An Approach for Extracting and Disambiguating Arabic Persons' Names Using Clustered Dictionaries and Scored Patterns

Omnia Zayed, Samhaa El-Beltagy, and Osama Haggag

Center of Informatics Science, Nile University, Giza, Egypt
{omnia.zayed,samhaaelbeltagy,osama.haggag}@gmail.com

Abstract. Building a system to extract Arabic named entities is a complex task due to the ambiguity and structure of Arabic text. Previous approaches that have tackled the problem of Arabic named entity recognition relied heavily on Arabic parsers and taggers combined with a huge set of gazetteers and sometimes large training sets to solve the ambiguity problem. But while these approaches are applicable to modern standard Arabic (MSA) text, they cannot handle colloquial Arabic. With the rapid increase in online social media usage by Arabic speakers, it is important to build an Arabic named entity recognition system that deals with both colloquial Arabic and MSA text. This paper introduces an approach for extracting Arabic persons' name without utilizing any Arabic parsers or taggers. Evaluation of the presented approach shows that it achieves high precision and an acceptable level of recall on a benchmark dataset.

1 Introduction

Named entity recognition (NER) has become a crucial constituent of many natural language processing (NLP) and text mining applications. Examples of those applications include Machine Translation, Text Clustering and Summarization, Information Retrieval and Question Answering systems. An exhaustive list can be found in [5]. Arabic NER has attracted much attention during the past couple of years, with research in the area achieving results comparable to those reported for the English language.

Approaches for recognizing named entities from text have been divided into three categories which are "Rule Based NER", "Machine learning based NER" and "Hybrid NER". The "Rule Based NER" combines grammar, in the form of handcrafted rules, with gazetteers to extract named entities. "Machine learning based NER" utilizes large datasets and features extracted from text, to train a classifier in order to recognize a named entity. Hence this approach converts the named recognition task into a classification task. Machine learning algorithms could be further divided into either supervised or unsupervised. The "Hybrid NER" combines the machine learning approach with the rule based approach. A comparison between the rule based approach and the machine learning approach is given in [13]. As mentioned in [1, 13, 17], it is difficult to extend the rule based approach to new domains because of the necessity of

E. Métais et al. (Eds.): NLDB 2013, LNCS 7934, pp. 201–212, 2013.

complicated linguistic analysis to detect the named entities. Conversely, the difficulty of the machine learning approach lies in that it requires a precise selection of features from a training dataset which is tagged in a certain manner to recognize new entities from new testing dataset in the same domain.

To reach acceptable results however, employment of an Arabic parser is a must in any of the above listed approaches. While this is perfectly valid for extracting named entities from MSA, it is difficult to apply on colloquial Arabic, which is currently used extensively in micro-blogging and social media contexts. The main difficulty of applying previously devised approaches on this type of media, is the fact that existing Arabic parsers cannot deal with colloquial Arabic at any acceptable degree of accuracy. Without the utilization of such parsers, the degree of ambiguity in Arabic person name detection rises significantly for reasons that are detailed in section 2.

This paper introduces an approach for extracting Arabic persons' names, the most challenging Arabic named entity, without utilizing any Arabic parsers or taggers. The presented approach makes use of a limited set of dictionaries integrated with a statistical model based on association rules, a name clustering module, and a set of rules to detect person names. The main challenges addressed by this work could be summarized as:

- Overcoming the person name ambiguity problem without the use of parsers, taggers or morphological analyzers.
- Avoiding the shortcomings of both rule based NER and machine learning based NER approaches including employment of complex linguistic analysis, huge sets of gazetteers, huge training sets, feature extraction from annotated corpus…etc. in order to be able to extend the approach to new domains, primarily colloquial Arabic, in our future work.

Evaluation of the presented approach was carried out on a benchmark dataset and shows that the system outperforms the state of the art machine learning based system. While the recall of the system falls below the state of the art hybrid system, the precision of the system is comparable to it.

The rest of the paper is organized as follows: Section 2 discusses Arabic specific challenges faced when building NER systems; Section 3 describes the proposed approach in detail. In Section 4, system evaluation on a benchmark dataset is discussed. Section 5 highlights an overview of the literature on NER systems in Arabic language. Finally conclusion and future work is presented in Section 6.

