IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 4, No 3, July 2010
ISSN (Online): 1694-0784
ISSN (Print): 1694-0814


# Integrated Machine Learning Techniques for Arabic Named Entity Recognition

**Samir AbdelRahman, Mohamed Elarnaoty, Marwa Magdy and Aly Fahmy**

**Computer Science Department**
**Faculty of Computers and Information, Cairo University**
**5 Dr. Ahmed Zewail Street, Postal Code: 12613, Orman, Giza, Egypt**

**Abstract**

Named Entity Recognition (NER) task has become essential to improve the performance of many NLP tasks. Its aim is to endeavor a solution to boost accurately the identification of extracted named entities. This paper presents a novel solution for Arabic Named Entity Recognition (ANER) problem. The solution is an integration approach between two machine learning techniques, namely bootstrapping semi-supervised pattern recognition and Conditional Random Fields (CRF) classifier as a supervised technique. The paper solution contributions are the exploit of pattern and word semantic fields as CRF features, the adventure of utilizing bootstrapping semi-supervised pattern recognition technique in Arabic Language, and the integration success to improve the performance of its components. Moreover, as per to our knowledge, this proposed integration has not been utilized for NER task of other natural languages. Using 6-fold cross-validation experimental tests, the solution is proved that it outperforms previous CRF sole work and LingPipe tool.

***Keywords:*** *Bootstrapping Pattern Recognition, Conditional Random Fields, Arabic Named Recognition, Cross-Validation.*

## 1. Introduction

Named Entity Recognition (NER) is an information extraction subtask to classify proper names from unstructured texts into categories of names. NER task has been used to evolve many Natural Language Processing (NLP) subtasks, such as Information Retrieval and Question Answering [1, 5].

In the last few decades, Arabic Named Entity Recognition (ANER) task has been garnered much efforts to boost its performance. The ANER challenging task is to gather huge corpora or immense white lists/gazetteers that deal with possibly most of Arabic language challenges such as orthography, ambiguity, and complexity. Indeed, most ANER researches tend to collect such data carefully to include all possible language cases having these peculiarities; though, this data may not exist; the task may not be accurate and it is time-consuming. Consequently, some researchers prefer to use small data set coupled with some tools to help them encounter these problems. We decided to follow this approach and we used the Research and Development International (RDI)[1] toolkit to assist us to deal with such difficulties.

Many ANER researches have been prompted to use rule-based technique [1, 9, 18, 20] or machine translation techniques [12, 13, 19]. However, their authors state that the proposed systems work effectively if abundant large size corpora for ANER analysis and training phases exist. In the last decade, many machine learning techniques [4, 5, 6] have been exploited using only a set of some language features coupled with small training data sets to build accurate ANER models. Four main points may be concluded from these models. First, the model may lose the identification of some named entities due to the selected features and the small size of such corpora. Second, Conditional Random Fields (CRF) is proved to be one of the most effective learners for ANER task. Third, many Arabic language features have been probed for supervised learning techniques; nevertheless, Arabic semantic fields feature has not been yet considered. Fourth, up to our knowledge, bootstrapping semi-supervised pattern recognition [7] has not been tested as an ANER technique.

We present an integration between machine learning techniques to tackle ANER problem for identifying 10 Named Entity (NE) classes namely Person, Location, Organization, Job, Device, Car, Cell Phone, Currency, Date, and Time classes. The proposed integration components are a bootstrapping semi-supervised pattern recognizer and CRF as a supervised learner. This

[1] http://www.rdi-eg.com/.

integration is novel since to date, up to our knowledge, it has not been utilized in any natural language NER task. The contributions of the proposed solution for ANER task are as follows:
1- We show evidence that bootstrapping semi-supervised pattern recognition technique is a promising ANER technique.
2- We prove that semantic fields feature is one of the effective CRF features.
3- In our approach, both integration components boost iteratively each other. While, the pattern recognizer extracts all expected patterns to let CRF identify more named entities, CRF is trained to employ some local optimal features, including pattern index and word semantic fields, with the aim of generating potential seeds to help in removing the recognizer noisy patterns.

To evaluate the first system contribution, we test the resultant patterns manually and we measure their effectiveness to boost CRF in the proposed integration. Moreover, the other two contributions are verified against previous CRF solo work [5] and LingPipe[2] tool.

The paper is organized as follows: Section 2 illustrates primary Arabic Language challenges; Section 3 explains the thoughts of some related systems; Section 4 presents our approach main three components; Section 5 pinpoints our approach specifications and parameters; Section 6 illustrates how to integrate our solution components; Section 7 submits the experimental results and analysis. Finally, Section 8 draws our conclusions and future work.

## 2. Arabic Language Challenges

Arabic language is a high complex language which embeds five critical challenges for NLP tasks. First, Arabic is not a case-sensitive language; it has no capital letters. Latin languages consider this feature very significant in NER tasks since their NEs usually start with capital letters.

Second, Arabic is a high inflectional language; often a single word has more than one affix such that it may be expressed as a combination of prefix(s), lemma, and suffix(s). The prefixes are articles, prepositions, or conjunctions. The suffixes are generally objects or personal/possessive anaphora. For example, the Arabic word "وبمصريتنا" is interpreted in English as "and that we are Egyptians".

Third, Arabic has some variants in spelling and typographic forms. Other languages, such as English, normalize all such variants. For example, "غرام-جرام"/ "Gram" [3] is a spelling deviation and "أستراليا-استراليا"/ "Australia" is a typographic variant.

Fourth, Arabic texts have different sorts of ambiguities (different meanings). For example, "رجب"/"Ragab" in Arabic may be used as a person name, month, or a fear verb.

Fifth, Arabic resources, such as corpora, gazetteers, and NLP tools, are either rare or not free. This defect makes collecting and analyzing such resources time-consuming particularly if the NER technique depends on such resources [12, 13, 19, 20].

