Received: October 30, 2017

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Arabic Named Entity Recognition Using Topic Modeling

Ismail El Bazi[1]* **Nabil Laachfoubi[1]**

[1]*Computer, Networks, Mobility and Modeling Laboratory, Department of Mathematics and Computer, FST, Univ Hassan 1st, Settat, Morocco*
* Corresponding author's Email: ismailelbazi@gmail.com

Abstract: In this article, we introduce novel features for Arabic Named Entity Recognition (NER) based on Latent Dirichlet Allocation (LDA), a widely used topic modeling technique. We investigate and analyze three different approaches for utilizing LDA, including two newly proposed ones, namely Topical Prototypes approach and Topical Word Embeddings approach. Our Experiments show that each of the presented approaches improves the baseline features, among which the Word-Class LDA approach performs the best. Moreover, the combination of these topic modeling approaches provides additive improvements, outperforming traditional word representations as Skip-gram word embeddings and Brown Clustering. The proposed LDA-based features, learned in an unsupervised way, are fully language-independent and have proven to be very effective to enrich and boost NER models for Arabic, a morphologically rich language.

Keywords: Arabic, Distributional semantic, Named entity recognition, LDA, Topic modeling.

1. Introduction

Generally, a supervised Named Entity Recognition (NER) model needs huge amounts of manually annotated data and an appropriate feature set in order to achieve good performance. Since the creation of very large annotated corpora is a fastidious task and will need much time and resources, recent research has focused on taking advantage of large-scale unlabeled text data, freely available, to learn word representations in an unsupervised way and exploit it as features to boost supervised NER systems [1].

There are two major categories of word representations: word clusters (WC) and word embeddings (WE). Word clusters are groups of words which ideally include semantically similar words. A typical algorithm to induce WC is Brown clustering (BC) [2]. Brown clusters were successfully applied in semi-supervised NER [3]. More recently, the focus has switched to word embeddings, a new type of word representations. Word embeddings are continuous, real-valued and finite dimensional vector representations of words. Typically, WE can be induced from massive unlabeled text corpora through two main approaches: neural network based models (e.g., Skip-gram and CBOW models) and matrix factorization models (e.g., Canonical Correlation Analysis and Principal Component Analysis). Word embeddings were also successfully applied in semi-supervised NER [4].

Another interesting way to exploit large unlabeled text data is Distributional Semantic Models (DSM). DSMs are based upon the distributional hypothesis [5] which assumes that "You shall know a word by the company it keeps.". In other words, we can determine the meaning of a token using the words in the context of which it often occurs. These models are high-dimensional vector representations of word meanings obtained by automatically collecting word-context co-occurrence statistics for each word in a corpus of textual data.

DSMs can be broadly subdivided into two families, context word methods and context region methods [6].

The context word methods use only a short context within a moving window nearby the word to induce its semantics. Typically, this short context is defined only by words surrounding the processed word. HAL [7] and BEAGLE [8] are examples of context word models.

The context region models use the whole document in which a word occurs as the context to learn its semantics. This large context can range from a sentence or a paragraph to an entire text corpus. Latent Semantic Analysis (LSA) [9] and Latent Dirichlet Allocation (LDA) [10] are typical examples of context region models.

In the present paper, we are interested in using LDA, the state-of-the-art probabilistic topic modeling technique, as features to boost supervised NER models. We chose Arabic, a highly agglutinative and inflectional language which suffers from data sparseness as the target language of our study.

In previous works, LDA was included as features mainly in two ways: (i) the direct use of a topic probability for the classifier [11], (ii) as word class clusters induced via soft clustering [12].

In our study, we propose two novel approaches for utilizing topic models as features, namely Topical Word Embeddings (TWE) features, and Topical Prototypes (TP) features. We carefully investigate their impact on the Arabic NER task and analyze if the proposed approaches are complementary to each other or not. Moreover, we also compare these LDA features with two traditional word representations: Brown clustering and Skip-gram word embeddings.

Section 2 provides a review of the earlier work about NER and the use of LDA-based features. Section 3 outlines the three approaches for utilizing topic modeling features that we choose to investigate in our study. In Section 4, we report the experiments and also show and discuss our results.

2. Related work

A myriad of Machine Learning (ML) algorithms has been applied to the NER task. The most commonly used ones are Conditional Random Field (CRF), Support Vector Machine (SVM), and Perceptron. Nevertheless, the performance of these models heavily relies on hand-crafted features, which is often challenging and time consuming to develop and maintain and require a lot of domain knowledge expertise.

