Arabic Named Entity Recognition—A Survey and Analysis

Amal Dandashi, Jihad Al Jaam and Sebti Foufou

Abstract As Arabic digital data has been increasing in abundance; the need for processing this information is growing. Named entity recognition (NER) is an information extraction technique that is vital to the processes of natural language processing (NLP). The ambiguous characteristics of the Arabic language make tasks related to NER and NLP very challenging. In addition to that, work related to Arabic NER is rather limited and under-studied. In this study, we survey previous works and methodologies and provide an analysis and discussion on the feature sets used, evaluation tools and advantages and disadvantages of each technique.

1 Introduction

Named Entity Recognition (NER) was initially introduced as an information extraction technique. NER is a task that locates, extracts and automatically classifies named entities into predefined classes in unstructured texts (Nadeau and Sekine [7]. It covers proper names, temporal expressions and numerical expressions. Proper names are classified into three main groups: persons, locations and organizations. A class can be divided into sub-classes to form an entire hierarchy, i.e., location can be classified into city, state and country. The majority of NER studies have been focused on the English language, as it is the internationally dominant language, while research on other languages for the NER task has been limited.

Arabic is a richly morphological language of complex syntax. The lack of simplicity in the characteristics and specifications of the Arabic language make it a challenging task for NER techniques. Arabic can be classified into three types:

A. Dandashi (✉) · J. Al Jaam · S. Foufou
Department of Computer Science, Qatar University, Doha, Qatar
e-mail: amal.dandashi@qu.edu.qa; amaldandashi@gmail.com

J. Al Jaam
e-mail: jaam@qu.edu.qa

S. Foufou
e-mail: a.karkar@qu.edu.qa

G. De Pietro et al. (eds.), *Intelligent Interactive Multimedia Systems and Services 2016*, Smart Innovation, Systems and Technologies 55,
DOI 10.1007/978-3-319-39345-2_8

Classical Arabic, Modern Standard Arabic and Colloquial Arabic. It is imperative for the task of NER to be able to distinguish between those three types. Classical Arabic is the formal version of Arabic used for over 1,500 years in religious scripts. Modern Standard Arabic is that used in today's newspapers, magazines, books, etc. Colloquial Arabic is the spoken Arabic used by Arabs in their informal day to day speech and differs in dialect for each country and city. There are several specifications of the Arabic language that do not make NER an easy task; lack of capitalization, agglutination, optional short vowels, ambiguity inherent in named entities and lack of uniformity in writing styles. Add to that common spelling mistakes and shortage of technological resources such as tagged corporas and gazetteers, we have several issues to tackle for tasks associated to natural language processing (NLP).

In this study, we aim to analyze and report on the performance of recent NER studies dedicated to the Arabic language. In Sect. 2, we include analysis on the features and tag sets used, algorithm and methodology, performance evaluation and pros and cons of each technique. In Sect. 3 we include a descriptive listing of Arabic NER tools currently available. Finally, Sect. 4 concludes the study.

2 Analysis on Recent Studies

2.1 Study 1

The impact of using different sets of features in three different machine learning frameworks is investigated by Benajiba et al. [3], for the Arabic NER task. The machine learning frameworks tested are support vector machines (SVM), maximum entropy (ME), and conditional random fields (CRM). Nine different data sets of genres and annotations are explored along with lexical, contextual and morphological features. In order to evaluate robustness to noise of each approach, different feature impacts are measured in isolation and incremental combination.

Methodology: The three comparable approaches which were used for named entity recognition (NER) are:

1. SVM: known to be robust to noise and have strong generalization capabilities for large feature sets. The *Yamcha* toolkit is used to apply SVM to the NER task.
2. ME: known to provide model with the least biases possible. A customized ME approach is implemented to carry out the experiments, and the *Yasmet* tool is used for weight estimation.
3. CRF: oriented towards segmenting and labeling sequence data, and can represent probability distributions. Conditional probabilities of classes are maximized during the training phase. *CRF++* is used for experimentation.

The optimal feature sets investigated are: Contextual (CXT), lexical, gazetteers, morphological features, part-of-speech tags, nationality and corresponding English capitalization.

