test the effect of word expressions and the inclusion in the Arabic language system of social media data. Note that, such topics have become the main concern in social media, which use both MSA and Dialectal Arabic (DA) with some changes by language, although somewhat similar, there is a slight difference, this was a cause of unfortunate performance when using MSA for NER and applied to DA. In their research, they

have provided gazetteers that were free of NER data systems. While, most NER systems depend mainly on gazetteers, which can make things more complex in dealing with social media for processing because of the poor coverage, in addition to the fact that these gazetteers are a great burden in terms of cost. After conducting many experiments and researchers, it was shown that their best score was 72.68%, which was about (2% to 3%) higher than DA-NER system-based gazetteers, then they were considered better than other systems, which improved NER performance more than others.

The study [7] have adopted this topic specifically to provide a new approach in this area. They have searched on linear kernels to find SVM-rank. Where they have adopted standards of accuracy and efficiency in assessing the performance of the segmenter. Specifically, two main tests were used in segmenter assessments: Information Retrieval (IR) and Machine Translation (MT). After several experiments and comparisons between Farasa and the state-of-the-art Arabic segmenter, it has been shown that Farasa and the state-of-the-art Arabic segmenter of both types (Stanford / MADAMIRA) were almost equal in performance and efficiency.

A new study on Farasa that has been conducted by [8]. Farasa is open-source and written with Java software. The aim of their study was to conduct new sets of tests considering the news articles in various fields. Their approach was mainly to assess the likely allowed divisions for a word in the Arabic language using linear kernels based primarily on SVM rank. Their tests were based on several key principles, including Likelihood of stems, Presence in lexicons containing valid stems and named entities, suffixes, prefixes and their combination, finally the underlying stem templates. The researchers presented their results for their new approach, Farasa, as one of the fastest and most accurate techniques in this field. According to their studies, Farasa has overtaken both the QATARA and MADAMIRA techniques, which referred to the state-of-the-art Arabic segmenter, in terms of accuracy and speed.

3. NAMED ENTITY RECOGNITION TOOLS

This Section presents a detailed explanation of the most common tools used in the extraction of Arabic named entities, whether the names of persons, places, organizations, etc. Then discuss each of the tools in terms of definition, characteristics, advantages, and structure, as well as an example of each tool using the text from data that used.

These tools are chosen based on using different techniques to identify the named entity. For example, the Gate tool uses machine learning techniques, while each of the rest is based on a set of grammatical and morphological rules for extracting named entities as well as the variety of pre-processing steps that used in each tool. In this paper, the studied tools are: Farasa, API Entity extraction, Madamira, GATE and Rosette

3.1 Farasa

Farasa is a tool related to the Arabic language which means insight and is defined for the precise and fast processing in Arabic texts using some entities. These tools contain a set of elements and most important are, Dependency Parser, POS tagger, and tokenization or segmentation module, finally Arabic text Discretize. There are two main NLP tasks used to measure the accuracy of performance which are Information Retrieval (IR) and Machine Translation (MT) [7].

2.2 Entity Extraction API

The named entity extraction tool sometimes called API, which does its work efficiently in short texts, which distinguishes it from other available services. The API works in many languages, including French, English, German, Arabic, Italian, Spanish, Portuguese, Russian and others. Using the API makes it easy to enrich used data, tag text in an automatic way, and extract Wikipedia entities [9].

3.3 Madamira

It is a new version of a combination of two tools which are MADA and AMIRA, it has benefited from some elements motivated from AMIRA and MADA [10]. This tool applies preprocessing steps to remove noise from the input text. The input text enters the Morphological Analysis process then expresses the usefulness of this process by finding all probable analyzes of each word in the text, regardless of the context.

3.4 Gate – General Architecture for Text Engineering

GATE is a tool designed for the development and deployment of specialized software modules and components in automatic natural language processing. This tool was developed in 1995, at the University of Sheffield [11]. It has been applied in various NLP projects; it has been used to extract data in several languages for use in different responsibilities and clients.

- GATE can help researchers and developers in several ways:
- Define the structure or organizational structure of language processing programs.
- Provide a framework with data storages, allowing the use and integration of the system.
- It is a suitable tool to service applications related to the processing of natural languages such as text recognition and summary and others [12].

Journal of Theoretical and Applied Information Technology
31st March 2019. Vol.97. No 6

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

3.5 Rosetten tities are the main component in textual data, whether these entities indicate the names of different people, products, organizations, dates or others. Rosette can detect these entities by using innovative techniques in machine learning statistical technique and text analysis. Rosette identifies, interprets, and provides the correct structure for the entities and the data in general. Rosette supports twenty different languages that were promoted, in addition, to give eighteen kinds of detected entities; also, rosette can be developed with high speed and continuously test. Getting special support for the industrial support.

4. THE PROPOSED TECHNIQUE

English still progresses more than Arabic in NER systems. Arabic is still in its first steps towards such systems. There are few sources and annotated data provided information in this area, and on the other hand, the impact of the lack of lexicons. Therefore, an emphasis has been placed on systems that have enhanced multilingual resources, including machine translation. In this research, the Arabic language has great importance to the development of ANER systems using a parallel technique.

The basic idea of the proposed technique is to employ a variety of the existing tools in the task of identifying the named entity; so that these tools are exposed to the same conditions and in the same environment. Based on the use of those tools, the most efficient recognition is chosen for the Arabic named entity. The fusion technique can be defined as integration between two or more ANER tools, this technique can be tested to determine the efficiency of the AENR technique obtained according to the level of accuracy and recall etc.