## 3. Related work

As a typical supervised learning system, [5] used their owned tagged corpus named ANERcorp[4] and gazetteers, ANERGazet, to identify Person, Location, Organization, and Miscellaneous classes with F-measure (Equation 4) of 73.34%, 89.74%, 65.76%, and 61.47% respectively. They prove that CRF precede their ME (Maximum Entropy) work [6] by 12 points in the F-measure average of all classes. We compare our work with this CRF work and we use its tagged corpus and gazetteers as a subset of the research data set (Sections 4, 5, and 7). In this research, we propose a local optimal feature set (Section 5) for each NE class; moreover, we intend someday to investigate [4] and/or other intelligent techniques to find out the optimum feature set for each NE class.

Indeed, the supervised learning technique is designed to use a set of features and small size of training data to identify possible categories. In spite of the fact that supervised learning technique precision (Equation 2) is relatively high, its recall (Equation 3) is degraded because its training examples and features may not cover some NE occurrences. In this research, we aim to combine Arabic semantic fields and patterns with other CRF features to boost ANER task. This integration is designed to increase the number of correct NEs and hence to improve the CRF recall.

To date, no other natural language NER system follows our proposed integration. Indeed, [8, 14, 15, 21] use bootstrapping supervised learning techniques to improve the corpus training classifier with the most reliable examples extracted from unlabeled data set. Interestingly,

[2] Alias-i vendor publishes the LingPipe ANER performance on all 6-fold experiments of the corpus used in this research: http://alias-i.com/lingpipe/demos/tutorial/ne/read-me.html.

[3] Throughout the paper, we use "A"/ "E" as a notation to indicate "E" is an English interpretation of "A".

[4] http://users.dsic.upv.es/~ybenajiba/.

[22] and TextRunner[5] present a sort of our proposed integration to serve NE relation extraction task in which the systems are fed with some predefined patterns of relations such that the pattern recognizer and CRF are trained to extract all similar relations and their entities. Even though all mentioned works are sorts of bootstrapping supervised learning, they are not designed for identifying NE classes and their contexts.

## 4. The Proposed Solution Components

We use **Conditional random fields (CRF)[6] classifier**. It is a discriminative probabilistic model [17] which is used for segmenting and labeling the sequential data. It is a generalization of Hidden Markov Model in which its undirected graph consists of nodes to represent the label sequence $y$ corresponding to the sequence $x$. The aim of CRF model is to find $y$ that maximizes $p(y \mid x)$ (Equation 1) for that sequence. Section 5.3 describes our proposed CRF feature set.

$$p(y \mid x) = \frac{1}{z(x)} * \exp\Big(\sum_{t}\sum_{k} \lambda_k f_k(y_{t-1}, y_t, x)\Big)$$
$$z(x) = \sum_{y \in Y} \exp\Big(\sum_{t}\sum_{k} \lambda_k f_k(y_{t-1}, y_t, x)\Big) \quad (1)$$
; $\lambda_k$ is the weight of $f_k$

**Dual Iterative Pattern Relation Expansion (DIPRE)** [7] is the first pattern extraction algorithm. Its aim is to bootstrap gigantic set of web pages in order to find iteratively all occurrences of the relation instances in the corpus coupled with their pattern representatives. The longest matching algorithm is used to find the most prefix and suffix occurrences. This paper proposes a bootstrapping pattern recognizer which adapts this algorithm (Section 5.1) and its evaluation criterion (Section 6).

**The Research and Development International (RDI) toolkit** mainly consists of Arabic RDI-ArabMorpho-POS tagger [2] and RDI-ArabSemanticDB tool [3]. **RDI-ArabMorpho-POS tagger** includes Arabic Morphology and Part of Speech (POS) models. The POS tagging depends on the word morphology features instead of nowadays Arabic POS taggers which use light stemming to tag the words. It is proved that the morphology model coverage is 99.8% and the tagger accuracy is 90.4% average of five experiments. **RDI-ArabSemanticDB tool** is composed of Arabic Lexical Semantics Language Resource (database) and its related interface. The tool aims to store and handle mammoth number of Arabic word roots with their (semantic) lexical features. The database archives approximately 40,000 Arabic words, 1840 semantic fields, and 20 semantic relations, such as synonyms, antonym, hyponymy, and causality. This research uses these RDI tools (Sections 5.2 and 5.3) to help it tackle the mentioned Arabic challenges.

## 5. The Proposed Solution Parameters

Our approach parameters, namely NE classes, data set, CRF feature set, and evaluation methods are demonstrated as follows.

### 5.1 Named Entity Classes and Patterns

The proposed solution aims to recognize the following ten NE classes from the News Corpus:
**1.** Person: names of people;
**2.** Location: names of map places;
**3.** Organization: names of companies, associations, firms or organizational entities;
**4.** Job: names of positions that could be employed or occupied;
**5.** Device: names of home machines, such as television, radio, and washing machine;
**6.** Car: names of vehicles;
**7.** Cell phone: names of portable phones;
**8.** Currency: names of all money forms;
**9.** Date: names of particular calendar points or time periods in which events may be occurred;
**10.** Time: names of non-spatial continuum in which events occur permanently.

The proposed solution designs the following pattern scheme for all classes:

$$< NE\ pattern > \leftarrow < prefix >< NE >< suffix >$$
$$< NE > \leftarrow < NE\ occurrence >^{1+}$$
$$< prefix > \leftarrow < prefix\ occurrence >^{0+}$$
$$< suffix > \leftarrow < suffix\ occurrence >^{0+}$$

This syntax reveals that each pattern may express the related NE as the context defined by <NE pattern> and NE itself formulated by <NE>; where NE is the first 3 letters of the related class name; we use this notation throughout this paper. The context is composed of NE prefix and suffix parts such that each part may consist of zero or more Arabic words; also, any NE form may be composed of one or more NE Arabic words. For each mentioned class, its <NE> is tagged with BIO scheme as B-class, I-class, and O-class; where B is the beginning of the class words, I to present inside class words, and O to tag the other words.

[5] http://www.cs.washington.edu/research/textrunner/
[6] http://crfpp.sourceforge.net/.

### 5.2 Data Collection and Preprocessing

ANERcorp was used to train and test CRF classifier to identify all mentioned classes except Device, Car, and Cell Phone. Arabic search engine[7] was used to crawl the web getting 5MB and 150MB News documents for these classes and the pattern recognizer data respectively.