Performance Evaluation:

1. The NLE-corpus is used. It comprises of text collected from several newswire web resources, and the text is manually annotated. The tag set used is CoNLL set which includes four classes: Person, Location, Organization and Miscellaneous.
2. The ACE 2003, 2004 and 2005 corpora are used. The data in ACE is annotated for the following tasks: entity detection and tracking, relation detection and recognition, event detection and recognition. ACE 2003 consists of two data genres: Broadcast News (BN) and Newswire (NW). The ACE 2004 also includes the Arabic Treebank (ATB). The ACE 2005 includes the Weblogs (WL) but not the ATB genre.

Experimental setup:

1. Metrics: CoNLL evaluation metrics are used for precision, recall and F1-measure (harmonic mean between precision and recall).
2. Experiments: Three sets of experiments take place; a baseline, a parameter setting of experiments and feature-engineering experiments. The latter involves exploring individual features, ranking features according to impact and evaluating the SVM, ME and CRF approaches. In order to find the optimal number of features, evaluation of SVM, ME and CRF approaches involves combining the top N-elements of the ranked features each increment, starting from $N = 1$ up to $N = 22$.

The performance evaluations are done using fivefold cross validation on each corpus independently. For the NLE-corpus, the same ratio of test data size to training data size that has been used in CoNLL competitions has been also used for this study. For the ACE data, the authors used the same ratio of test data size to training data size which has been used for the ACE evaluation.

Almost all the corpora tested achieved high performance and improvement over the baseline. For ACE 2003, an F1 score of 83.34 for the BN genre. The WL genre yielded the worst results, which may be due to the inclusion of dialectical language and randomness of the WL data compared to the other genres. The NW genre of the ACE 2005 data yielded an F-measure of 77.06.

The features ranked according to impact are as follows, first place is the POS tagger, second is the CAP feature, and that is confirmation that the lack of capitalization in languages such as Arabic complicates the NER task considerably. Several of the MFs are ranked in a scattered manner ranging from third to twenty-second. LEX features ranks range from fifth till nineteenth place, and show that marking the first and last three characters of a word can be useful for a NER approach. Incremental features selection resulted in better performance, since using

a selected feature set leads to avoidance of providing the classifier with noisy information.

The ME approach is the most sensitive to noise and obtained significantly lower results than that of the CRF and SVM, specifically when the number of features exceeded six. When the top seven features were used, the SVM approach depicted better performance than that of the CRF approach. Otherwise, the performance of SVM and CRF obtained similar results. And while the latter two approaches give different false alarms, they tend to miss the same named entities and thus cannot be used to complement each other.

2.2 Study 2

Zitouni et al. [11] present a statistical approach to Arabic mention detection and chaining (MDC) systems. The approach is based on the Maximum Entropy (ME) principle. The system first detects mentions in an input document and then chains the identified mentions into entities. The ME framework allows for a large range of feature types, including lexical, morphological, syntactic and semantic features. The authors consider the additional challenges of the Arabic language, as one needs to correctly identify and correct enclitic pronouns by adding segmentation as a processing step. In Arabic text, nominals and pronouns are attached to words, hence special attention needs to be given when processing Arabic text in order to be able to detect partial parts of a word as a mention.

Methodology: The MDC system in this study proceeds in two main phases: detecting mentions and then partitioning the detected mentions into entities. The latter phases are based on the ME technique. This approach is language independent and must be modified to accommodate the Arabic Language specifications. The methodology is detailed as follows:

1. Arabic word segmentation: The weighted finite state transducer (WFST) is used for this section, which was initially a manually trained corpus and later refined using unsupervised learning on a large corpus of 155 million words. The segmentation process is as follows; partition Arabic text into a sequence of segments (tokens), and separation of delimited words into prefixes, stems and suffixes.
2. Mention detection: identify and characterize the main actors in an Arabic text; people, locations, organizations, geopolitical entities. It is formulated as a classification problem, where the labels are assigned to tokens in the text, indicating whether it starts a specific mention, is inside a given mention or is outside any mention. The ME classifier is selected for this process, as it integrates arbitrary types of information and makes classification decisions by aggregating all information for a specific classification. ME associates a set of weights with features, and computes the probability distribution. Weights are estimated during the training phase of the data set to maximize the likelihood. In this study ME

model is trained using the sequential conditional generalized iterative scaling technique (SCGIS), and used a Gaussian prior for regularization.
3. Features: the features used in this mention detection system are: Lexical, Stem n-gram gazetteer-based, syntactic and features from other named-entity classifiers.
4. Mention chaining: uses a machine learning based approach:

   - Entities in a document are created from mentions incrementally and in a synchronous fashion. This is done by constructing a data structure called a Bell tree.
   - First mention is used to create the root of the Bell tree.
   - Subsequent mentions may start a new entity or link with an existing entity.
   - End result is a Bell tree structure with each leaf node representing a coreference outcome.
   - Coreference resolution problem is converted into scoring the competing paths in Bell tree.