The proposed technique is built based on the fusion principle of more than one tool together since the fusion technique can be accomplished at the level of features or at the level of technique "tool". The fusion technique is used between more than one tools to reach the best parallel that determines the best results in identifying named entities. The proposed technique includes two main tasks: Firstly, build a corpus in a collection of Arabic magazine articles and draw the names of the Arabic names in a manual way. Second, use five famous tools to recognize the entities from the used corpus (as explained in Section three), and compare the results of these tools together to merge them at the results level. Figure 1 shows the detailed architecture of our system:

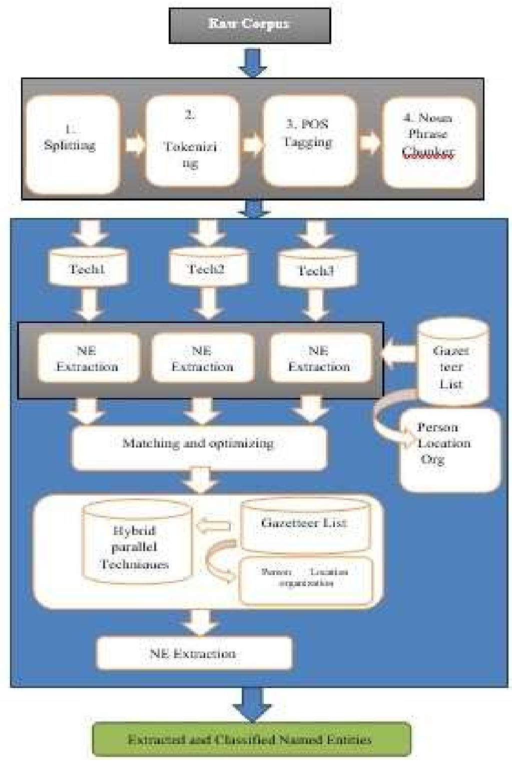

Figure 1. The Architecture Of The System

4.1: Implementation of Tools

In this step, the chosen tools were applied to the raw files and the results are stored sequentially for each tool with a serial number from 1 to 100 files.

4.2: Comparison of Automated Entities with Annotated Entities:

After obtaining the results of each tool, it is necessary to judge the accuracy of each of them in the possibility of identifying the names of Arabic entities. For this purpose, an automated framework is designed to compare the named entities obtained from each tool with the names of the extracted manually, i.e. the results of tool files are compared with results of annotation files.

This comparison is done by word-to-word matching and the use of the part of matching (not exact matching) is considered. As this match does not affect the prefixes and the suffixes, such as the article (ال التعريف) or suffix that indicates the plural and so on. For example, the matching between the University Hospital of Jordan and a hospital in Jordan University is 100% match, while in the exact matching result is 0, and this is not true.

Journal of Theoretical and Applied Information Technology
31st March 2019. Vol.97. No 6

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

4.3: Evaluation Performance Measures

In this step, the performance of each tool is evaluated based on the similarity results of the previous step. There are many widely used evaluation measures such as accuracy, recall, and precision. These measures, which are indeed useful to tune an NER system and they are considered as the standard evaluation techniques in this field, are mainly calculated based on four parameters, which are called the "confusion matrix". They are a true positive, true negative, false positive and false negative (TP, TN, FP, FN respectively). Where the value of TP is one when the tool can recognize the name of the entity. In other words, the name of the entity appears in both the tool and the manual annotation (i.e. the number of terms retrieved by the system and had to be returned). While the TN is "one" when the term is not a named entity and the tool does not retrieve it (i.e. the number of words that the system has not retrieved and should not have retrieved). FP is the term that the system identifies as named entities and in fact, does not belong to them. Finally, the named entities that the system failed to identify in the FN. Certainly, the total number of terms derived from these four parameters will be equal to the total number of raw text terms. This study used ten evaluation measures, three of them are widely used in related studies such as [6].

4.4: Fusion Technique

Fusion technique: this is the technique that depends on combining the results between two different techniques. This integration can be at the level of the feature extracting, the level of selection of features, the level of recognition based on these features or on the level the decision resulted from voting more than the technique. This technique is build based on the integration of the final results of the tools. For example, the Fusion technique, which is built from the Madamira tool and the Farasa tool: is the application of the union rule between the names of the information that resulted from all these tools. According to that, many matching occurrences have been conducted between each one of these tools with the rest tools to get the Fusion technique, which achieves the best results in terms of evaluation performance measures. The algorithm below summarizes the steps of the proposed technique:

Algorithm1: the main steps of the fusion technique
Input: different texts and articles from Jordanian newspapers
Output: Named entities

1. Begin
2. Store articles as text files.
3. Extract named entities manually depending on the experts (Annotators)
4. Store media names independently, for each text and rename them (Annotated files)
5. Apply original texts by tools.
6. Match the resulting file for each tool with the (Annotated files file).
7. Store the results of the confusion matrix and system performance metrics.
8. Implemented more than one scenario in the Fusion technique.
9. Determine the optimal scenario.
10. End

The next part explains the interface of the proposed technique, the figure bellow demonstrates the main interface of text matching step.