ANERGazet gazetteers (Section 3) were used for presenting some NE occurrences of the first three mentioned classes; we ourselves crawled the web to compile gazetteers for Job, Device, Car, and Cell phone classes. Moreover, a Ministry gazetteer was compiled having the most common names of Arabic Ministries. We found that this gazetteer would be helpful to recognize Job class since several of its News occurrences would be preceded by names of ministries.

To sustain the solution pattern recognizer, sets of seeds for the first 7 mentioned classes were selected from their gazetteers due to their majority uses in Arabic. We didn't use gazetteers for Currency, Date, and Time classes because we found that their keywords would be naturally very limited. So, we selected only few of their high used occurrences as sets of seeds.

RDI-ArabMorpho-POS tagger was used to help us in normalizing, extracting features, and tagging all research corpora tackling all the mentioned Arabic challenges. In all corpora, all occurrences of numbers, months, weeks, nationalities, and ministries were tagged by <Num>, <Mon>, <Wee>, <Nat>, and <Min> respectively.

### 5.3 CRF Feature Set

We categorize our proposed solution features into three types namely: 1) unigram word feature (UF) for the word $w_i$ feature, 2) window gram feature (WF) for n-gram word feature before and after $w_i$; it is decided to select[8] $n = 3$, and 3) bi-feature (BF) for the combination of two features. The CRF feature presentation (Equation 1) would be like $f_i(x, y_{t-1}, y_t) = 1$ for $x$="بوش"/"Bush", and $y_t$=B-Pers, and 0 otherwise. The solution 15 features according to this categorization are described as follows:

**1. Word [WF]** is the word itself.

**2. POS [WF]** is the Part of Speech (POS) tagging of the word**.** RDI-ArabMorpho-POS tagger was used to extract 6 tags, namely Noun, Verb, Transliteration, Number, Symbol, and Undefined.

**3. BPC [WF]** presents the Base Phrase Chunks (atomic parts) [16] of a sentence. Yamcha[9] training toolkit was used to extract this feature. The toolkit usually tags each token by B-Tag, I-Tag, or O-tag such that each tag may be NP, VP, PP, CONJP, ADJP, or ADVP.

**4. Gaz [WF]** is a binary feature to present the existence of the word in the gazetteers.

**5. Gram Character [UF]** presents the first/last two and three characters of the word. This feature has important effect in ANER. For example, "عبد"/"Abd" is very repetitive prefix in Arabic person names.

**6. Semantic fields [WF]** feature presents the RDI-ArabSemanticDB word semantic fields identification. Its main role is to gather all occurrences related to each other in single semantic fields identification. For example, "مجموعة"/"Group","مؤسسة"/"Association","هيئة"/"Organization", and "شركة"/ "Company" occurrences are related to Company domain so they may be good indicators for Organization class recognition.

**7. Pattern [WF]** is either the index of the pattern in which the word is included or zero if no pattern is matched (Section 7).

**8. Morphological features [UF]** are the binary set of the word lexical features extracted by RDI-ArabMorpho-POS tagger. They are attributes of each word namely:

**8.1 Suffix and Prefix** is the feature to indicate if the word has suffix or prefix sub-words, e.g. "لزميله"/"**to his** friend". Since by definition most of ANEs have no suffix and prefix, the feature could be a good sign for NE existence.

**8.2 Diptote("الممنوع من الصرف"):** many proper names and names of places in Arabic are diptote, e.g. "أحمد"/ "Ahmed" and "باكستان"/ "Pakistan".

**8.3 Definiteness "ال"/"the":** this feature is very crucial in recognition of many NE classes. For example, several Arabic names of organizations start with this article as "الأمم المتحدة"/ "The United Nations".

**8.4 Interjection Article**, such as "يا"/"oh", is very important in Person class recognition since the name, which occurs after this type of articles, mostly is a person name.

**8.5 Relative Pronoun:** this feature has special Arabic words, such as "الذي، التي، الذين", have some effective uses as English words like "who, whom, which". Indeed, most nouns that precede them are NEs.

**8.6 Nasikh Particle**, such as "... ، كان ، ان", always needs subject after it and usually the subject is a NE.

**8.7 Interrogation Article**, such as "ء"/ "Hamza", has an English interpretation as a questioning article. However, it is often succeeded by a NE occurrence in Arabic.

**8.8 Relative Adjective [BF]** is the feature always combined with one of the above features in ANER task.

[7] http://www.alzoa.com/.

[8] Our research values of thresholds were designed to anticipate all possible lengths of NE occurrences.

[9] http://chasen.org/~taku/software/yamcha/

Examples of this feature are "المجموعة **الأستثمارية**"/"**The Investment** Group" and "رونالدو **البرازيلي**"/"The **Brazilian** Ronaldo". These examples show the use of Adjective Article coupled with Definiteness feature.

It is worth pointing out that [5] uses only the first 4 above features. Also, [6] uses only the first 5 mentioned features in addition to 5 morphological features including only feature 8.3. We decided to dedicate a CRF classifier for each NE class. For each classifier, the Greedy Regression Feature Selection Algorithm [11] is run to get the classifier local optimal set of features. The greedy algorithm results per NE class, commonly for all class fold experiments, shown as follows:

- Person, Car and Job: all features;
- Organization: all features except 8.6;
- Location: all features except 8.5 and 8.6;
- Device: all features except 8.1, 8.4, 8.5, 8.6, and 8.7;
- Cell phone: all features except 8.2, 8.3, 8.4, 8.6, and 8.7;
- Currency: features 1, 3, 5, and 7 ;
- Date: features 1, 3, 5, 6, 7, 8.3, and 8.8;
- Time: features 1, 3, 5, 6, 7, 8.2, 8.3, and 8.8;

### 5.4 Evaluation Criteria

To gauge the solution CRF performance, precision (2), recall (3) and F-measure (4) are used.

$$precision = \frac{True\ \ Positive\ \ NEs}{retrieved\ \ NEs} \quad (2)$$

$$recall = \frac{True\ \ Positive\ \ NEs}{relevant\ \ NEs} \quad (3)$$

$$F\text{ - }measure = 2 * \frac{(\ precision\ \ * recall\ \ )}{(\ precision\ \ + recall\ \ )} \quad (4)$$