5. Automatic Content Extraction (ACE) entity detection and recognition: detecting certain specified types of entities. All mentions pertaining to each entity are to be detected and four pieces of information are to be defined as follows:

   - Mention type can classify as: person, organization, location, geopolitical entity, facility, vehicle or weapon.
   - Mention subtype or subcategory of a mention type
   - Mention class, whether generic or specific
   - Mention type, whether named, nominal or pronominal.

Performance Evaluation: The Arabic ACE 2007 is used for experimentation. 323 documents (80000 words) are used for training and 56 documents (18000) for testing. The result is 17,634 mentions (7816 named, 8831 nominal and 987 pronominal) for training, and 3566 for testing (1673 named, 1682 nominal, and 211 pronominal).

Performance metrics for mention detection use F-measure to report evaluation results. Measuring performance for mention chaining involves the use of two performance metrics; the first is a Constrained-Entity Alignment F-measure (CEAF), which measures the percentage of mentions that are in the right entities. The second metric is the ACE value, the metric used for the ACE task. The ACE value is computed by summing values of aligned system entities, subtracting values of false alarm entities, and normalizing over the values of reference entities. A perfect coreference system would get a 100 % ACE-value.

Discussion: The Arabic main-type mention detection system obtained an F-measure of 80.0 in the ACE evaluation. The authors started the system evaluation by only allowing access to the lexical features and gradually increasing features with each increment. A system using only lexical, stem and gazetteer features achieves a measure of 76.5. Syntactic and other classifier feature outputs add more than 3 F-measure points to overall performance (80 vs. 76.5).

Results of the system evaluation depict a reasonable performance of the mention types: geopolitical entities, person and vehicles, with an F-measure of 89.2, 81.5 and 75.0, respectively. The remaining mention types received a low performance with the F-measure ranging from 68.5–55.1.

Lexical attributes and distance features were found to be the most essential for coreferencing. Lexical attributes features lead to 76.3 F-measure and 68.8 ACE-value, and addition of the distance feature improves F-measure by 4.1 and ACE-Value by 6.1 points. While Stem Match and syntactic features do not help as much as the latter features, they do improve performance in small increments by acting as extra optimizing features, which is a typical component of a statistical system that has reached a good performance level. While syntactic feature improves ACE-value by 0.4 %, it slightly delays the CEAF by 0.3 %, which is an indication that different configurations may lead to better results.

2.3 Study 3

Zitouni and Benajiba [10] proposed a semi-supervised approach that utilizes multilingual parallel data (English-Arabic) in an effort to enhance the mention detection (MD) task. The challenge with MD is directly related to the complexity of the morphology of a given language. For this reason, the authors explore the idea of using a MD system designed to suit the needs of a resource rich language (RRL), namely English, to improve the performance of a MD system for a target language (TL), namely Arabic. To maximize the scope of the study to be potentially applicable for several languages, the authors have chosen to experiment with a limited set of annotation types for the TL, as well as a larger set. The proposed system uses the maximum entropy (ME) principle, which combines arbitrary types of information to make classification decisions.

The hypothesis of this study is that an MD system with a rich feature set such as English may be used to boost performance for TL such as Arabic, provided that the donor language system's resources are capable of surpassing its TL counterpart's. To test this theory, MD tags are projected from RRL to TL via a parallel corpus, and several linguistic features about the automatically tagged words are extracted. Then several experiments have been conducted, involving adding these new features to the TL baseline MD system. The benefit of the ME technique lies with its ability to integrate multiple features seamlessly. However, in some cases it may lead to an overestimation of confidence, particularly in low-frequency features. This problem arises when a hard constraint is reinforced on a feature whose estimate is not reliable. The adjustment used in this experiment involves using the sequential conditional generalized iterative scaling (SCGI) technique.