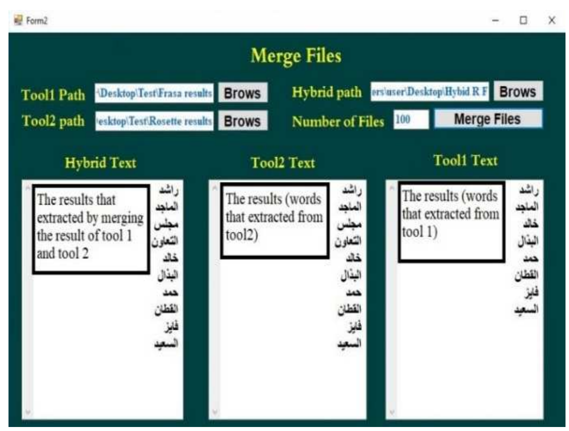

***Figure** (2.A): The interface of the proposed model.*

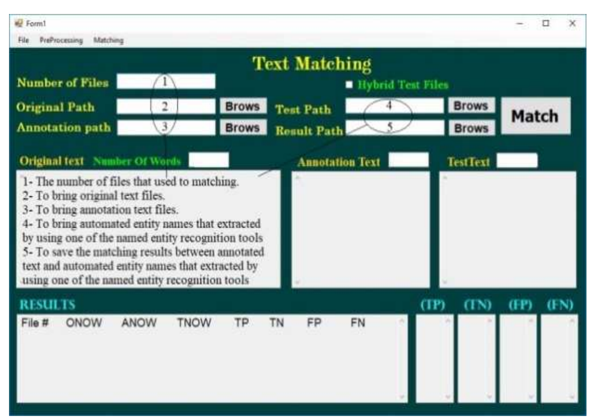

***Figure** (2.B): The interface of the proposed model*

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Figure (2.A) shows the matching results between annotated text and automated named entity that extracted using tools, the annotated text shows all words that categorize as a Named entity, while the test text represents the words retrieved by the tool as a named entity. For the result space, the first column represents the serial order and the second column represents all the words in the original text regardless of the word classification. The third column represents the column of the annotated words. The fourth column represents the results of the tool, while the rest of the columns represent true positive, true negative, false positive and false negative respectively.

Figure (2.B) shows the matching results between annotated text and automated named entity that extracted using a parallel technique. Parallel text displays the result of the merge all the words retrieved by both tools. Without duplicating as well as how to retrieve them from the places where they were stored.

5. The Result and Analysis

This Section describes the results of experiments that conducted to prove the efficiency degree of the proposed technique; the efficiency was measured using various techniques which explained in section 5.1. followed by the results and analysis in section 5.2.

5.1 Evaluation Performance Measures

Sensitivity (or Recall), Specificity, Accuracy, ROC, Precision and F-measure are considered as the main measures that mainly depend on the matrix of confusion.

- **Recall or sensitivity:** The following formula to be calculated:

$$\text{Sensitivity} = \frac{TP}{TP+FN} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (1)$$

- **Specificity:** The following formula to be calculated:

$$\text{Specificity} = \frac{TN}{TP+FN} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2)$$

- **Accuracy:** The following formula to be calculated:

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+TN+FN} \quad \ldots\ldots\ldots\ldots\ldots\ldots (3)$$

- **Precision:** The following formula to be calculated:

$$\text{Precision} = \frac{TP}{TP+FP} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (4)$$

- **F-measure:** The following formula to be calculated:

$$\text{F-measure} = 2 * \frac{recall*precision}{recall+precision} \quad \ldots\ldots\ldots\ldots\ldots (5)$$

$$\text{F-}\beta = \frac{(1+\beta^2)(recall*precision)}{(\beta^2)(recall+precision)} \quad \ldots\ldots\ldots\ldots\ldots (6)$$

Where the symbol (β) is most likely equal to 0.5, 1, or 2.

- **Matthews Correlation Coefficient (MCC):** The following formula to be calculated:

$$\text{MCC} = \frac{TP*TN-FP*FN}{\sqrt{(TP+FP)*(FN+TN)*(FP+TN)*(TP+FN)}} \ldots\ldots (7)$$

Fallout (False Positive Rate (FPR)): the proportions of non-relevant cases are identified as relevant ones incorrectly in information retrieval by the effort of review. Considering that the False Positive Rate plus the True Negative Rate is 100%, and False Positive Rate= the subtraction result of the True Negative Rate from 100%.

5.2 Experimental Results

This section contains the results of the experiments conducted to evaluate the proposed technique. Five different tools studied, and ten parallel techniques were conducted to try optimizing the recognition of named entity. The results are divided into two main experiments parts. In the first experiment part, this study applied five of the most common tools which are: Rosette, Madamira, Farasa, Entity, and Gate. While in the second experiment part, the previous five tools were used to create parallel, and thus resulted in ten parallel intersections without repetitions, which are: Rosette – Madamira (R-M), Rosette – Farasa (R-F), Rosette – Entity (R-E), Rosette – Gate (R-G), Madamira - Farasa (M-F), Madamira - Entity (M-E), Madamira - Gate (M-G), Entity - Farasa (E-F), Entity - Gate (E-G) and Farasa - Gate (F-G).

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Build a database:

In this step, 100 articles are collected in Arabic from various Jordanian newspapers and each of these articles is divided in the form of a text file with a" *.txt*" extension; so that the content of the file is textual data only (i.e. it does not contain images, hyperlinks, or others). Therefore, these articles are divided into three different sections: short-sized articles, medium-sized articles, and long-sized articles. Each of which consists around of 35 articles "text file"; so that the text file containing less than 60 words was considered a short file. The text file containing 60-80 words categorized as medium size files, and more than 81 words are considered long. Where the process of dividing these data has been done electronically and called these files "raw files". After getting the text files, we extracted manually the named entities. So that, the resulted100 files contain only the named entities generated from the raw data set, which is called Annotated files. When got the help of two Arabic language specialists to carry out the manual extraction of the names of entities. The first expert reviewed all the raw texts and extracted the names of the entities, then the second expert checked the step by the first expert to reduce the error rate to obtain the highest possible accuracy. Table 1 summarizes the statistics of the dataset.