2 Arabic Specific Challenges for Persons' Names Recognition

The Arabic language is a complex and rich language, which steps up the challenges faced by researchers when developing an Arabic natural language processing (ANLP) application [11]. Recognizing Arabic named entities is a difficult task due to a variety of reasons as explained in detail in [1, 11]. Those reasons are revisited with examples:

- One of the major challenges of Arabic language is that it has many levels of ambiguity [11]. A significant level of ambiguity is the semantic ambiguity in which one word could imply a variety of meanings. For example, the word "نبيه" could imply the phrase (his prophet), the adjective (intelligent) or the name of a person (Nabih).
- Arabic named entities could appear with conjunctions or other connection letters which complicates the task of extracting persons' names from Arabic text such as "ومحمد" (and Mohammed), "كمحمد" (as Mohammed), "لمحمد" (to Mohammed), "فمحمد" (then Mohammed) or "بمحمد" (with Mohammed).
- Most of the Arabic text suffers from lack of diacritization. Lack of diacritization causes another level of ambiguity in which a word could belong to more than one part of speech with different meanings [1, 11]. For example, the word "نهي" without diacritics could imply the female name (Noha), or the verb (prohibited).
- Arabic lacks capitalization as it has a unified orthographic case [1]. In English some named entities can be distinguished because they are capitalized. These include persons' names, locations and organizations.
- Arabic text often contains not only Arabic named entities, but translated and transliterated named entities to Arabic [11] which often lack uniform representation. For example, the name (Margaret) can be written in Arabic in different ways such as "مرجريت","مارجريت", "مرغريت" or "مارغريت".
- Many persons' names are either derived from adjectives or can be confused with other nouns sharing the same script. Examples of ambiguous Arabic male names include [Adel, Said, Hakim, and Khaled] their different adjective or noun polysemy are [Just, Happy, Wise, and Immortal]. Examples of some ambiguous female names include [Faiza, Wafia, Omneya, and Bassma] which could be interpreted as [Winner, Loyal, Wish, and Smile]. Examples of some ambiguous family/last names are [Harb, Salama, Khatab and Al-Shaer] which translate to [War, Safety, Speech/Letter and The Poet].
- Moreover, some Arabic persons' names match with verbs such as [Yahya, Yasser, and Waked] their different verb polysemy are [Greets, Imprisons, and Emphasized]. In addition, some foreign persons' names transliterated to Arabic could be interpreted as prepositions or pronouns such as [Ho, Anna, Ann, and, Lee] their different prepositions or pronouns are [He, I, That, Mine].

The combination of the above listed factors, makes the recognition of Arabic person names the most challenging of Arabic named entities to extract without any parsers. Simply building a system based on straightforward matching of persons' names using dictionaries, will often result in mistakes. The traditional solution for this is using parsers or taggers. However, extracting persons' names from colloquial Arabic text invalidates this solution as existing parsers fail to parse colloquial Arabic at an acceptable level of precision mainly due to sentence irregularity, incompleteness and the varied word order of colloquial Arabic [17]. In this paper, the ambiguity problem is addressed in two ways. First, publicly available dictionaries of persons' names are grouped into clusters. Second, a statistical model based on association rules is built to extract patterns that indicate the occurrence of persons' names. These approaches will be explained in detail in section 3.

3 The Proposed Approach

In this work, a rule based approach combined with a statistical model, is adopted to identify and extract person names from Arabic text. Our approach tries to overcome two of the major shortcomings of using rule based techniques which are the difficulty of modifying a rule based approach for new domains and the necessity of using huge sets of gazetteers. Section 5 highlights the differences between the resources needed by our approach and previous approaches.

Our approach consists of two phases, as shown in Fig. 1. In the first phase, "The building of resources phase", person names are collected and clustered, and name indicating patterns are extracted. In the second phase, "Extraction of persons' names phase", name patterns and clusters are used to extract persons' names from input text. Both of these phases are described in depth, in the following subsections.