The evaluation of generated patterns is carried out as a Matcher Module task (Section 6). Also, $F_{Pr,Re}$ (Equation 5) [10] and unbiased $F_{TP,FP}$ (Equation 6) [10] are used to gauge our 6-fold cross-validation experiments.

$$\begin{aligned} \Pr &= \frac{1}{k} * \sum_{i=1}^{k} precision^{(i)} \\ \mathrm{Re} &= \frac{1}{k} * \sum_{i=1}^{k} recall^{(i)} \\ F_{\Pr,\mathrm{Re}} &= 2 * \frac{(\Pr\ * \mathrm{Re})}{(\Pr + \mathrm{Re})} \end{aligned} \quad (5)$$

$$\begin{aligned} TP &= \sum_{i=1}^{k} TP^{(i)};\ TP \text{ is True Positive} \\ FP &= \sum_{i=1}^{k} FP^{(i)};\ FP \text{ is False Positive} \\ FN &= \sum_{i=1}^{k} FN^{(i)};\ FN \text{ is False Negative} \\ F_{TP,\ FP} &= \frac{2 * TP}{(2 * TP + (FP + FN))} \end{aligned} \quad (6)$$

We use Equation 5 to compare the proposed integration with the published performance results of LingPipe for each ANERcorp fold. Moreover, we use the unbiased measurement (Equation 6) to measure the effectiveness of the pattern feature on CRF performance.

## 6. The Solution Matcher Module

The proposed solution runs orderly a sequence of the three modules namely CRF classifier, the pattern recognizer, and the matcher module till no new NE occurrence is extracted. For each NE class (Section 5.1), these modules are run and fed up with the collected data sets (Section 5.2) and a subset of the solution feature set (Section 5.3). While CRF classifier yields some NE occurrences as the best seeds for the pattern recognizer, the recognizer uses the matcher module to produce good patterns for boosting CRF classifier. The matcher module steps are summarized as follows:

**1.** All recognizer module patterns are sorted by their number of occurrences. The pattern having number of occurrences less than threshold $\tau_1$ =5 is removed. Subsequently, each pattern is manually tagged by <prefix>, <NE>, and <suffix> to formulate it as the prefix phrase, NE occurrences, and the suffix phrase respectively. Finally, the sorted patterns are indexed such that the pattern positive and negative index values are dedicated for its <NE> and <prefix>/<suffix> respectively.

**2.** All preprocessed CRF corpora are tokenized. Each token coupled with its prefix and suffix words are inspected to exactly match the above indexed patterns. This step is to discover possible sequence of tokens that match each pattern. The step output is to label each token with first matched pattern index. Hence, the token label is zero, if the token is not a substring of any indexed pattern, or negative/positive value otherwise.

**3.** Using the tags of CRF corpora of the underlying class, each pattern precision is calculated by dividing its true positives by its matched occurrences. These results are the set of patterns and their precisions.

**4.** The set of patterns are ordered by their precisions such that the patterns having precision less than threshold $\tau_2$ =0.3 are removed. The module doesn't filter the patterns whose NE occurrences aren't found in the

tagged CRF corpora and which are manually found that they are good pattern candidates. The step result is the set of candidate patterns.

**5.** Step 2 is repeated to retag the CRF corpora. If CRF doesn't yield new NE occurrences, the algorithm is stopped otherwise the pattern reorganizer is recalled.

## 7. Experimental Analysis

The experimental analysis aims to verify the proposed integration contributions. The extracted patterns are manually evaluated to prove their reliabilities. 6-fold cross-validation experiments were run on all CRF training corpora. We compared the results of the main 3 NE classes with LingPipe; LingPipe is a supervised learning state-of-art NER tool that uses Dictionary-based tagger with Hidden Markov Model chunker for identification procedures. ANERcorp has 4901 sentences with 150286 tokens. While we divided the corpus to 6 folds of 25000 tokens, LingPipe segmented the corpus to 6 folds of 817 sentences.

### 7.1 Pattern Aspects and Examples

Table 1. Pattern Experiment Data Attributes

| Class | Gaz. | Seeds | Patterns (Start) | Patterns (Final) |
|---|---|---|---|---|
| Per | 2328 | 35 | 92285 | 175 |
| Loc | 2183 | 32 | 51183 | 85 |
| Org | 403 | 28 | 27100 | 74 |
| Job | 70 | 15 | 47491 | 88 |
| Dev | 253 | 20 | 2501 | 25 |
| Car | 223 | 20 | 514 | 22 |
| Cel | 184 | 22 | 3110 | 18 |
| Cur | - | 10 | 782 | 12 |
| Dat | - | 20 | 23524 | 61 |
| Tim | - | 30 | 10112 | 101 |

For each NE class, Table 1 shows some features of pattern experiments, namely gazetteer size (Gaz.), the number of seeds (Seeds), the total number of the first iteration extracted patterns (Patterns Start), and the total number of last iteration (Patterns Final). In this Table, we present only for each class the worst fold results, i.e. the fold that gets initially the largest number of patterns, given that we fix the gazetteers and the seeds in all experiments.

Some Arabic NE keywords are frequently occurred in numerous other contexts. This produces huge frequencies of extracted patterns in the first iteration. In the last iteration, the number of generated patterns is dramatically reduced compared with the first iteration results due to the use of CRF and its training data to yield all possible good patterns removing noisy ones.

In the all pattern experiments, some patterns are revealed commonly among all classes. For example, the pattern which includes the phrase "التالي ذكرهم"/"such as" is frequently used among all classes. The typical occurrence of this pattern is " الرؤساء التالي ذكرهم الملك عبدالله و الرئيس أوباما "/"the presidents such as king Abdullah and president Obama". Some highly ranked NE class patterns, indicators, or phrases, and the corresponding occurrences are listed as follows:

**1. Person:**

- <D-verb> $^{0|1}$ <Job> $^{0+}$ <Per><Job> $^{0+}$ <Nat>
  <D-verb> ← "أعلن-صرح-قال-أكد"/ "affirm"
- <Job> <Nat><Per>
- <Per><Job>(<Loc>|<Org>)
- "السيد-السيده-الأستاذ-الأستاذه"/"Mr.-Mrs." as good person name indicators habitually come before names.
- "بن-الأول-الثاني-..."/"the son of -the first – the second,.." may be often occurred as parts of person names.
- "حفظه الله-سدد خطاه- رعاه الله- المغفور له"/"Bless him" may be frequently occurred before or after person names.
- "تم الأتفاق مع(بين)- بالتفاوض مع(بين)- بحضور كل من"/ "agreement with(between) – negotiation with (between) – in the attendance of" may occur before person names.