Methodology: In order to validate the hypothesis of this study, English and Arabic corpuses are used, part of the ACE 2007 evaluation. In Arabic, words are composed of zero or more prefixes, a stem, and a zero or more suffixes, which are all considered tokens. Any contiguous sequence of tokens may present a mention.

The first step is segmentation of text, and then classification is performed on tokens. MD systems across these English and Arabic languages use a large range of features, which may be classified as: lexical, syntactic and information obtained from other named-entity classifiers with customized semantic tags. As additional features, the assigned classifier tags are utilized: lexical, syntactic, semantic and stem-n-gram features.

Cross language mention projection: In order to use a RRL like English, to enhance MD in a TL, like Arabic, a parallel corpus with word alignment must be utilized, and a MD system in the RRL must be available. The steps are as follows:

1. Extracting necessary features
2. Running MD on the RRL text of the parallel corpus, resulting with tagged text
3. Utilizing word alignment file to project mentions from RRL to TL
4. Alignment file has many-to-many structure, which describe which words from the RRL side correspond to the words on the TL side.
5. Words with no alignment will not be tagged
6. Result is a text in the TL annotated with mentions obtained by propagation from RRL.
7. Once Arabic corpus is tagged, features are extracted: Gazetteers, model-based features, lexical content, surrounding phrase head-words, and parser-based features.

Performance Evaluation: The main goal is to investigate the effect of using a semi-supervised multilingual approach to enhance the Arabic MD system. First the use of an unlabeled data corpus and a MD system based on a RRL (English) is explored to show how Arabic MD system benefits. Then the monolingual technique is explored in order to compare results. Experiments are conducted on specified partitions of the ACE 2007 data set.

Corpus Alignment techniques used: (1) hand aligned data, (2) automatically aligned data, (3) Combined hand aligned and automatically aligned data, and (4) monolingual data.

The following features are used for the cross-lingual (English-Arabic system): baseline, En-nLexCon, En-nPhHead, En-nSynEnv, En-Gaz, En-Model and Comb.

Discussion: The impact of features obtained through cross-lingual system proved to be far more effective in enhancing system performance.

Hand-aligned data results: This model achieved a 57.7 F-measure. The low performance is an indication of the level of noise, and we may understand that better performance can be achieved when there is no human annotated data. However, when the Arabic MD system is poor in resources, significant improvements are obtained. When only Lexical features are used, there is an improvement of approximately1.9 points. When the Arabic MD system uses a rich feature-set Syntac, a 1.5 point improvement is obtained. The model based feature (En-Model) depicted the greatest performance (76.22) when the Arabic MD system uses a feature-set that includes more than just the Lexical features. Comb features

achieved higher performance than Baseline features (77.18 and 75. 68, respectively).

Automatically-aligned data results: the 22 million word corpus is split into four subsets, each with 5, 10, 17 and 22 million words, respectively. From each of these subsets, the impact of the features extracted from each subset is analyzed separately. The most optimal results were found from the 17 million word subset. Best results have been obtained with the use of the En-nSynEnv feature for the 17 million word subset, with an F-measure of 76.02.

Combined hand-aligned and automatically-aligned data: automatically aligned data helped capture more of the unseen mentions, whereas the hand-aligned data helped decrease the number of false alarms. When the Comb feature is used with the Stem baseline model, the F-measure obtained (76.85) is 1.2 points higher than the baseline model that uses Lexical, Stem and Syntactic features (75.68). Using a propagation approach like this one, definitely helps bootstrap the process and achieves better performance.

Monolingual data: the same 17 million words subset features are used to demonstrate performance for monolingual data experimentation. The only difference is that the Arabic data is tagged with the baseline Arabic MD system. The En-nSyncEnv system using only Lexical features with project from the English MD system has very similar performance with the Baseline system used with the monolingual experiment (75.62, 75.68, respectively). When more resources are used with Arabic, such as Syntac feature, together with the monolingual features, Ar-nSynEnv, results are very comparable to those achieved when the system uses Lexical features with cross-lingual En-nSyncEnv features (75.61, 75.62, respectively).