Recently, Neural Networks (NNs) have been shown to be very effective for linguistic sequence labeling tasks, such as NER. In contrast with traditional ML approaches, NNs have the ability to extract effective features automatically from the training dataset without the need of human intervention, instead of relying on handcrafted features. A wide variety of neural network architectures have previously been proposed for NER. For instance, Collobert et al. [13] designed a Convolutional Neural Networks (CNNs) with a CRF output layer and achieved very promising results on various sequence tagging tasks such as POS tagging, chunking (CHUNK), NER and semantic role labeling (SRL). One major contribution of Collobert's work is to create a system that automatically learned internal representations from a huge amount of unlabeled corpora that can be efficiently exploited by all NLP tasks without the use of any task-specific engineering. Huang et al. [14] introduced Long Short-Term Memory (LSTM) based architectures for sequence tagging composed of a bidirectional LSTM (BiLSTM) and a CRF prediction layer (BiLSTM-CRF). Their system was robust and had less dependence on word representations but hinged on handcrafted spelling features to achieve state-of-the-art accuracy on POS, CHUNK and NER benchmarks. Inspired by [13], Chiu and Nichols [15] proposed a hybrid architecture that combines bidirectional LSTM and CNNs to model both character- and word-level features. They evaluated their model on CoNLL-2003 and OntoNotes 5.0 NER datasets and presented competitive results using capitalization, suffix, POS tags, and lexicon features. One limitation of this work is that it used task-specific special features to improve their results.

One of the first truly end-to-end neural network system was introduced by Lample et al. [16]. It was quite similar to [14] but without the use of any man-made spelling features. Their BiLSTM-CRF model relied on character-based and word representations with dropout regularization to obtain state-of-the-art performance in four languages, namely English, German, Polish, Dutch, and Spanish. Ma and Hovy [17] employed CNNs instead of LSTMs to create character-level representation in an end-to-end BiLSTM-CNNs-CRF architecture and achieved state-of-the-art performance on POS tagging and CoNLL NER without any data preprocessing or feature engineering. Yang et al. [18] presented a neural architecture similar to the one in [16] but replaced LSTM with Gated Recurrent Unit (GRU) network. Their model employed a hierarchical GRU to encode both word-level and character-level embeddings and obtained state-of-the-art results on POS tagging, chunking, and NER, in multiple languages.

More recently, Strubell et al. [19] proposed Iterated Dilated Convolutional Neural Networks (ID-CNNs) as a faster alternative to Bi-LSTMs for sequence labeling. While the popular Recurrent Neural Networks (RNNs) like LSTMs and GRUs are expressive and accurate, they did not fully exploit GPU parallelism opportunities, and thus their speed and computational efficiency are limited. Rather, ID-CNNs have clear computational advantages since they efficiently use the GPU parallel computation and minimize the training and testing time. Their ID-CNNs-CRF proposed model was 14 times faster than the Bi-LSTM-CRF at test time while retaining comparable accuracy. Moreover, Aguilar et al. [20] presented a novel multi-task approach to tackle the challenging task of NER for social media data. They adopted a neural network architecture composed of a CNNs component to capture orthographic features at the character level and a BiLSTM component to learning contextual and syntactical information at the word level. Once trained, the NN was used as a feature extractor to feed a CRF classifier. Their model obtained the first position in the 3rd Workshop on Noisy User-generated Text (WNUT-2017) with an entity F1-score of 41.86% and a surface F1-score of 40.24%.

Concerning topic modeling, very few works have explored the effect of topic modeling on the NER task. Chrupala [12] proposed the use of Latent Dirichlet Allocation in order to induce soft, probabilistic word classes and have shown that plugging automatically and efficiently induced LDA word class features to NER task can achieve better results in comparison with Brown clustering while scaling linearly with the number of classes. In the same way, as in [12], Tkachenko and Simanovsky [21] applied LDA to create word class features and compare them with various features on three well known English benchmarks: OntoNotes version 4, CoNLL 2003, and NLPBA 2004 dataset. Similarly, the same approach was successfully used with Mongolian NER [22].

In [11], Konkol and colleagues introduced new features specific for NER based on latent semantics. LDA are among these new features. They incorporated LDA directly as a feature to the classifier using the topic's probability and also explored the effect of stemming a preprocessing step for LDA. Their experiments have shown that LDA feature improved the baseline for all the four languages: English, Spanish, Dutch and Czech and that the use of stemming was more helpful for highly inflectional languages like Czech. Surprisingly, the best LDA results were achieved using smaller values of topics: 20 topics for stem based LDA and 50 topics for word based LDA.