**2. Location:**

- <Num> <Loc-keys><Noun> $^{0+}$
  <Loc-Keys> ← "ميدان-شارع"/ "street-square"
- <Jobs-Keys><countries>
  <Job-Keys ← "رئيس-ملك-أمير"/"President-King-Prince"
- "العاصمة/The Capital" <Nat><Loc>
- <Loc-Keys><Loc><Nat> $^{0|1}$
  <Loc-Keys> ← "ولايه- إقليم- محافظه"/ "State-Province"
- Directions such as North, East, West, and South come sometimes before or after the location name as good location name indicators.
- "يعيش في – الموزع المعتمد في"/ "resident in - official distributer in" may come before location names.
- "جزيره- جبل- ميدان- بحر"/"Island-Mountain-Square-Sea" may occur as parts of location names.

**3. Organization:**

- "أفتتاح-تأسيس-إنشاء"/ "establishment" <Org>
- Many job names appear before organization names. Moreover, many key phrases, such as " مسئول العلاقات العامه ل – المسئول الإعلامي ل – المتحدث الرسمي ل"/"The General Public Representative for", are repeatedly exposed before organization names.
- <Org>"أحد الشركات التابعه (المملوكه ل)"/"One of the companies followed to(owned by)" <Org>
- As parts of organization names, the words, such as "صحيفه – منظمه – حزب – شركه – وكاله – إتحاد – معرض",

come as translations of "journal- association– party- company- agency- union- exhibition".

**4. Job:**

- <Job><Ministry>
- <Job>$^{1+}$
- Many special verbs precede the job names such as declaration verbs or suggestion verbs to state or propose clue. These verbs are commonly found before job names which precede person names as illustrated above.
- Many appraisal words, such as " زعامة – رئاسة- فخامة – سعادة – معالي", are commonly indicators which precede job names.
- As some famous job names, "دكتور–مهندس-وزير- نائب-لاعب-قائد-ملك-أمير"/ "Vice-doctor-engineer-minister-player-leader-king-prince" are frequently used.

**5. Other classes:**

- In their patterns, some common prefixes are found in Car, Device, and Cell Phone such as "من نوع -من طراز"to indicate type of the productions; also, prefixes like "الموزع الجديد – وكيل ل"/"the distributor or agent of" and "أنتاج- أطلاق-إعلان عن طرح"/"production-commencing - putting into markets" are occasionally used.
- In its pattern, currency names are often preceded by <Num>. Some common prefixes are found, such as " تبلغ- قيمه – بسعر – أسعار تراوحت بين- تكلفت نحو", to present the ranges of currency values or costs. Other phrases, such as "تراجع الدولار أمام الين" or "تراجع الين في أسواق المال", may be generalized to present the declination of currencies in front of each other or in some location names.
- The proposed integration could extract many potential Time/Date expressions. Indeed, Date pattern [20] is extracted easily having occurrences such as " السبت 19 من كانون الثاني/يناير1999" to indicate "Saturday, 19th Kanoun the second/January 1999". Other Date expressions are founded, like "الحالي-نهاية الشهر الهجري-بداية" or " خلال الأسبوع الربع الثالث من العام الحالي-القادم", to indicate date start or end at the current day, week, month, or year quarter. Time patterns include some Arabic idioms of the specific day time or time duration such as " حوالي نصف ساعه بعد الثامنه مساء" which may be translated as "Approximately half an hour after 8 evening".

## 7.2 The Integration Solution Results

Fig.1 shows some interesting conclusions. In Person and Organization classes, the proposed CRF model supersedes LingPipe Models. In Location class, we retagged ANERcorp to recognize Location NEs to sustain our extracted <Loc> and <Loc pattern>. For example, we tagged "ميدان بيت القاضي"/ "Kady House Square" as a <Loc> and "Kady House" as another <Loc>. However, in the original ANERcrop tagging, "Kady House" only is tagged as a Location NE. So, we definitely couldn't compare exactly our CRF Location results with any previous ANERcorp-based work.

After boosted by patterns, the $F_{Pr,Re}$ solution results of the mentioned 3 classes are 67.89%, 88.60%, and 65.17% respectively. These results prove that patterns boost highly our solution CRF work and LingPipe model as well.
We don't know the fold used by [5]; they state [4] only that they used 5/6 of ANERcrop tokens for CRF training and the remaining tokens for testing purpose. Indeed, our nearest fold outcomes to their work are the 5th fold ( Fig. 2) outputs. The proposed solution CRF results of this fold are 72.16%, 79.20%, and 67.18% for Person, Location, and Organization classes respectively. After boosted with patterns, the solution outcomes are 74.06%, 89.09%, and 75.01% respectively. These results precede the mentioned CRF work results (Section 3).

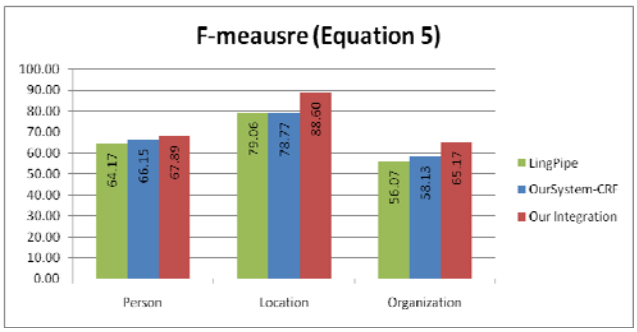


Fig. 1 Comparison between LingPipe and Our Approach

In the whole 6-fold experiments using Equation 5, Fig.1 concludes that the proposed integration precedes LingPipe and the Proposed CRF features without patterns. In Fig. 2 sub-figures via Equation 4, the proposed integration outcomes, for each class fold experiment, prove that they are better than the proposed CRF experiments without pattern feature. It means that pattern enhances CRF work even some folds may include some unfamiliar proper names (Fig 2.3 and 2.10). The thorough conclusion of all experiments shows that CRF may be affected with some unpredictable word sequences such that all mosque names which are mentioned extensively (Fig 2.3) in Organization folds 2 and 3. Moreover, in Time class (Fig 2.10), fold 3 includes repeatedly some irregular phrases such as " أكثر من ساعه"/"more than one hour" which weakens the classification results.