The proposed approach of cross-lingual propagation MD systems is efficient and practical due to the fact that various resources are already available in RRL such as English. There are many parallel corpora that cover many language pairs that can be used. The proposed approach uses parallel aligned corpora and MD system designed to enrich a TL system by using information in a RRL system. Experimentation led to the conclusion that this approach is more effective than dealing with languages with limited resources with the use of unsupervised data. Results have shown that performance decreases when too many resources are used, but even when all resources are used, significant gains are observed. When the TL has access to limited resources, a 2.2 point improvement is made, whereas when all resources are used for the TL, only a 0.5 point improvement is observed. With absence of human-annotated Arabic data, performance F-measure reaches to 57.6 using only mention propagation from RRL system. In conclusion, the approach using cross-lingual information outperforms the approach using monolingual information.

2.4 Study 4

Chen and Ng [4] propose a hybrid approach to coreference resolution aimed for the CoNLL-2012 shared task is proposed. The hybrid approach consists of combining specific features of rule-based and learning-based techniques. The system models coreference in OntoNotes for three challenging languages that come from very different language families; English, Chinese and Arabic. The system is designed to resolve references in all three languages. There are four main tracks of the proposed system; closed track for all three languages, and an open track for the Chinese language. The system also exploits genre specific information, and optimizes parameters with respect to each genre. The system performs mention detection as an initial step, which is followed by coreference resolution. The parameters of those two components are jointly optimized with respect to the desired evaluation measure.

Methodology:

1. Mention detection component: two step approach is employed;
    a. Extraction step: identifying named entities (NE) and employing language-specific heuristics to extract mentions. In order to increase upper bound on recall, mention extraction is done by utilizing syntactic parse trees.
    b. Pruning step: improve precision by employment of language-specific heuristic pruning and language-independent learning-based pruning.
2. Coreference Resolution: Employs heuristics and machine learning, via the Stanford multi-pass sieve approach. As most of these sieves are unlexicalized, the multi-pass sieve (heuristic rules) approach is optimized by incorporating lexical information via machine learning techniques. Each sieve heuristic rule extracts a conditional-based coreference relation between two mentions. Sieves are ordered by precision, and in order to resolve a set of mentions in a document, the resolver makes multiple passes over them, finally using the rules only in the final sieve to find an antecedent for each mention.
    a. Sieves for English: Composed of 12 sieves, modeled after those employed by the Stanford Resolver, with some optimizations.
    b. Sieves for Chinese: For Chinese, the authors participate in close and open track. The sieves used for both tracks are the same. The Chinese resolver is composed of 9 sieves: Chinese Head Match, Precise Construct, Pronouns, Discourse Processing, Exact String Match, Strict Head Match (A,B,C), and Proper Head Match.
    c. Sieves for Arabic: Only one sieve is employed for Arabic; the Exact Match sieve. Other sieves such as Head Match and Pronouns sieve were also implemented but excluded from the study as they did not yield good results.

Performance Evaluation: For each language genre, the following steps are implemented: (1) learn lexical probabilities from training set, (2) obtain most optimal parameter values, using two parameter estimation algorithms, and (3) chose

the parameter with that obtained the better performance on the development set to be the final set of parameter estimates for the resolver.

Discussion: Mention detection results and coreference results are both depicted in terms of recall, precision and F-measure. Ablation experiments are performed, showing for each language track combination the results obtained without editing: the rule relaxation parameters, the probability thresholds, and all parameters tested. However, there were no rule relaxation parameters for Arabic.

The average F-measure for the English closed track is 60.3, while the average for the Chinese closed track is 68.6, and the closed Arabic track 47.3. The Arabic track obtained the lowest results due to the fact that the Arabic language is a highly inflectional language and the authors claim to have little linguistic knowledge of the language to design effective sieves. As for the open track, which was implemented solely for the Chinese language, the average F-measure obtained is 79.0.

2.5 Study 5

Arabic knowledge bases are valuable lexical semantic resources with high influence on several Natural Language Processing tasks. There are two main types of knowledge bases: collaborative knowledge bases (CKB) and linguistic knowledge bases (LKB). The main difference between the two is that CKBs are developed by voluntary nonprofessionals on the web that follow nonbinding guidelines, while LKBs are developed by linguistic professionals and are guided by theoretical linguistic models. CKBs are ever-growing and available to everyone for free use (such as Wikipedia or Wiktionary), while LKBs are rigid and unavailable for free use, apart from WordNet. Aljazeera in itself, is considered to be among the richest Arabic knowledge bases on the web. While it is not considered a CKB nor an LKB, it shares characteristics from both; it is developed and edited by linguistic professionals, it is ever-growing and freely available online, and its content is semantically interlinked.