***Table 1**: Description of the Dataset*

Dataset section	Number of texts	Number of words	Number of named entities
Short Texts	33	1736	276
Medium Texts	34	2323	335
Long Texts	33	5613	769

All experiments were conducted on the three types of datasets which are short, Medium and long text. This assessment has been carried out based on eight performance standards as explained in sub-section 5.1. Tables 2,3 and 4 show the evaluation results for the first experiment part, while tables 5,6 and 7 represents the results that were performed based on the second experiment part. Table 2 describes the evaluation results for the first experiment based on a short text that contains 1736 words in the text files, where rosette outperformed the other tools in terms of accuracy (Acc.=0.989).

Figure 3 shows the four evaluation performance measures which are: Accuracy, Precision, Recall and F-score with short text.

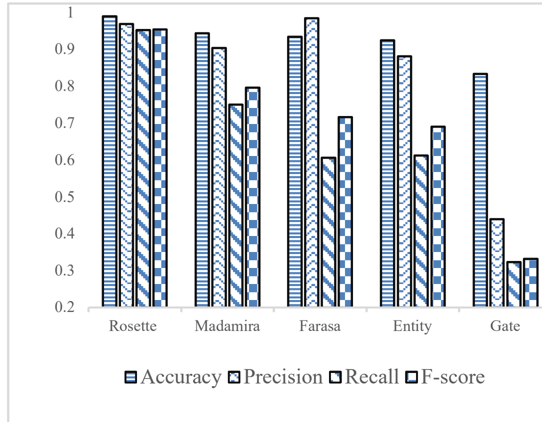

***Figure 3**: Implementation of Tools - Short Text*

Figure 4 shows the four-evaluation performance of the five tools based on the measures: Accuracy, precision, recall, and f-score with Medium text.

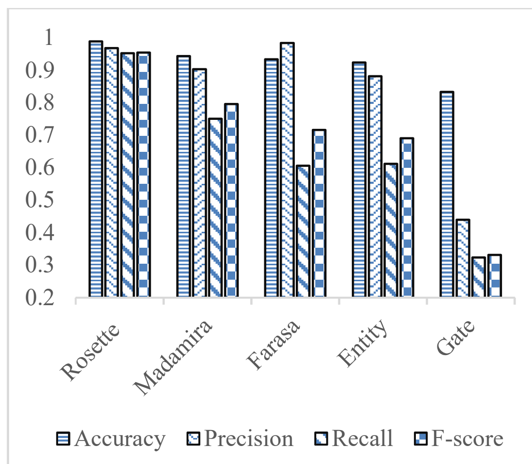

***Figure 4**: Implementation of Tools - Medium Text*

Figure 5 shows the four evaluation performance measures which are: Accuracy, Precision, Recall and F-score with Long text.

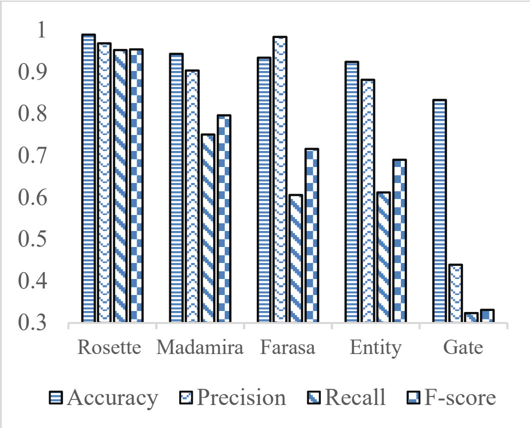

***Figure 5**: Implementation of Tools - Long Text*

5. CONCLUSION AND FUTURE WORK

This paper presented a comparison of five different types of ANER tools, as well as the use of the merging technique of these tools to enhance the identification of named entities in Arabic. The parallel techniques were built to improve the ability to identify the named entities. To investigate the results, ten common performance measures were used, which are Accuracy, Precision, Recall, F-score, Specificity, MCC, Fallout, Eβ0.5, Eβ1, and Eβ1.5. The F-Measure for Rosette-Farasa (R-F) was around 0.957%, 0.975% and 0.940% for short, Medium and long texts respectively. It's also concluded that the combined techniques affected seriously by the two tools. It's not necessary that two strong tools when resulted in a strong combination in terms of accuracy, for example, both of Madamira and Rosette resulted in high accuracy however their combination resulted in lower accuracy. Also, when comparing the values of the Recall scale, where some parallel systems achieved a great advantage compared to the best results achieved on the level of each tool alone, the results were about 0.985% (R-M), 0.979% (R-M), and 0.975(R-E) for short, Medium and long texts respectively. For future work, there is a plan to study the possibility of developing system performance using other techniques such as Ruled -based and parallel techniques with semantic information features.

REFERENCES:

[1] "Larsen-Freeman, D., & Long, M. H. (2014). An introduction to second language acquisition research. Routledge.".

[2] "Abuleil, S., & Alsamara, K. Enhance the Process of Tagging and Classifying Proper Names in Arabic Text. In Proceedings of the International Arab Conference on Information Technology (ACIT'2006) (p. 43).".