On Social Media for English, Ritter et al. [23] introduced a novel approach to distant supervision using topic models. They applied LabeledLDA [24] as a distant supervision approach based on Freebase dictionaries to label named entities in twitter text and obtained 25% increase in the F1 measure over co-training approach. More recently, Jansson and Liu [25] explored a new approach that combines Deep Learning (DL) with LDA topic modeling. DL architecture was composed of two-layer bidirectional LSTM and a CRF output layer, while the online LDA method was applied to generate a topic representation for each tweet which was used as features for the DL model.

As far as we are aware, this is the first work that uses LDA in the context of Arabic NER.

3. Approaches for utilizing topic modeling features

This section describes the three proposed approaches of including LDA as feature to the Arabic NER task that we investigated in this paper.

3.1 Word-class LDA

Latent Dirichlet Allocation was initially suggested by Blei et al. [10] for topic modeling. The idea behind LDA is to find coherent topics shared among subsets of a collection of documents. LDA is a generative probabilistic model which induces a group of hidden topics. Each topic is described by a multinomial distribution over word types in the corpus. The graphical representation of LDA is presented in plate notation in Fig. 1.

For each document d in a corpus D and K latent topics, the generative process of the LDA model is formalized as follows:

$$\begin{aligned}
&\boldsymbol{\varphi_k} \approx \mathbf{Dirichlet}(\boldsymbol{\beta}),\ \ \mathbf{k} \in [\mathbf{1}, \mathbf{K}] \\
&\boldsymbol{\theta_d} \approx \mathbf{Dirichlet}(\boldsymbol{\alpha}),\ \ \mathbf{d} \in [\mathbf{1}, \mathbf{D}] \\
&\mathbf{Z_{n_d}} \approx \mathbf{Multinomial}(\boldsymbol{\theta_d}),\ \ \mathbf{n_d} \in [\mathbf{1}, \mathbf{N_d}] \\
&\mathbf{W_{n_d}} \approx \mathbf{Multinomial}\left(\boldsymbol{\varphi_{Z_{n_d}}}\right),\ \ \mathbf{n_d} \in [\mathbf{1}, \mathbf{N_d}]
\end{aligned} \tag{1}$$

The random variable φ_k represents probabilities of words in topic k. The variable θ_d represents probabilities distribution over topics for document d. the parameters α and β are hyper parameters of the Dirichlet distributions, where α represents the prior weight related to document-topic density and β

represents the prior weight related to topic-word

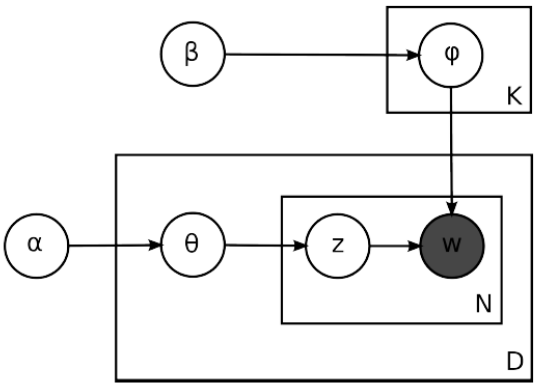

Figure.1 Graphical model for LDA [12]

Table 1. LDA/Word-Class mapping

Standard LDA	Word-Class LDA
Document	Word type
Word	Context feature
Topic	Word class

density. The variable Z_{n_d} is the topic assignment for word n in the document d. The variable W_{n_d} is the observed word for position n in the document d.

Chrupala [12] proposed a probabilistic soft word-class model which is based on LDA where each word type is associated with a probabilities distribution over latent word classes and each class is a distribution over contextually co-occurring features. A direct mapping between the standard LDA and the Word-Class LDA model initiated by Chrupala can be shown in table 1.

He interprets the generative process of the standard LDA topic model presented in Eq. (1) as follows:

- K : number of latent word classes,
- D : size of the vocabulary,
- N_d : number of right and left contexts in which word token d occurs,
- Z_{n_d} : class assigned to word token d in the n^{th}_d context,
- W_{n_d} : n^{th}_d context feature of word token d.