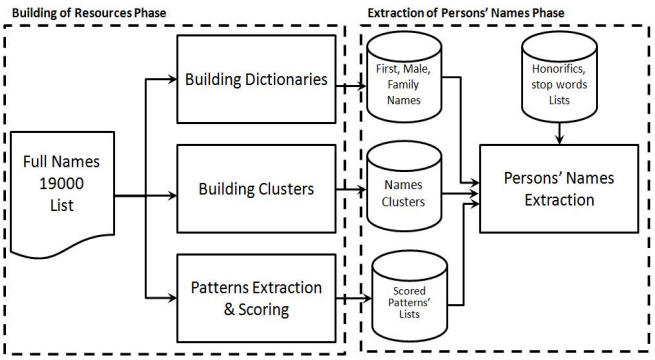

Fig. 1. System Architecture

3.1 The Building of Resources Phase

In this phase the resources on which the system depends are prepared. This phase is divided into 4 stages. In the first stage, persons' names are collected from public resources. In the second stage, dictionaries of first, male/middle and family persons' names are built from collected resources. In the third stage, names are grouped together into clusters to address the Arabic persons' names ambiguity problem as will be detailed later. In the fourth and final stage, a corpus is used to build and score patterns which indicate the occurrence of a person's name. Scoring of the patterns is done using association rules.

Name Collection. Wikipedia[1], with its huge collection of names under the people category, offers an excellent resource for building a database for persons' names. Kooora[2], which is an Arabic website for sports, also provides a large list of football

[1] http://ar.wikipedia.org/wiki/تصنيف:تراجم
[2] http://www.kooora.com/default.aspx?showplayers=true

and tennis players' names. In this stage, Wikipedia and Kooora websites were used to collect a list of about 19,000 persons' full names. Since the aim of this work is not just to recognize names of famous people, but instead to identify the name of any person even if it does not appear in the collected lists, the collection was further processed and refined in order to achieve this goal in the "Building the dictionaries" stage.

Building of Dictionaries. In this stage, the list of names collected in the previous stage (we call this list the "full_names_19000_list") was processed in such a way so as to separate first names from family names in order to create three names lists which are first, male/middle, and family names lists. Collecting a list of male names is important as a male name is often used as a family name. It is difficult to know whether a first name is a male or female name, but any middle name is always a male name. At the beginning, input names in the list are normalized using the rules presented in [12]. This step addresses the different variations of Arabic persons' name representation. As described in [17], Arabic names could have affixes such as prefixes or embedded nouns. A word preceded or followed by those affixes must not be split on white spaces, instead the word and its affix should be considered as a single entity. For example, the male name عبد العزيز (Abdulaziz) should not be split as عبد (Abd) denoting the first name and العزيز (Alaziz) denoting a family name, instead it should be treated as single entity عبد العزيز (Abdulaziz) and considered as a first name. Table 1 lists the different variations of Arabic persons' names with examples [17].

Table 1. Different variations of writing Arabic persons' names

Case	Example	Extracted Complex Entity
Simple case (no affixes)	احمد محمود Ahmad Mahmoud	Not applicable
Prefix case {عبدAbd, ابوAbou, بن Bin, ال Al, …etc}	**عبد** العزيز **ال** سعود Abdulaziz Al Saud	"عبد العزيز" First Name "ال سعود" Family Name
Double prefix case {ابوعبدAbouAbd, بن عبد Bin Abd, … etc}	سلطان **بن عبد** العزيز **ال** سعود Sultan bin Abdulaziz Al Saud	"بن عبد العزيز" Middle Name "ال سعود" Family Name
Embedded noun case {الدين El-Deen, الله Allah, … etc}	هيردي نور **الدين** Herdi Noor Al-Din	"نور الدين" Family Name
Complex name (prefix + embedded noun)	تقي **الدين** محمد **بن** معروف Taqi al-Din Muhammad ibnMa'ruf	"تقي الدين" First Name "بن معروف" Middle Name

Any honorifics or titles preceding or following a name, were removed using a compiled list of honorifics[3] that can precede or follow a name.

[3] All lists mentioned in this paper are available for download from: `http://tmrg.nileu.edu.eg/downloads.html`

Building of Name Clusters. In a simplistic world, once the name lists are built, they can be used to identify previously unseen names by stating that a full name is composed of a first name followed by zero or more male names followed by (a male name or a family name). However, as stated before, the inherent ambiguity of Arabic names, does not lend itself to such a simplistic solution. One of the problems of simple matching is the possibility of incorrectly extracting a name which is a combination of an Arabic name and a foreign name. For example, given the phrase: اتهم ايمن بوش (Ayman accused Bush), using a simple matching approach would result in the extraction of the full name ايمن بوش (Ayman Bush) even though it is highly unlikely that an Arabic person's name such as ايمن (Ayman) will appear besides an American person's name such as بوش (Bush). In the example above, the translation put the verb "accused" between "Ayman" and "Bush", but in the Arabic representation, both names are placed next to each other and preceded by the verb. Since Arabic text often contains not only Arabic names, but names from almost any country transliterated to Arabic, incorrectly identifying those could affect the system's precision significantly. A more common form of error resulting from simple matching is encountered when prepositions or pronouns match with names in the compiled name lists as explained in section 2. For example when the phrase ان محمد (That Mohammed) is encountered, the simple matching approach will result in the incorrect extraction of the full name:ان محمد (Ann Mohammed).