[3] "Kanan, T., Ayoub, S., Saif, E., Kanaan, G., Chandrasekarar, P., & Fox, E. A. (2015). Extracting named entities using named entity recognizer and generating topics using latent Dirichlet allocation algorithm for Arabic news articles. Department of Computer".

[4] "Yangarber, R., Lin, W., & Grishman, R. (2002, August). Unsupervised learning of generalized names. In Proceedings of the 19th international conference on Computational Linguistics-Volume 1(pp. 1-7). Association for Computational Linguistics.".

[5] "Oudah, M., & Shaalan, K. (2017). NERA 2.0: Improving coverage and performance of rule-based named entity recognition for Arabic. Natural Language Engineering, 23(3), 441-472.".

[6] "Zirikly, Ayah, & Diab, M. T. (2015). Named Entity Recognition for Arabic Social Media. 176-185.".

[7] "Abdelali, Ahmed, Darwish, K., Durrani, N., & Mubarak, H. (2016). Farasa: A Fast and Furious Segmenter for Arabic. In HLT-NAACL Demos, (pp. 11-16).".

[8] "Darwish, K., & Mubarak, H. (2016). Farasa: A New Fast and Accurate Arabic Word Segmenter. In LREC.".

[9] "Krug, M., Wiedemann, F., & Gaedke, M. (2014, April). Enhancing media enrichment by semantic extraction. In Proceedings of the 23rd International Conference on World Wide Web (pp. 111-114). ACM.".

[10] "Habash, N., Diab, M., Darwish, K., El-Hajj, W., Al-Khalifa, H., Bouamor, H., ... & El-Haj, M. (2017). Proceedings of the Third Arabic Natural Language Processing Workshop. In Proceedings of the Third Arabic Natural Language Processing Workshop.".

[11] "Cunningham, H. (2002). GATE, a general architecture for text engineering. Computers and the Humanities, 36(2), 223-254.".

[12] "Derczynski, L., Maynard, D., Rizzo, G., van Erp, M., Gorrell, G., Troncy, R., ... & Bontcheva, K. (2015). Analysis of named entity recognition and linking for tweets. Information Processing & Management, 51(2), 32-49.".

Table 2: Implementation Of Tools - Short Text

Tool	Acc.	Prec.	Recall	F-score	Spec.	MCC	Fallout	Eβ0.5	Eβ1.0	Eβ2.0
Rosette	**0.989**	0.968	**0.952**	**0.954**	0.995	**0.951**	0.005	**2.385**	**0.954**	**0.382**
Madamira	0.943	0.903	0.75	0.796	0.985	0.782	0.015	1.991	0.796	0.318
Farasa	0.934	**0.984**	0.606	0.716	**0.998**	0.726	**0.002**	1.79	0.716	0.286
Entity	0.924	0.881	0.612	0.69	0.989	0.684	0.011	1.725	0.69	0.276
Gate	0.833	0.439	0.323	0.331	0.942	0.272	0.058	0.827	0.331	0.132

Table 3 The evaluation results for the first experiment based on Medium text that contains 2323 words in the text files, where Rosette outperformed the other tools in terms of accuracy (Acc.=0.993), (Recall = 0.963) and (F-score=0.996). While Gate tool achieved the worst accuracy value (Acc. =0.849)

Table 3: *Implementation Of Tools - Medium Text*

Tool	Acc.	Prec.	Recall	F-score	Spec.	MCC	Fallout	Eβ0.5	Eβ1.0	Eβ2.0
Rosette	**0.993**	0.984	**0.963**	**0.971**	0.996	**0.969**	0.004	**2.428**	**0.971**	**0.389**
Madamira	0.941	0.912	0.676	0.763	0.988	0.749	0.012	1.908	0.763	0.305
Farasa	0.934	**0.99**	0.576	0.7	**0.999**	0.713	**0.001**	1.75	0.7	0.28
Entity	0.922	0.895	0.54	0.639	0.988	0.643	0.012	1.598	0.639	0.256
Gate	0.849	0.462	0.252	0.313	0.955	0.261	0.045	0.782	0.313	0.125

Table 4: Implementation Of Tools - Long Text

Tool	Acc.	Prec.	Recall	F-score	Spec.	MCC	Fallout	Eβ0.5	Eβ1.0	Eβ2.0
Rosette	**0.985**	0.944	**0.94**	**0.935**	0.993	**0.931**	0.007	**2.339**	**0.935**	**0.374**
Madamira	0.939	0.82	0.689	0.733	0.982	0.711	0.018	1.832	0.733	0.293
Farasa	0.931	**0.964**	0.529	0.652	**0.999**	0.67	**0.001**	1.629	0.652	0.261
Entity	0.928	0.881	0.631	0.705	0.986	0.696	0.014	1.762	0.705	0.282
Gate	0.863	0.462	0.307	0.331	0.96	0.292	0.04	0.828	0.331	0.132

Table 5 describes the evaluation results for the second experiment based on short text that contains 1736 words in all text files, where Rosette-Farasa (R-F) outperformed the other merged techniques in terms of accuracy (Acc.=0.989), (Recall = 0.966) and (F-score=0.957) while Entity-Gate (E-G) achieved the worst accuracy value (Acc. =0.889).