All three Arabic knowledge bases (CKBs, LKBs and Aljazeera.net) have several applications in Natural Language Processing. However, the following challenges must be noted: (3) Arabic CKBs are less structured than LKBs and Aljazeera.net, and contain more noisy information, (2) Arabic CKBs rely on social control for accuracy and precision, whereas LKBs and Aljazeera.net rely on professional editorial provision, and (3) While CKBs and LKBs have encyclopedic and linguistic orientation, Aljazeera.net is formed on events-based orientation. Al-Kouz et al. [1] form the following hypothesis for this study: Arabic CKBs, LKBs and Aljazeera.net could form a complementary resource, and that an Arabic Semantic Graph (ASG) could be constructed based on these complementary knowledge bases. The authors hence present a framework design for the development of a semantic graph extractor for Aljazeera.net, a high performance Java-based API that has capabilities to extract implicit and explicit information.

Methodology: The Aljazeera.net content is rich in quantity and quality. An efficient framework design is needed in order to use CKBs and Aljazeera as a complementary knowledge resource for large scale NLP tasks. The authors have developed a general purpose framework designed to extract and build the ASG from these knowledge bases. The Arabic Semantic Graph Extraction Framework (ASGEF) comes with high performance Java API(JAKL) packages that are to be available freely for the research communities. The APIs are designed to work on an object oriented programming paradigm with the following objects in consideration; ALJAZEERA, NEWSARTICLE, SEMANTICENTITY, NAMEDENTITY, SEMANTICENRICHMENT, and CATEGORY.

1. The ASGEF is to be built based on Aljazeera.net platform as the main knowledge resource, and CLBs used as complementary semantic enrichment sources.
2. Java based Data Machine and Time Machine APIs are developed in order to utilize Aljazeera.net platform.
3. Time Machine can be used to crawl Aljazeera.net from a specific point in time in offline mode, only after first use of Data Machine.
4. Time Machine can also be used to crawl new content on Aljazeera.net in online mode.
5. Parsing stage starts: two parsers are used, the File Parser and the Web Parser.
6. File Parser is capable of parsing hierarchical directory structure, and translating it into Arabic hierarchical related entities.
7. Ontology Builder API then transforms hierarchical related entities into Arabic Ontology.
8. Data Set Builder API transforms hierarchical related entities into a manually annotated data set.
9. The Web Parser parses HTML pages within the hierarchical directory structure to extract text.
10. The Content Extractor API employs the File and Web Parser results to extract a semi-structured data set.
11. The result is extracted semi-structured data with explicit semantic data represented as interlinked articles and hyperlinked named entities. This data is applicable to named entity detection tasks.
12. Java APIs are utilized to access local dumps of Arabic CKB resources.
13. The SGB provides the JAKL, which operates on semantically related and enriched entities that created from Aljazeera.net, Wikipedia, and Wiktionary.
14. The output of the SGB is the Arabic Semantic Graph, which can be published to Linked Open Data (LOD).

Discussion: The system proposed in this study could be used in several NLP tasks and large-scale research projects that involve analysis, computation of semantic relations between entities, and access of Arabic knowledge base semantic graph structure, among other information retrieval applications. The ASGEF architecture has the following functionalities: (1) enable large-scale Arabic NLP tasks with

computational efficiency, (2) enable reproducible experimental results based on Aljazeera.net along with CKBs such as Wikipedia and Wiktionary, (3) enable reliable mathematical representation of knowledge, (4) free and easy to use.

3 Arabic NER Tools

3.1 Fassieh

Attia et al. [2] introduce an Arabic annotation tool called Fassieh. It enables the production of large annotated Arabic text corpora, classified according to the following Arabic linguistic models: Part of speech tagging, morphological analysis, phonetic transcription and discretization, and lexical semantic analysis. The inherent ambiguity of these analysis models is statistically resolved with Fassieh. The system also presents various other auxiliary features which enable a normalized, guided and efficient proof-reading for the factorized corpus. These features include morpheme-based dictionaries, short-contect statistical ranking, illustrative GUI tools such as character and word status coloring. Fassieh is not only an annotation tool, but also incorporates evaluation, demonstrative and tutorial functions for Arabic Natural Language Processing.