Given the fact the "full_names_19000_list" contains Arabic, English, French, Spanish, Hindi, and Asian persons' names, written in the Arabic language, we decided to cluster these names and allow name combinations only within generated clusters.

As a pre-processing step, the 19,000 persons' names list is traversed to build a dictionary in which the first name is a key item whose corresponding value is a list of the other middle and family names that have occurred with it. The variations of writing Arabic persons' names mentioned in the previous subsection are considered. This dictionary is converted to a graph, such that first names, middle names and family names form separate nodes. Edges are then established between each first name and its corresponding middle and family names. The resulting graph consisted of 17393 nodes, and 22518 undirected edges.

The Louvain method [9] was then applied to the graph for finding communities within the network. A community in this context is a cluster of names that are related. The Louvain method defines a resolution parameter; this parameter manages the size of communities. The standard resolution parameter $\boldsymbol{p}$ value is 1.0. A smaller value for $\boldsymbol{p}$ results in the generation of smaller communities while a larger value for $\boldsymbol{p}$ results in larger communities. By trying several values for this resolution parameter on the ANERcorp[4] [3] dataset, the value of $\boldsymbol{p}$=7 was found to produce the best results.

The outcome was a set of 1995 clusters. Each name is assigned a class number denoting which community (cluster) it belongs to.

Fig. 2 shows a snapshot of the resulting clusters. It can be observed from visualizing the data that most of the culturally similar names were grouped together; it can be noted that most of the names common in the Arabic-speaking regions were grouped

[4] http://www1.ccls.columbia.edu/~ybenajiba/downloads.html

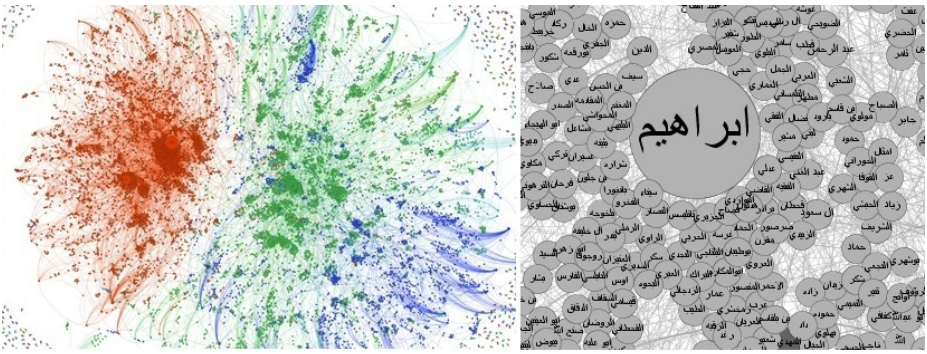

Fig. 2. Visualization of generated clusters, to the left are all generated clusters, lone clusters can be seen on the border and the two largest clusters are those of Arabic names (left) and Western names (right). To the right is a closer view of a subset of the Arabic names cluster.

together. The same applies to English and French names and to other names that are kind of unique to their region such as Asian names.

Extracting Scored Patterns. In this stage, a statistical model is built to automatically learn patterns which indicate the occurrence of a person's name.

Initially each name in the "full_names_19000_list" is used as a query to search news articles to build learning dataset from the same domain that we are targeting to extract persons' names from. Akhbarak[5] API and Google Custom Search API[6] were used to search and retrieve news stories.

Around 200 news article links were crawled (whenever possible) for each person name in the "full_names_19000_list". A total number of around 3,800,000 million links were collected using this procedure. After downloading the pages associated with these links, BoilerPipe[7] was used to extract the content or body of each news article. Very similar stories were detected and removed.