Table 5: Implementation Of Parallel Techniques For Short Texts

Parallel	Accuracy		Precision	Recall	F-score	Specificity	MCC	Fallout	Eβ0.5	Eβ1.0	Eβ2.0
	Value	Rank									
R-M	0.982	3	0.901	**0.985**	0.933	0.981	0.929	0.019	2.333	0.933	0.373
R-E	0.984	2	0.894	0.983	0.927	0.985	0.924	0.015	2.317	0.927	0.371
R-F	**0.989**	**1**	**0.958**	0.966	**0.957**	**0.993**	**0.954**	**0.007**	**2.392**	**0.957**	**0.383**
R-G	0.941	7	0.684	0.955	0.783	0.936	0.769	0.064	1.957	0.783	0.313
M-E	0.960	4	0.850	0.894	0.853	0.976	0.840	0.024	2.132	0.853	0.341
M-F	0.948	6	0.903	0.777	0.814	0.984	0.800	0.016	2.034	0.814	0.325
M-G	0.900	8	0.614	0.789	0.663	0.926	0.626	0.074	1.658	0.663	0.265
E-F	0.957	5	0.897	0.810	0.827	0.987	0.818	0.013	2.067	0.827	0.331
E-G	0.889	10	0.602	0.687	0.615	0.934	0.567	0.066	1.538	0.615	0.246
F-G	0.894	9	0.614	0.675	0.613	0.940	0.569	0.060	1.533	0.613	0.245

Journal of Theoretical and Applied Information Technology
31st March 2019. Vol.97. No 6

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Figure 6 shows the four evaluation performance measures of the parallel technique which are: Accuracy, Precision, Recall and F-score with short text.

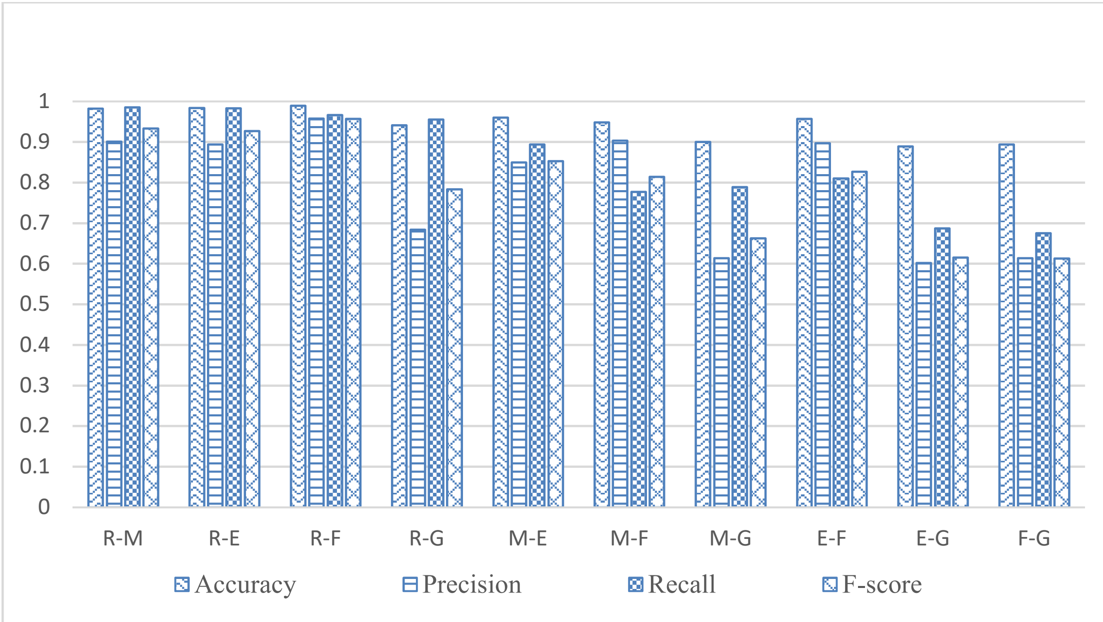

Figure 6: Implementation Of Parallel Techniques For Short Texts

Table 6 Explain the evaluation results for the second experiment based on Medium text that contains 2323 words in all text files, where Rosette-Farasa (R-F) outperformed the other merged techniques in terms of accuracy (Acc.=.994), (Recall = 0.976) and (F-score=0.975) while Entity-Gate (E-G) achieved the worst accuracy value (Acc. =0.899).

Table 6: Implementation Of Parallel Techniques For Medium Texts

Parallel	Accuracy		Precision	Recall	F-score	Specificity	MCC	Fallout	$E\beta_{0.5}$	$E\beta_{1.0}$	$E\beta_{2.0}$
	Value	Rank									
R-M	0.985	2	0.921	**0.979**	0.946	0.985	0.940	0.015	2.366	0.946	0.378
R-E	0.984	3	0.922	0.970	0.942	0.985	0.935	0.015	2.354	0.942	0.377
R-F	**0.994**	**1**	**0.978**	0.976	**0.975**	**0.995**	**0.973**	**0.005**	**2.439**	**0.975**	**0.390**
R-G	0.958	4	0.761	**0.979**	0.852	0.953	0.839	0.047	2.131	0.852	0.341
M-E	0.951	5	0.870	0.827	0.833	0.977	0.815	0.023	2.083	0.833	0.333
M-F	0.947	7	0.911	0.738	0.800	0.987	0.786	0.013	1.999	0.800	0.320
M-G	0.912	8	0.669	0.728	0.686	0.945	0.642	0.055	1.714	0.686	0.274
E-F	0.951	5	0.913	0.748	0.800	0.987	0.791	0.013	2.001	0.800	0.320
E-G	0.899	10	0.636	0.651	0.626	0.943	0.579	0.057	1.564	0.626	0.250
F-G	0.905	9	0.670	0.652	0.644	0.954	0.601	0.046	1.610	0.644	0.258