For the learning of the LDA model, he uses Gibbs Sampling method to estimate two sets of word representations: the θ_d parameters represent the word class probability distribution given a word token, while the φ_k represent the feature distribution given a word class. Thus this soft word-class model is a more expressive representation than hard word clustering:

(i) Soft LDA-based clustering can successfully model shared ambiguities,
(ii) It provides an additional source of external knowledge which helps find the class of a word based on its context,
(iii) It allows the expression of graded similarity between word types,
(iv) Training of soft LDA-based clustering is much faster in comparison with hard clustering approaches.

3.2 Topical word embeddings

Topical Word Embeddings is a multi-prototype word embedding framework introduced by Liu et al. [26] which uses topic modeling to learn embeddings based on both words and their topics. This allows for each word to have different embeddings under different topics in contrast with a typical word embeddings where we represent each word by a single vector.

TWE performs LDA with Gibbs Sampling to obtain word topics: given a sequence of words $D = \{w_1,\ldots, w_M\}$, each word token w_i is assigned into a specific topic z_i and the word-topic pair $< w_i , z_i >$ is used to learn topical word embeddings. Liu et al. extended the popular word embedding Skip-gram [27] to implements TWE models. They proposed three TWE models to induce topical word vectors: TWE-1, TWE-2 and TWE-3.

TWE-1. Each topic is considered as a pseudo word, and we learn topic representations and word representations separately and simultaneously, then we build topical word embeddings of $< w_i , z_i >$ by concatenating the embedding of w_i and z_i.

TWE-2. We consider each word-topic pair $< w_i , z_i >$ as a pseudo word and induce topical word embeddings directly as a unique vector. Each word-topic pair can have their own parameters.

TWE-3. Similar to TWE-1, we have distinct vectors for each word and each topic, but the length of word vectors and topic vectors are not necessarily the same. The embedding of each word-topic pair is built using the concatenation of the corresponding word vector and the topic vector, for learning. The parameters of each word vector w_i and topic vector z_i are the same of all word-topic pairs.

The computational complexity of the TWE models is reported in table 2, where W is the vocabulary size, C the window size, M the corpus length, T the number of topics, K_W the word vectors

length and K_T the topic vectors length. For Skip-Gram, TWE-1 and TWE-2: $K_W = K_T = K$

Table 2. Model Complexities [26]

Model	Model Parameters	Computational Complexity
Skip-Gram	WK	CM(1 + logW)
TWE-1	(W + T)K	IM + 2CM(1 + logW)
TWE-2	WTK	IM + CM(1 + logWT)
TWE-3	WKW + TKT	IM + CM(1 + logWT)

Table 3. AQMAR Arabic prototypes

Entity Class	Prototypes
B-LOC	أمريكا, مصر, القدس, دمشق, البرتغال
I-LOC	أفريقيا, المتحدة, الجنوبية, المنورة, المقدسة
B-PER	محمد , لويس, أحمد, رونالدو, الرازي
I-PER	أبي, طولون, بكر, عبد, بن
B-ORG	ريال, اتحاد, الفيفا, نادي, منتخب
I-ORG	لشبونة, سبورتنغ, مدريد, البرتغالي, يونايتد
B-MISC	الشابكة, الإلكترونات, اليورانيوم, الميكانيكا, البروتونات
I-MISC	الإدخال, الكلاسيكية, التشغيل, الصليبية, الإخراج
O	و, من, في , . ,،

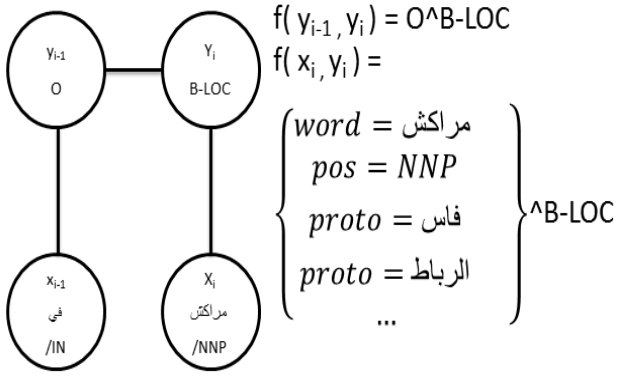

Figure.2 An example of prototype features for NER

In this study, we chose to use only TWE-1 and TWE-3 since they produce the best results.

3.3 Topical prototypes

We propose a novel method of including topic models as features for supervised models. Topical Prototypes are mainly inspired by the work of Guo et al. [28] and prototype-driven learning [29] which was initially introduced by Haghighi and Klein for unsupervised sequence modeling.

Influenced by prototype-driven learning, Guo et al. proposed distributional prototype approach as a new way of utilizing the word embedding features in semi-supervised learning.