For each class row, Table 2 shows the most repetitive number of iterations executed for all folds (#Iter), the results of the first experiment iteration listed in the lower sub-row, and the last iteration indicated by the upper sub-

row. The impact of this Table is to analyze the roles of system parts to boost each other using Equation 6.

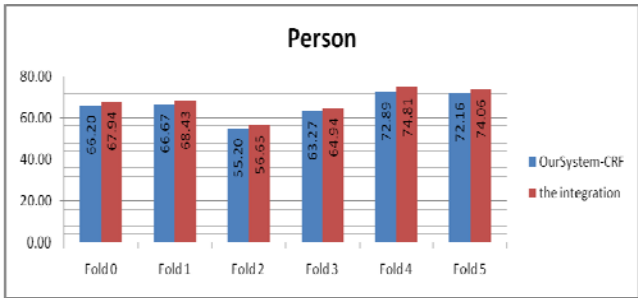


Fig. 2.1 Person Class Comparison: Our CRF with/without Patterns

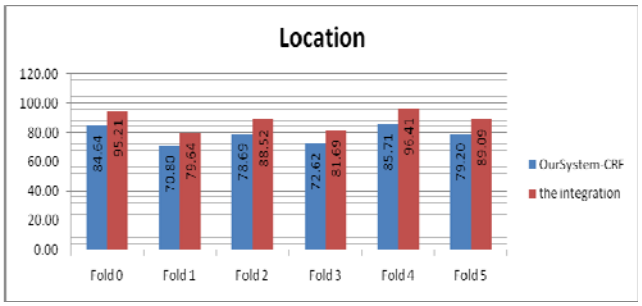


Fig. 2.2 Location Class Comparison: Our CRF with/without Patterns

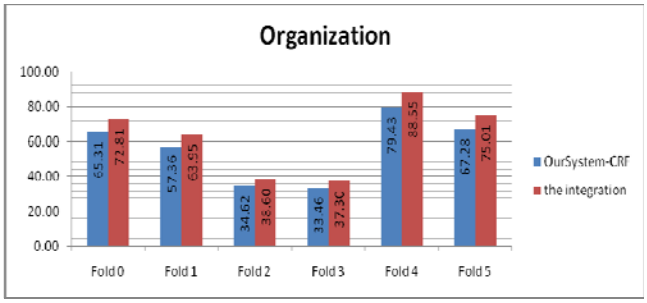


Fig. 2.3 Organization Class Comparison: Our CRF with/without Patterns

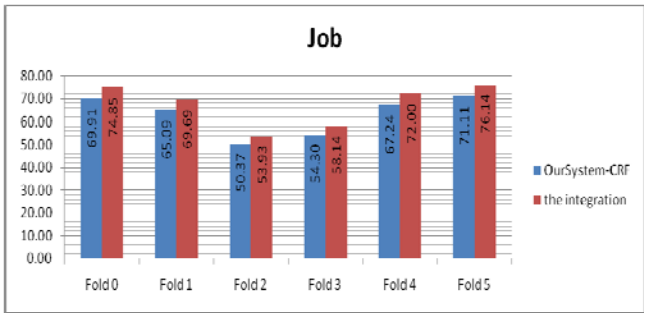


Fig. 2.4 Job Class Comparison: Our CRF with/without Pattern

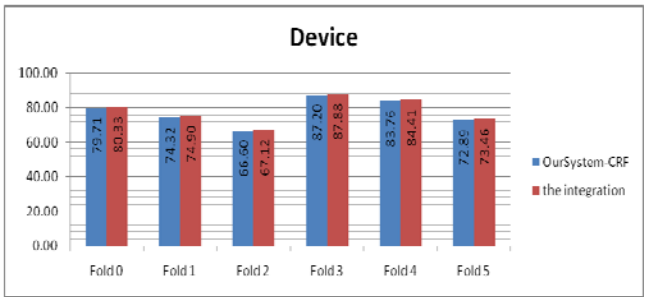


Fig. 2.5 Device Class Comparison: Our CRF with/without Patterns

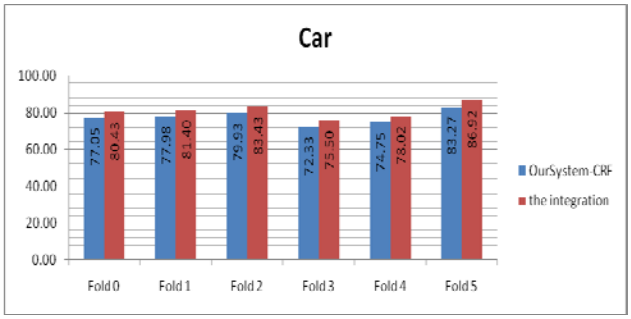


Fig. 2.6 Car Class Comparison: Our CRF with/without Patterns

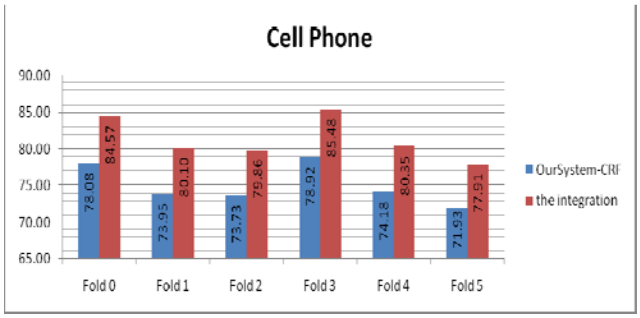


Fig. 2.7 Cell Phones Class Comparison: Our CRF with/without Patterns

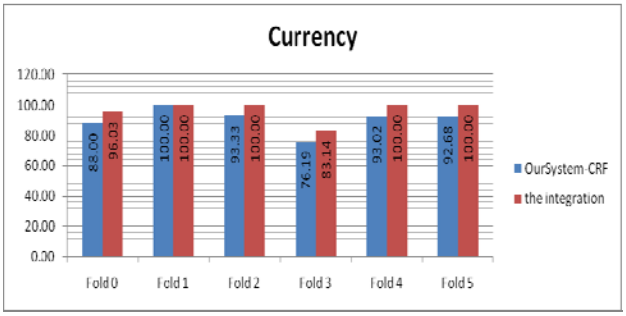


Fig. 2.8 Currency Class Comparison: Our CRF with/without Patterns

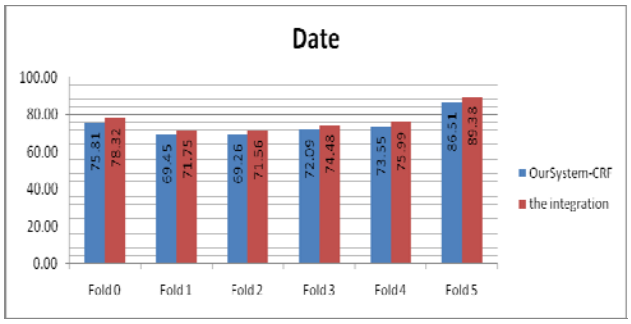


Fig. 2.9 Date Class Comparison: Our CRF with/without Patterns

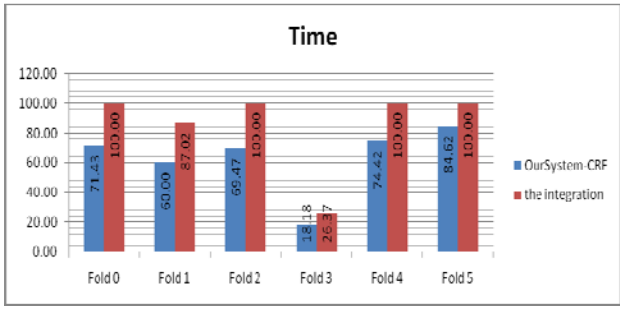


Fig. 2.10 Time Class Comparison: Our CRF with/without Patterns

The extracted patterns have the major impact to increase CRF classifier performance in all experiments. Fortunately, both CRF and the pattern recognizer successfully bootstrap each other to recognize Person, Organization, Job, Car, and Cell Phone. The achieved $F_{TP.FP}$ increments are 1%, 7%, 5%, 3%, and 6% having 3, 3, 2, 3, and 2 iterations respectively. In some classes, such as Person, Device, and Date, the effects of patterns on CRF are not major. In other classes except Time class, such impacts are moderately large. Auspiciously, Time classifier is boosted with 27% $F_{TP,FP}$ increment. The reasonable analysis of these sorts of impacts demonstrates that these CRF boosting varieties are totally anchored in how the related features solely without pattern could help the classifier in its recognition task. Moreover, the analysis also verifies the importance of patterns to help CRF proficiently.