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Figure 7 shows the four evaluation performance measures of the parallel technique which are: Accuracy, Precision, Recall and F-score with Medium

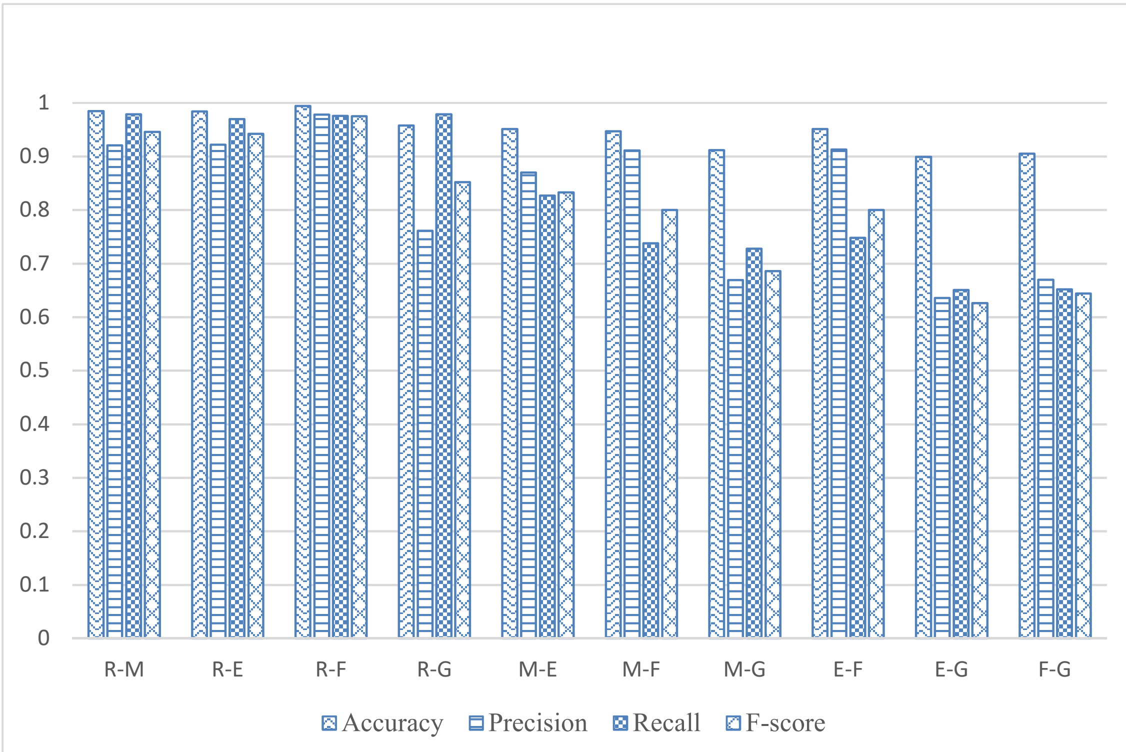

Figure 7: Implementation Of Parallel Techniques For Medium Texts

Table 7 Represent the evaluation results for the second experiment based on long text that contains 5613 words in all text files, where Rosette-Farasa (R-F) outperformed the other merged techniques in terms of accuracy (Acc.=0.988), (Recall=0.97) and (F-score=0.94) while Entity-Gate (E-G) achieved the worst accuracy value (Acc. =0.903).

Table 7: Implementation Of Parallel Techniques On Long Texts

Parallel	Accuracy		Precision	Recall	F-score	Specificity	MCC	Fallout	Eβ0.5	Eβ1.0	Eβ2.0
	Value	Rank									
R-M	0.976	3	0.852	0.958	0.893	0.979	0.886	0.021	2.233	0.893	0.357
R-E	0.978	2	0.877	**0.975**	0.917	0.980	0.909	0.020	2.292	0.917	0.367
R-F	**0.988**	**1**	**0.940**	0.953	**0.940**	**0.993**	**0.936**	**0.007**	**2.349**	**0.940**	**0.376**
R-G	0.956	4	0.735	0.954	0.815	0.957	0.806	0.043	2.039	0.815	0.326
M-E	0.946	7	0.798	0.832	0.796	0.970	0.775	0.030	1.991	0.796	0.318
M-F	0.950	5	0.839	0.756	0.782	0.983	0.762	0.017	1.954	0.782	0.313
M-G	0.915	8	0.635	0.731	0.657	0.947	0.621	0.053	1.643	0.657	0.263
E-F	0.947	6	0.872	0.771	0.797	0.984	0.781	0.016	1.992	0.797	0.319
E-G	0.903	10	0.633	0.671	0.615	0.948	0.579	0.052	1.537	0.615	0.246
F-G	0.910	9	0.658	0.615	0.611	0.960	0.574	0.040	1.528	0.611	0.244

Journal of Theoretical and Applied Information Technology
31st March 2019. Vol.97. No 6

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Figure 8 shows the four evaluation performance measures of the parallel technique which are: Accuracy, Precision, Recall and F-score with Long text