Following this step, unigram patterns around each name are extracted. Three lists are formed. A complete pattern list keeps set of complete patterns around the name with their count. A complete pattern consists of <$word_1$><name><$word_2$>. The <name> part just indicates that a name has occurred between words: $word_1$ and $word_2$. Two type of unigram pattern lists are kept: a "before" list keeps the patterns that appear before a name with their counts (example: اكد (confirmed)) and an "after" list stores patterns that occur after a name with their count (example: ان (that)).

Finally the support measure employed by association rules [2] is used to score each pattern in the three lists. Support is calculated as the ratio of the count of a pattern followed by a name over the total count of all patterns followed by a name.

The newly created three lists of scored patterns are saved descendingly according to the value of the score.

---

[5] http://www.akhbarak.net/

[6] https://developers.google.com/custom-search/v1/overview

[7] http://code.google.com/p/boilerpipe/

3.2 Extraction of Persons' Names Phase

The persons' names extraction process is dependent on the previous pre-prepared resources which are the dictionaries of first, middle, and family names, divided into clusters, a list of honorifics, a list of stop words and the patterns lists. Rules are implemented to extract persons' names from the unseen dataset of the same targeted domain. The benchmark dataset, ANERcorp [3] is used to evaluate the proposed system. The system assumes that any full name consists of a first name followed by one or more male names followed by zero or one family name. A family name appearing on its own (Bush for example), must have previously appeared as part of full name within the same text, in order to be extracted. In some text pieces, a part of a full name may appear on its own as in the phrases: واضاف كلينتون (and Clinton added), or قال محمد (Mohammed said In order to be able to disambiguate and extract such names; a list of "disambiguous names" is used. The "disambiguous names" list is a manually created list extracted from our previously created names lists and contains names that do not share the same meanings with other adjectives, nouns …etc.

When extracting names from text, employed rules can be divided into two classes: rules for "learning new names" and rules for "matching known names". In the "matching known names" rules, the generated name clusters are used to ensure that all candidate portions of a name fall in the same cluster to avoid matching mistakes and to solve the ambiguity problems mentioned previously. One of the rules used to "match known names" in the extraction phase is as follows:

```
For each word wi in the target text:
  If wi in patterns_before_list
    If wi+1 in honorific_list
    Check for names from wi+2 in the same cluster;
      Stop when a delimiter d is_found where d ∈
  (pattern_after|stop_word|punctuation|title_start)
    Else
  Check for names from wi+1 in the same cluster;
  Stop when a delimiter d is_found where d ∈
     (pattern_after|stop_word|punctuation|title_start)
  Else if wi in honorific_list
  Check for names from wi+1 in the same cluster;
    Stop when a delimiter d is_found where d ∈
(pattern_after|stop_word|punctuation|title_start)
```

The above rule is used to extract names from a sentence such as:

قال *الرئيس* محمد مرسي**ان** مصر تخطو ...

President Mohammad Morsi said that Egypt is stepping through …

This rule is generalized to extract names from sentences which contain multi honorifics before the person's name such as:

قال *رئيس الوزراء الاسرائيلي* ايهود اولمرت **إنه** عازم ...

Prime Minister of Israel Ehud Olmert said that he will …

An example of one of the rules used to "learn new names" is to check for a pattern from "the patterns before list" followed by an unknown name (not in the dictionaries) with the prefix عبد (Abd) followed by known male name and/or family name (the previous stopping criterion is used).

Another rule to learn new unknown family names is to check for a pattern from "the patterns before list" followed by a known first name followed by an unknown name such as:

وقال مدير *المؤسسه* فريدون موافق**ان** المستثمر …

The Director of the Foundation Feridun Mouafiq said that the investor …

In this example فريدون (Feridun) is a known first name while موافق (Mouafiq) is unknown family name; our system is able to extract this person's full name correctly.

Other rules are employed, but are not included due to space limitations. The next section shows how the use of patterns and the use of clusters improve the system performance.

4 System Evaluation

The presented system was evaluated using the precision, recall and f-score measures based on what it extracted as names from the benchmark ANERcorp [3] dataset. As mentioned in [3] , ANERcorp consists of 316 articles which contain 150,286 tokens and 32,114 types. Proper Names form 11% of the corpus. Table 2 provides a comparison between the results of the presented system with two state of the art systems which are the hybrid NERA approach [1] and the machine learning approach using conditional random fields (CRF) [4].