To finalize the experiments, we test the role of semantic fields feature in ANER task. We find that missing this feature negatively influences all NE classifiers, except Currency classifier; it is irrelevant in Currency NE classification. When this feature is turned off, we have the first iteration F-measure decrement averages of all folds in Person, Location, Organization, and Job classes are 2%, 1%, 3%, and 5% respectively. Also, the averages of 2% decrements are realized for the remaining classes. It is found that the amount of each class keywords or occurrences, related semantically to each other, is proportional to such increments. Therefore, it is valuable to claim that this feature has a very vital role in the ANE CRF classification process.

Table 2. CRF With/Without (W/O) Patterns

| Class (# Iter) | | Precision | Recall | $F_{TP.FP}$ |
|---|---|---|---|---|
| Per (3) | W | 89.20% | 54.68% | 67.80% |
| | O | 87.01% | 53.23% | 66.05% |
| Loc (1) | W | 96.05% | 80.86% | 87.80% |
| | O | 89.37% | 69.25% | 78.03% |
| Org (3) | W | 84.95% | 60.02% | 70.34% |
| | O | 88.45% | 49.00% | 63.07% |
| Job (2) | W | 84.53% | 58.97% | 69.47% |
| | O | 87.13% | 51.51% | 64.74% |
| Dev (1) | W | 81.76% | 73.70% | 77.52% |
| | O | 83.66% | 71.58% | 77.15% |
| Car (3) | W | 83.55% | 78.51% | 80.95% |
| | O | 86.00% | 71.08% | 77.83% |
| Cel (2) | W | 86.03% | 75.87% | 80.63% |
| | O | 88.25% | 65.18% | 74.98% |
| Cur (1) | W | 100.00% | 97.08% | 98.52% |
| | O | 100.00% | 82.76% | 90.57% |
| Dat (1) | W | 81.30% | 73.10% | 76.99% |
| | O | 83.12% | 67.60% | 74.56% |
| Tim (1) | W | 100.00% | 92.40% | 96.05% |
| | O | 97.18% | 53.91% | 69.35% |

## 8. Conclusions and Future Work

We opened the gate for ANER researchers to hit the weakly supervised pattern generation as one of ANER adept techniques. As shown, the technique may generate good patterns and may work cooperatively with CRF. That was met using only little size of gazetteers/data sets. Also, the technique finds easily new NE occurrences/contexts without the need for extensive re-analysis work. Consequently, it would be better someday to investigate the technique performance if it works exclusively or cooperatively with the other ANER techniques.

CRF is boosted radically. So, its pattern and semantic fields features are proved to be effective. We currently investigate Arabic semantic relation types to cope with our approach as CRF features. Also, some intelligent feature selection algorithms are being explored to get optimum feature set for each NE class.