Encouraged by the great potential of this approach on word embeddings, we decided to investigate and propose a similar approach based on topic modeling.

The basic idea behind the topical prototypes is that similar words have a higher probability to be tagged with the same entity label. For example, Rabat, Cairo, and Istanbul are more likely to be tagged as a Location entity. Hence, it is convenient to classify and select a group of representative words of each entity label (prototypes) in order to use them to link similar words to the same prototypes using distributional similarity metrics.
To compute the topical prototype features, three steps are needed:

1. Perform LDA on a big text corpus and get topic distributions.
2. Given an annotated training corpus, calculate the Normalized Pointwise Mutual Information (NPMI) of each label l and all words w in the vocabulary (Eq. (2) and Eq. (3)), and choose the top m words as the prototypes of l.
3. Introduce the prototypes as features to our supervised model. For each word w in the annotated training corpus, calculate the Hellinger distance between w and all the prototypes using the associated topic distributions generated in the first step. If the Hellinger distance is above the predefined threshold (usually 0.5), those prototypes will be selected as the prototype features of the processed word. An illustration of prototype features is given in Fig. 2.

$$\delta_n(l,w) = \frac{\delta(l,w)}{-\ln p(l,w)} \quad (2)$$

$$\delta(l,w) = \ln \frac{p(l,w)}{p(l)p(w)} \quad (3)$$

Table 3 shows the top five prototypes extracted from the AQMAR training set using NPMI [30].

4. Experiments

4.1 NER model

Our NER model is a Conditional Random Fields classifier, which is considered as the state-of-the-art model for NER by many authors. CRFs are first-order linear-chain graphical models that estimate directly the conditional probabilities p(y|x) of a state

(label) sequence $y=y_1,\ldots,y_t$ given an observation (word) sequence $x=x_1,\ldots,x_t$ as :

$$p(y|x) = \frac{1}{Z(x)} \exp \sum_{i=1}^{t} \sum_{j=1}^{m} \lambda_j f_j(y_i, y_{i-1}, x_i) \quad (4)$$

Where:
Z(x): a normalization constant, which sums over all state sequences for the word sequence x,
t : the number of word tokens in the input sequence x,
m: the number of feature functions f_j,
$\lambda_j=\lambda_1,\ldots, \lambda_m$ are real-valued parameters of the model,
$f_j(y_i,y_{i-1},x_i)$ are real-valued feature functions.

As for Maximum Entropy classifiers, the parameters λ_j are estimated using a standard maximum conditional log-likelihood approach with a regularization term as a measure to reduce over-fitting.

Typically, the feature functions f_j are binary. An example of a feature function in the context of NER is given by Eq. (5).

$$f_j(y_i, y_{i-1}, x_i) = \begin{cases} 1 & x_i = 'Marrakesh' \ \wedge \\ & y_{i-1} = 'O' \ \wedge \\ & y_i = ' B-LOC' \\ 0 & otherwise \end{cases} \quad (5)$$

Our CRF classifier is built upon a set of standard baseline features. In order to exploit efficiently the large-scale unlabeled text data available, we add new features based on topic modeling technique LDA to our baseline. We chose CRFsuite package a fast implementation of CRF provided by Naoaki Okazaki [1], to implement our NER model.

4.2 Baseline features

Our baseline feature set was defined over a window of ± 1 token. It includes many features that have been found to work well for Arabic. The feature set for each word token was:

- The word token itself.
- Part-of-speech tag.
- Affixes: Prefixes and suffixes of length from 1 to 4 are extracted from the processed word.
- Morphological Features: aspect, case, gender, number, and NormWord. They were generated using the MADA toolkit and already available within AQMAR corpus.

Table 4. Morphological features

Feature	**Feature Values**
Aspect	Verb aspect: Perfective, Imperfective, Command, Not applicable
Case	Grammatical case: Genitive, Accusative, Nominative, Not applicable, Undefined
Gender	Nominal Gender: Masculine, Feminine, Not applicable
Number	Grammatical number: Plural, Dual, Singular, Not applicable, Undefined
NormWord	Normalized spelling of the word form (romanized)

Table 5. Corpus Statistics for AQMAR Dataset

	documents	**words**	**sentences**	**entities**
Test	20	52,650	1,976	3,781
Dev	8	21,203	711	2,073

The value definition of these features is provided in Table 4.