Table 2. Comparison between our system performance in terms of precision, recall and F-score with the current two state of the art systems

	Precision	Recall	F-score
Hybrid System	94.9	90.78	92.8
CRF System	80.41	67.42	73.35
Our System	**93.22**	**78.88**	**85.45**

From this comparison, it can be inferred that our system outperforms the state of the art machine learning system. However the recall of our system is still below the recall of the state of the art hybrid approach. Our system still needs some improvements to compete with the hybrid NERA approach.

Table 3. Effect of individual system's components on overall system performance

	Precision	Recall	F-score
Dictionaries Only	71.0	62.98	66.75
Dictionaries+ Clusters	77.24	58.62	66.65
Dictionaries+ Clusters+ Patterns	94.96	76.91	84.99
Dictionaries+ Clusters+ Patterns+ Disambiguation list	**93.22**	**78.88**	**85.45**

Table 3 shows the effect of using clusters, patterns and disambiguation lists on the system's performance.

5 Related Work

The majority of previous work addressing NER in Arabic language was developed for the formal MSA text which is the literary language used in newspapers and scientific books. NER from informal colloquial Arabic, currently being used widely in social media communication, has not been directly addressed. In [17], previous work on Arabic NER is discussed extensively. The currently used rule based approaches to extract named entities from MSA text, are dependent on tokenizers, taggers and parsers combined with a huge set of gazetteers. Although, those approaches might be for extracting persons' names from a formal domain, it will be hard to modify them for the colloquial domain [17].

There is some similarity between our approach and another approach based on local grammar [16] which uses reporting verbs as patterns to indicate the occurrence of persons' names. However our approach extracts patterns automatically from the domain under study, while the other approach is limited to a list of reporting verbs. NERA [15] is a system for extracting Arabic named entities using a rule-based approach in which linguistic grammar-based techniques are employed. NERA was evaluated on purpose-built corpora using ACE and Treebank news corpora that were tagged in a semi-automated way. The work presented in [10] describes a person named entity recognition system for the Arabic language. The system makes use of heuristics to identify person names and is composed of two main parts: the General Architecture for Text Engineering (GATE) environment and the Buckwalter Arabic Morphological Analyzer (BAMA). The system makes use of a huge set of dictionaries.

As mentioned in [1], the most frequently used approach for NER is the machine learning approach by which text features are used to classify the input text depending on an annotated dataset. Benajiba et al. applied different machine learning techniques [3–8] to extract named entities from Arabic text. The best performing of these makes use of optimized feature sets [4]. ANERSys [3] was initially developed based on n-grams and a maximum entropy classifier. A training and test corpora (ANERcorp) and gazetteers (ANERgazet) were developed to train, evaluate and boost the implemented technique. ANERcorp is currently considered the benchmark dataset for testing and evaluating NER systems. ANERSys 2.0 [7] basically improves the initial technique used in ANERSys by combining the maximum entropy with POS tags information. By changing the probabilistic model from Maximum Entropy to Conditional Random Fields the accuracy of ANERSys is enhanced [8].

Hybrid approaches combine machine learning techniques, statistical methods and predefined rules. The most recent hybrid NER system for Arabic uses a rule based NER component integrated with a machine learning classifier [1] to extract three types of named entities which are persons, locations and organizations. The reported results of the system are significantly better than pure rule-based systems and pure machine-learning classifiers. In addition the results are also better than the state of the

art Arabic NER system based on conditional random fields [4]. The system was extended to include more morphological and contextual features [14] and to extract eleven different types of named entities using the same hybrid approach.

Compared with other approaches, our system utilizes a far more limited set of resources. All our system requires is a large set of names, which can be easily obtained from public resources such as Wikipedia and a list of honorifics. Our system also, avoids the use of parsers or taggers and the need for annotated datasets.

6 Conclusion and Future Work

This paper presented a novel approach for extracting persons' names from Arabic text. This approach integrated name dictionaries and name clusters with a statistical model for extracting patterns that indicate the occurrence of persons' names. The used approach overcomes major limitations of the rule based approach which are the need for a huge set of gazetteers and domain dependence. More importantly, the fact that the presented work uses no parsers or taggers, and uses publicly available resources to learn patterns, means that the system can be easily adapted to work on colloquial Arabic or new domains. Our rule based approach was able to overcome the ambiguity of Arabic persons' names using clusters. Building the patterns' statistical model using association rules improved the tasks of Arabic persons' names disambiguation and extraction from any domain. System evaluation on a benchmark dataset, showed that the performance of the presented technique is comparable to the state of the art machine learning approach while it still needs some improvements to compete with the state of the art hybrid approach.