**Acknowledgments**

This work was supported by the Center of Excellence for Data Mining and Computer Modeling-Cairo University

through Information Technology Industry Development Agency funds. All authors thank profoundly Engineer Amr Magdy Gomaa, a NLP researcher, for his endless contributions through the research experimental phase.

**Samir AbdelRahman** is an Assistant Professor at Computer Science Department, Faculty of Computers and Information, Cairo University. His M.Sc. and Ph.D. have been received from Cairo University in Computer Science Specialty. He is interested in Machine Learning, Named Entity Extraction, Text Mining, and NLP parsers. His research papers have been published mainly in Computational Linguistics/ Informatica journals and Coling conferences. His Arabic Opinion Mining work is supported and funded by the Center of Excellence for Data Mining and Computer Modeling-Cairo University.

**Mohamed Elarnaoty** is a Teaching Assistant at Computer Science Department, Faculty of Computers and Information, Cairo University. He is interested in Mathematics, Machine Learning, Named Entity Extraction, Opinion Mining, and NLP parsers.

**Marwa Magdy** is a Teaching Assistant at Computer Science Department, Faculty of Computers and Information, Cairo University. Her research interests are in Arabic NLP: Intelligent Language Tutoring System, Spelling Checkers, Named Entity Extraction, and Opinion Mining.

**Aly Fahmy** is a Professor at Computer Science Department, Faculty of Computers and Information, Cairo University. He is the Principle Investigator of the Center of Excellence for Data Mining and Computer Modeling-Cairo University. His research concerns mainly in Computational Linguistics, Text Mining, Data Mining, and Software Engineering.

## References

[1] S.Abuleil, "Extracting Names From Arabic Text For Question-Answering Systems", 7th International Conference of Computer-Assisted Information Retrieval Applications, University of Avignon, RIAO 2004, France.

[2] M.Attia and M.Rashwan, "A Large Scale Arabic POS Tagger Based on a Compact Arabic POS Tag Set and Application on the Statistical Inference of Syntactic Diacritics of Arabic Text Words", NEMLAR, 2004.

[3] M.Attia, M.Rashwan, A.Ragheb, M.A.H Al-Basoumy, and S.Abdou. "A Compact Arabic Lexical Semantics Language Resource Based on the Theory of Semantic Fields", Proceedings of the 6th international conference on Advances in Natural Language Processing, 2008.

[4] Y.Benajiba , M.Diab, and P.Rosso, "Arabic Named Entity Recognition using Optimized Feature Sets", EMNLP 2008: 284-293.

[5] Y.Benajiba, and P.Rosso, "Arabic Named Entity Recognition using Conditional Random Field, 6th Int. Conf. on Language Resources and Evaluation", LREC 2008.

[6] Y.Benajiba and P.Rosso, "ANERsys 2.0: Conquering the NER Task for the Arabic Language by Combining the Maximum Entropy with POS-tag Information", Workshop on Natural Language-Independent Engineering, 3rd Indian Int. Conf. on Artificial Intelligence, Pune, India, 2007.

[7] S.Brin, "Extracting Patterns and Relations form the World Wide Web", WebDB Workshop at EDBT, 1998.

[8] Z.Chen and H.Ji, "Can One Language Bootstrap the Other: A Case Study on Event Extraction", Proc. HLT-NAACL, Workshop on Semi-supervised Learning for Natural Language Processing. Boulder, 2009.

[9] A.Elsebai, F.Meziane, and F.Z.BelKredim, "A Rule Based Persons Names Arabic Extraction System", in Communications of the IBIMA Volume 11, ISSN: 1943-7765, 2009.

[10] G.Forman and M.Scholz, "Apples-to-Apples in Cross-Validation Studies: Pitfalls in Classifier Performance Measurement" , HP technical Reports, HPL-2009-359, 2009.

[11] I.Guyon and A.Elisseeff, "An Introduction to Variable and Feature Selection", Journal of Machine Learning Research 3 (2003) 1157-1182, 2003.

[12] A.Hassan, H.Fahmy, and H.Hassan, "Improving Named Entity Translation by Exploiting Comparable and Parallel Corpora", RANLP, AMML Workshop, 2007.

[13] H.Hassan and J.Sorensen, "An Integrated Approach for Arabic-English Named Entity Translation", Proceedings of the ACL Workshop on Computational Approaches to Semitic Languages, 2005.

[14] H.Ji and R.Grishman, "Data Selection in Semi-supervised Learning for Name Tagging", In Proceedings of COLING/ACL 06 Workshop on Information Extraction Beyond Document, Sydney, Australia, 2006.

[15] Z.Kozareva, "Bootstrapping Named Entity Recognition with Automatically Generated Gazetteer Lists", In Proceedings of EACL 2006, Trento, Italy, April 2006.

[16] T.Kudo and Y.Matsumoto, "Use of Support Vector Learning for Chunk Identification", CoNLL-2000.

[17] J.Lafferty, A.McCallum, and F.Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data", In Proc. of ICML, pp.282-289, 2001.

[18] J.Maloney and M.Niv, "TAGARAB: A Fast, Accurate Arabic Name Recognizer Using High Precision Morphological Analysis", In Proceedings of the Workshop on Computational Approaches to Semitic Languages, Canada, 1998.

[19] D.Samy, A.Moreno, and J.M.Guirao, "A Proposal For An Arabic Named Entity Tagger Leveraging a Parallel Corpus (Spanish-Arabic)", In Proceedings of Recent Advances in Natural Language Processing RANLP, Borovets, Bulgaria, pp. 459-465. 2005.

[20] K.Shaalan and H.Raza, "NERA: Named Entity Recognition for Arabic", The Journal of the American Society for Information Science and Technology (JASIST), John Wiley & Sons, Inc., NJ, USA, 60(8): 1652–1663, 2009.

[21] D.Wu, W.S.Lee, N.Ye, and H.L.Chieu, "Domain Adaptive Bootstrapping for Named Entity Recognition", Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, pages 1523–1532, Singapore, ACL and AFNLP, 2009.

[22] J.Zhu, Z.Nie, X.Liu, B.Zhang, and J.Wen, "StatSnowball: A Statistical Approach to Extracting Entity Relationships", WWW 2009 Madrid, 2009.