4.3 Corpus

The Arabic Wikipedia Named Entity Corpus (AQMAR) is a small hand-annotated corpus of 28 Arabic Wikipedia articles for Arabic named entities [30]. Each article was annotated by 1 of 2 annotators with the traditional four entity classes: Person, Organization, Location, and generic Miscellaneous (MIS) following the BIO tagging format. The AQMAR corpus consists of 74000 tokens and 2687 sentences.

In this study, we used the test part as the training corpus. We divide the development part into half; one was used as development corpus and the other as a testing corpus. Additional information about AQMAR is shown in Table 5.

4.4 Experimental setting

We take the Arabic Wikipedia[2] until December 2016 as our unlabeled data to train all the types of topic modeling representations. The pre-processing was conducted using Gensim library[3] by removing

[1] http://www.chokkan.org/software/crfsuite/

[2] https://dumps.wikimedia.org/arwiki/

[3] https://radimrehurek.com/gensim/

Table 6. The Performance of NER on the AQMAR Test Data with Number of Topics K ∈ {50,100}

	K=50	**K=100**
Features	**F1**	**F1**
Baseline	69,18	69,18
+ TWE1	70,1	69,61
+ TWE3	70,18	70,26
+ TopicPrototype	72,15	72,26
+ Word-Class-LDA	**72,69**	**72,91**
+ TWE1 + TopicPrototype	72,76	72,56
+ TWE3 + TopicPrototype	72,39	72,65
+ TWE1 + Word-Class-LDA	72,81	73,03
+ TWE3 + Word-Class-LDA	72,72	73,03
+ Word-Class-LDA + TopicPrototype	**72,86**	**73,2**
+ SGNS	69,61	69,61
+ Cluster-SGNS	71,94	71,94
+ Brown	72,9	72,9
+ SGNS + Word-Class-LDA + TopicPrototype	72,91	73,04
+ Cluster-SGNS + Word-Class-LDA + TopicPrototype	72,2	73,29
+ Brown + Word-Class-LDA + TopicPrototype	**73,85**	**73,5**

all MediaWiki markups and tokenizing the texts. We used the software package provided by Chrupala[4] to generate the Soft Word-Class LDA features. Setting the number of classes K ∈{50,100}, we perform1 000 passes of Gibbs sampling, set the LDA hyper-parameters to α=10/K and β = 0.01 and rank classes according to the posterior probability and add the 3 top ranked classes as a feature to our NER model.

For Topical Word Embeddings, we used the implementation proposed by the authors[5]. When learning TWE models, we set window size as 3 and the dimensions of both word and topic embeddings as the number of topics K ∈ {50,100}. For both TWE-1 and TWE-3, we obtain topical word embeddings via concatenation over the corresponding word embeddings and topic embeddings.

For Topical Prototypes features (TopicPrototype), with manual tuning, we set the number of prototype words (m) for each target label to 5 and the threshold to 0.05.

For comparison purposes, we chose two traditional word representations: Skip-Gram word embeddings, and Brown clustering.

We use the Gensim implementation of Skip-Gram Negative Sampling (SGNS) to induce the word embedding features and set the window size as 3 and the vector size to 200.

One way to better utilize the embeddings in the linear models is by clustering the word embeddings so we perform clustering on the SGNS embedding features via Sofia-ml toolkit and produce clustered embeddings features (Cluster-SGNS). We tune the number of clusters n from 100 to 1000 and use the combination of n =100, 200, 300, 400, 500, which achieves the best results.

Finally for Brown clustering features (Brown), we fix the number of word clusters to 500.

The training data of brown clustering is the same with that of training topic modeling representations and word embeddings.

4.5 Results and discussion

We choose the F-measure (F1) as the evaluation measure in all our experiments and use the standard conlleval[6] script to report the performance. The CoNNL evaluation is very restrictive in comparison with MUC or ACE evaluations. The named entity (NE) is correct, only if both the type and the boundary of the NE are tagged correctly. Other metrics have a more relaxed matching criterion by giving partial credit if the system matches only one of the NE attributes.

Table 6 shows the performances of NER on the AQMAR dataset. The best values are in bold.

As we can see, all of the three LDA-based approaches we investigate in this study improve the

[4] https://bitbucket.org/gchrupala/lda-wordclass/

[5] https://github.com/largelymfs/topical_word_embeddings

[6] http://www.cnts.ua.ac.be/conll2000/chunking/conlleval.txt

baseline. The best performance is obtained by the Word-Class LDA feature with an F-score improvement of 3.73% above the baseline. The Topical Prototypes features obtained a very competitive result with 3.08% improvement over the baseline. For Topical Word Embeddings features (TWE-1 and TWE-3), they outperformed the baseline with an average value of 0.86%.