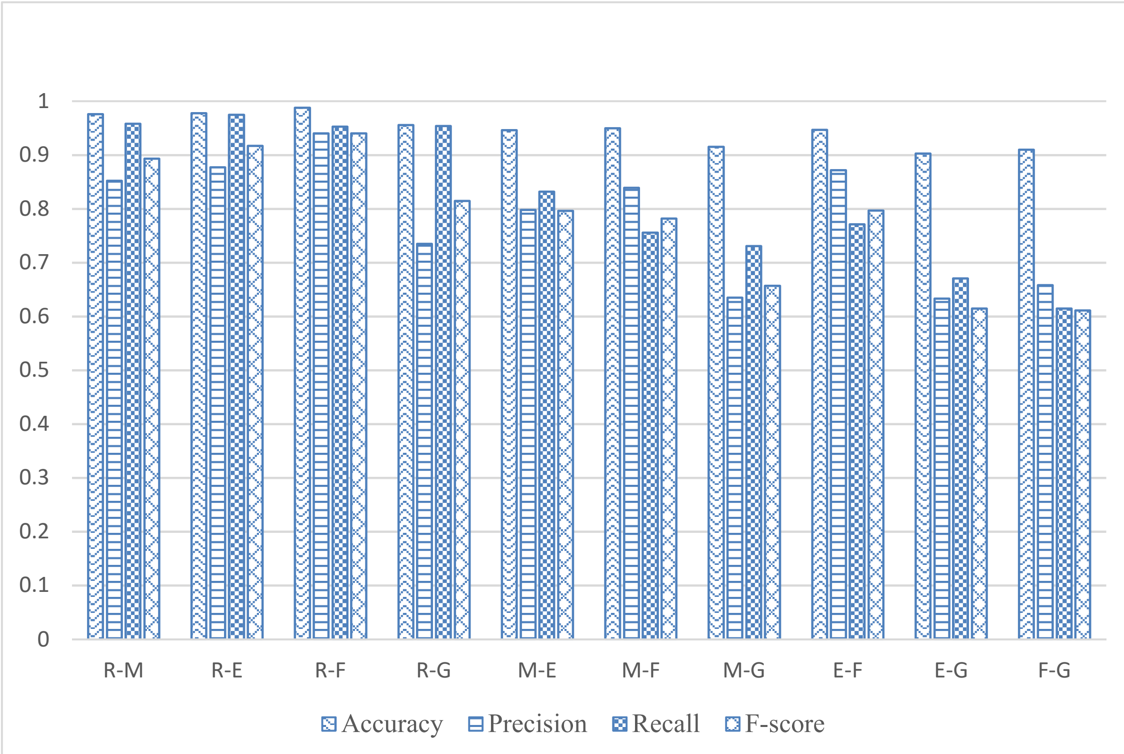

Figure 8: Implementation Of Parallel Techniques For Long Text

Figure 9 shows the result of the parallel technique in terms of the F-score, it shown that the best parallel technique was obtained from merging Rosette and Farasa with (0.975) with Medium texts, followed by the same combination with short texts, whereas the result provided that the combination of Farasa and Gate was the worst, especially with the long text (F-score=0.611).

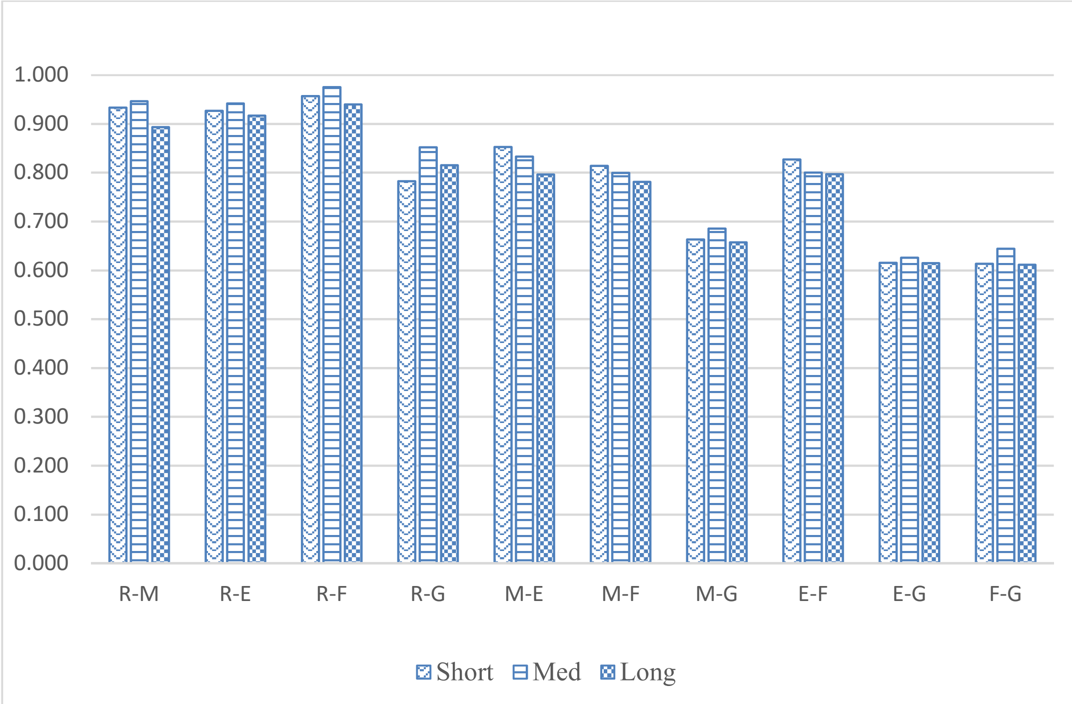

Figure 9: F-Score Measures Of The Parallel Techniques Using Different Texts Lengths