3.2 MATAR

Zaraket and Jaber [9] introduce and open source tagging tool with a visual interface. This tool enables the annotation of Arabic text corpora with the automated utilization of morphological tags. MATAR allows for a Boolean-based specification of tags, considering predicates and relations between morphological parts of text and values of a given morphological feature. Users can enter manual tags, edit existing ones, compare tag sets and compute accuracy results, all through a user-friendly interface.

3.3 MADAMIRA

MADARMIRA is a system designed by Pasha et al. [8] that can be utilized for morphological analysis and disambiguation of Arabic text. It combines aspects of two previously used systems for NLP; MADA [6] and AMIRA (Diab et al. [5]. MADAMIRA optimizes both previously mentioned systems with a more streamlined, robust, portable, extensible and faster Java-based implementation. It includes several tasks useful for NLP processes: part-of-speech tagging, tokenized forms of words, diacritization, lemma stemming, base phrases, and NER.

4 Conclusion

NER is among the most vital processes for the development of NLP systems. Accurate NER mechanisms ensure the success of a range of NLP systems like machine translation and information retrieval. Arabic textual information resources are increasing all over the internet, which makes the task of automated NER a necessary one, for the sake of classifying online data such as web pages, articles, informative texts, emails, blogs, etc. This study provides a survey and analysis of progress done towards Arabic NER. As the presence of named entities in one language leads to a correspondence in other languages, studies of NER in a specific language allots valuable insight and research for developing multi-lingual NLP systems. We hope this analytical study provides fruitful guidance for researchers dealing with Arabic NER systems.

Acknowledgements This publication was made possible by GSRA grant # 1-1-1202-13026 from the Qatar National Research Fund (a member of Qatar Foundation). The findings achieved herein are solely the responsibility of the author(s).

References

1. Al-Kouz, A., Awajan, A., Jeet, M., Al-Zaqqa, A.: Extracting Arabic semantic graph from Aljazeera. net. In: 2013 IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies (AEECT), (pp. 1–6). IEEE, Dec 2013
2. Attia, M., Rashwan, M.A., Al-Badrashiny, M.A.S.A.A.: Fassieh, a semi-automatic visual interactive tool for morphological, PoS-Tags, phonetic, and semantic annotation of Arabic Text Corpora. IEEE Trans. Audio Speech Lang. Process. **17**(5), 916–925 (2009)
3. Benajiba, Y., Diab, M., Rosso, P.: Arabic named entity recognition: A feature-driven study. IEEE Trans. Audio Speech Lang. Process. **17**(5), 926–934 (2009)
4. Chen, C., Ng, V.: Combining the best of two worlds: a hybrid approach to multilingual coreference resolution. In: Joint Conference on EMNLP and CoNLL-Shared Task, pp. 56–63. Association for Computational Linguistics, July 2012
5. Diab, M.: Second generation AMIRA tools for Arabic processing: fast and robust tokenization, POS tagging, and base phrase chunking. In: 2nd International Conference on Arabic Language Resources and Tools (2009)
6. Habash, N., Roth, R., Rambow, O., Eskander, R., Tomeh, N.: Morphological analysis and disambiguation for dialectal Arabic. In: HLT-NAACL, pp. 426–432 (2013)
7. Nadeau, D., Sekine, S.: A survey of named entity recognition and classification. Lingvisticae Investigationes **30**(1), 3–26 (2007)
8. 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. In: Proceedings of the Language Resources and Evaluation Conference (LREC). Reykjavik, Iceland
9. Zaraket, F.A., Jaber, A.: MATAr: Morphology-based Tagger for Arabic. In Computer Systems and Applications (AICCSA), 2013 ACS International Conference on, pp. 1–4. IEEE, May 2013

10. Zitouni, I., Benajiba, Y.: Aligned-parallel-corpora based semi-supervised learning for Arabic mention detection. IEEE/ACM Trans. Audio Speech Lang. Process. **22**(2), 314–324 (2014)
11. Zitouni, I., Luo, X., Florian, R.: A cascaded approach to mention detection and chaining in arabic. IEEE Trans. Audio Speech Lang. Process. **17**(5), 935–944 (2009)