We further combine the three LDA-based features to see if they are complementary to each other. As shown in Table 6, all the feature combinations improve the results which suggest that they are quite complementary. The most complementary ones are Word-Class features and Topical Prototypes features. By combining them, we further push the performance with nearly four points higher than the performance of the baseline features.

We also compare the proposed features with other classical word representations features. As depicted in Table 6, both Word-Class and Topical Prototypes outperform Word embeddings and clustered embedding features. The Word-Class features alone achieve comparable performance with Brown clusters. It is worth noting that a large number of classes (500) were needed for Brown Clustering, where a much lower number (100) was sufficient for Word-Class LDA to achieve similar results. When the Word-Class and Topical Prototypes features are used together, we outperform the Brown clusters. By combining these features with other word representation features we further improve the performance. The best results are achieved by combining Brown clusters with word-class and topical prototype features with 73.85% which nearly four and a half points higher that the baseline.

Our experiments on LDA features for NER are in line with previous works done for English, Spanish, Dutch and Czech languages, where they tried to incorporate LDA in NER either directly or using soft clustering [11, 12, 21]. The novelty of our approach is that we propose two new ways of using LDA for the NER task, the first one as a topical word embeddings and the second one as topical prototypes features. Moreover, we successfully apply it for Arabic, a morphologically rich language.

Since it is the first time that we use LDA in the context of Arabic, we were not able to compare our results with other Arabic NER systems.

Generally, the empirical results confirm that the use of LDA-based features is beneficial and can significantly boost the performance of Arabic NER systems. Moreover, the combination of introduced features and traditional word representation features as the word embeddings and Brown Clustering further improve the performance. For a morphologically rich language as Arabic, such combinations of features inferred in an unsupervised manner from a large-scale unlabeled text data, constitute a solid feature set which we can exploit to create state-of-the-art semi-supervised NER models. It is also interesting to note that the proposed LDA-based features are fully language independent and can be used efficiently by any languages, especially the Low-Resource ones.

5. Conclusion and future work

This paper explores the use of topic models as features for Arabic NER system. We present three different topic modeling approaches for a careful comparison and analysis. Using any of the three features, we obtain higher performance than the baseline, among which word-class and topical prototypes features perform the best. Moreover, the combination of these newly proposed features provides significant additive improvements and achieves 73.2 in F-measure which is four points over the baseline. The main contribution of our work lies in a successful design of novel features based on LDA. We believe that such features have not yet been investigated in the NER task, especially in the context of Arabic language. Interestingly, our experimental results support the idea that LDA features are an efficient and attractive choice for semi-supervised learning in comparison with traditional Brown clustering and could be very useful for boosting the performance of NER models for Arabic and also other Low-Resource languages since these features are mainly based on large-scale unlabeled text data, easily available for all languages.

In the future, we will study if there are more appropriate ways of including LDA into NER than the ones proposed in this paper. It would be also interesting to investigate whether the hierarchical clustering of Word-Class distributions [31] can be used successfully as features in a NER scenario. Another possible improvement of our system is applying the approaches introduced in [28] on the dense and continuous TWE embedding features.

References

[1] I. El bazi and N. Laachfoubi, “Arabic Named Entity Recognition using Word

Representations," *International Journal of Computer Science and Information Security*, Vol. 14, No. 8, p. 956, 2016.

[2] P. Liang, "Semi-supervised learning for natural language," *Massachusetts Institute of Technology*, 2005.

[3] L. Ratinov and D. Roth, "Design challenges and misconceptions in named entity recognition", In: *Proc. of the Thirteenth Conference on Computational Natural Language Learning*, pp. 147–155, 2009.

[4] J. Turian, L. Ratinov, and Y. Bengio, "Word representations: a simple and general method for semi-supervised learning", In: *Proc. of the 48th annual meeting of the association for computational linguistics*, pp. 384–394, 2010.

[5] J. R. Firth, *A synopsis of linguistic theory, 1930-1955*, Studies in linguistic analysis, 1957.

[6] B. Riordan and M. N. Jones, "Redundancy in Perceptual and Linguistic Experience: Comparing Feature-Based and Distributional Models of Semantic Representation," *Topics in Cognitive Science*, Vol. 3, No. 2, pp. 303–345, 2011.

[7] K. Lund and C. Burgess, "Producing high-dimensional semantic spaces from lexical co-occurrence", *Behavior Research Methods, Instruments, & Computers*, Vol. 28, No. 2, pp. 203–208, 1996.