This work is a part of a continuous work to extract named entities from any type of Arabic text whether it is the informal colloquial Arabic or the formal MSA. Our plans for the future are to improve the results obtained by this approach while avoiding model over-fitting. The main intention is to test this approach on a colloquial dataset collected from Arabic social media.

References

1. Abdallah, S., Shaalan, K., Shoaib, M.: Integrating rule-based system with classification for Arabic named entity recognition. In: Gelbukh, A. (ed.) CICLing 2012, Part I. LNCS, vol. 7181, pp. 311–322. Springer, Heidelberg (2012)
2. Agrawal, R., Imielinski, T., Swami, A.: Mining Association Rules between Sets of Items in Large Databases. In: Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, SIGMOD 1993, New York, pp. 207–216 (1993)
3. Benajiba, Y., Rosso, P., BenedíRuiz, J.M.: ANERsys: An Arabic Named Entity Recognition System Based on Maximum Entropy. In: Gelbukh, A. (ed.) CICLing 2007. LNCS, vol. 4394, pp. 143–153. Springer, Heidelberg (2007)
4. Benajiba, Y., Diab, M., Rosso, P.: Arabic named entity recognition using optimized feature sets. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, EMNLP 2008, pp. 284–293. Association for Computational Linguistics, Morristown (2008)

5. Benajiba, Y., Diab, M., Rosso, P.: Arabic named entity recognition: A feature-driven study. IEEE Transactions on Audio, Speech, and Language Processing 17(5), 926–934 (2009)
6. Benajiba, Y., Diab, M., Rosso, P.: Arabic named entity recognition: An svm-based approach. In: The International Arab Conference on Information Technology, ACIT 2008 (2008)
7. Benajiba, Y., Rosso, P.: Anersys 2.0: Conquering the ner task for the Arabic language by combining the maximum entropy with pos-tag information. In: IICAI, pp. 1814–1823 (2007)
8. Benajiba, Y., Rosso, P.: Arabic named entity recognition using conditional random fields. In: Workshop on HLT & NLP within the Arabic World. Arabic Language and Local Languages Processing: Status Updates and Prospects (2008)
9. Blondel, V.D., Guillaume, J., Lambiotte, R., Lefebvre, E.: Fast unfolding of communities in large networks. Journal of Statistical Mechanics: Theory and Experiment, 10008 (2008)
10. Elsebai, A., Meziane, F., Belkredim, F.Z.: A rule based persons names Arabic extraction system. In: The 11th International Business Information Management Association Conference, IBIMA 2009, Cairo, pp. 1205–1211 (2009)
11. Farghaly, A., Shaalan, K.: Arabic natural language processing: Challenges and solutions. ACM Transactions on Asian Language Information Processing 8(4), 1–22 (2009)
12. Larkey, L., Ballesteros, L., Connell, M.E.: Light stemming for Arabic information retrieval. Arabic Computational Morphology 38, 221–243 (2007)
13. Mansouri, A., Affendey, L.S., Mamat, A.: Named entity recognition using a new fuzzy support vector machine. In: Proceedings of the 2008 International Conference on Computer Science and Information Technology, ICCSIT 2008, Singapore, pp. 24–28 (2008)
14. Oudah, M., Shaalan, K.: A pipeline Arabic named entity recognition using a hybrid approach. In: Proceedings of the 24th International Conference on Computational Linguistics, COLING 2012, India, pp. 2159–2176 (2012)
15. Shaalan, K., Raza, H.: NERA: Named entity recognition for Arabic. Journal of the American Society for Information Science and Technology, 1652–1663 (2009)
16. Traboulsi, H.: Arabic named entity extraction: A local grammar-based approach. In: Proceedings of the International Multiconference on Computer Science and Information Technology, vol. 4, pp. 139–143 (2009)
17. Zayed, O., El-Beltagy, S., Haggag, O.: A novel approach for detecting Arabic persons' names using limited resources. In: Complementary Proceedings of 14th International Conference on Intelligent Text Processing and Computational Linguistics, CICLing 2013, Greece (2013)