[8] M. N. Jones and D. J. Mewhort, "Representing word meaning and order information in a composite holographic lexicon", *Psychological Review*, Vol. 114, No. 1, p. 1, 2007.

[9] T. K. Landauer and S. T. Dumais, "A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge", *Psychological Review*, Vol. 104, No. 2, p. 211, 1997.

[10] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet Allocation", *J. Mach. Learn. Res.*, Vol. 3, pp. 993–1022, 2003.

[11] M. Konkol, T. Brychcın, and M. Konopık, "Latent semantics in named entity recognition", *Expert Systems with Applications*, Vol. 42, No. 7, pp. 3470–3479, 2015.

[12] G. Chrupala, "Efficient induction of probabilistic word classes with LDA", In : *Proc. of 5th International Joint Conference on Natural Language Processing*, pp. 363–372, 2011.

[13] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch", *Journal of Machine Learning Research*, Vol. 12, No. Aug, pp. 2493–2537, 2011.

[14] Z. Huang, W. Xu, and K. Yu, "Bidirectional LSTM-CRF models for sequence tagging", *arXiv preprint arXiv:1508.01991*, 2015.

[15] J. P. Chiu and E. Nichols, "Named entity recognition with bidirectional LSTM-CNNs", *arXiv preprint arXiv:1511.08308*, 2015.

[16] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer, "Neural architectures for named entity recognition", *arXiv preprint arXiv:1603.01360*, 2016.

[17] X. Ma and E. Hovy, "End-to-end sequence labeling via bi-directional lstm-cnns-crf", *arXiv preprint arXiv:1603.01354*, 2016.

[18] Z. Yang, R. Salakhutdinov, and W. Cohen, "Multi-task cross-lingual sequence tagging from scratch", *arXiv preprint arXiv:1603.06270*, 2016.

[19] E. Strubell, P. Verga, D. Belanger, and A. McCallum, "Fast and Accurate Entity Recognition with Iterated Dilated Convolutions", In: *Proc. of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 2660–2670, 2017.

[20] G. Aguilar, S. Maharjan, A. P. L. Monroy, and T. Solorio, "A Multi-task Approach for Named Entity Recognition in Social Media Data", In: *Proc. of the 3rd Workshop on Noisy User-generated Text*, pp. 148–153, 2017.

[21] M. Tkachenko and A. Simanovsky, "Named entity recognition: Exploring features", In: *Proc. of KONVENS 2012*, pp. 118–127, 2012.

[22] W. Wang, F. Bao, and G. Gao, "Mongolian Named Entity Recognition System with Rich Features", In: *Proc. of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pp. 505–512, 2016.

[23] A. Ritter, S. Clark, Mausam, and O. Etzioni, "Named Entity Recognition in Tweets: An Experimental Study", In: *Proc. of the Conference on Empirical Methods in Natural Language Processing*, pp. 1524–1534, 2011.

[24] D. Ramage, D. Hall, R. Nallapati, and C. D. Manning, "Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora", In: *Proc. of the 2009 Conference on Empirical Methods in Natural Language Processing*, Vol. *1*, pp. 248–256, 2009.

[25] P. Jansson and S. Liu, "Distributed Representation, LDA Topic Modelling and Deep Learning for Emerging Named Entity Recognition from Social Media", In: *Proc. of the 3rd Workshop on Noisy User-generated Text*, pp. 154–159, 2017.

[26] Y. Liu, Z. Liu, T.-S. Chua, and M. Sun, "Topical Word Embeddings." , In: *Proc. of AAAI*, pp. 2418–2424, 2015.
[27] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space", *arXiv preprint arXiv:1301.3781*, 2013.
[28] J. Guo, W. Che, H. Wang, and T. Liu, "Revisiting Embedding Features for Simple Semi-supervised Learning", In: *Proc. of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 110–120, 2014.
[29] A. Haghighi and D. Klein, "Prototype-driven Learning for Sequence Models", In: *Proc. of the Main Conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, pp. 320–327, 2006.
[30] B. Mohit, N. Schneider, R. Bhowmick, K. Oflazer, and N. A. Smith, "Recall-oriented Learning of Named Entities in Arabic Wikipedia", In: *Proc. of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 162–173, 2012.
[31] G. Chrupala, "Hierarchical Clustering of Word Class Distributions", In: *Proc. of the NAACL-HLT Workshop on the Induction of Linguistic Structure*, pp. 100–104, 2012.