Survey on Named Entity Recognition

Dicle Öztürk (dicleozturk@gmail.com)

7 Aralık 2018

Contents

1 Problem Definition

Named entity recognition is the task for detecting the names for people, locations, organizations; temporal or numeric expressions. It can be treated as a word or phrase-level text classification / sequence tagging task. For example, in the sentence "Achille-Claude Debussy, born in 1862 in St. Germain and known since the 1890s, was a talented French composer.", ideally the named entities are Achille-Claude Debussy, 1862, St. Germain, 1890s and French.

It is considered a type of sequence tagging problem. The length of the tagged text varies; it might be multiple words, single word or a part of word depending on the tagging scheme chosen. Thus, classical binary learning methods for fixed sized instances may not work well. This effects dataset construction and learning model. The classification target can still be assummed to be words but it is completely dependent on its neighbour words, even further words.

It is one of the primary information extraction tasks and has many use cases in problems such as event detection, web search, translation, NLP for historical texts. More pervasively, it is quite useful in text classification. In sentiment analysis, for example, it helps detecting the sentiment towards an entity. In message classification and generation, it is useful for capturing what / whom the message is talking about and for generating a meaningful answer. Generally, feature extraction mechanisms can make use of named entities; for example, the number of named entities in a text might indicate the formality level of it and can contribute to the overall performance when made use of.

2 Datasets for NER

Datasets for NER should be tagged for named entities in it. For this, certain set of tags and their possible members should be available.

Shared tasks in three conferences, MUC 1997[1], CoNLL 2002[2] and 2003[3], appear to have set the standards for named entity categories. In MUC 1997 there are three groups with their own subcategories: ENAMEX has PERSON, ORGANIZATION, LOCATION classes; TIMEX has DATE and TIME classes; and lastly NUMEX has MONEY and PERCENTAGE classes. And in CoNLL tasks, there four categories: PERSON, LOCATION, ORGANIZATION and MISCELLENEOUS (for any other name special but not in the previous three ones)[4]. In this regard, the annotated dataset of CoNLL 2003 shared task is assummed the gold standard for English, Spanish, Dutch and German [1]. The texts are news articles.

The datasets are mostly news texts, Wikipedia articles or some other in-domain texts. Recently, texts from social media like tweets and blogposts are also annotated and studied for NER like [2, 3]. As the informal voice of these texts as well as potentially many unseen named entities encountered, NER on such texts are said to be harder [4].

Annotation efforts for NER are diverse and ongoing. While there are standard datasets, the set of entity names changes over time rapidly and even entity name categorization has not been quite settled. Although performance values of the systems are high, texts types may change over time and most NER systems are not very good at capturing unseen named entites [4]. Annotating for NER data is discussed in some papers [5, 6].

As a highly crowded special domain, named entity extraction in clinical and biomedical texts comprises a subfield; several studies can be found [7]

3 Learning Models

Basically, there are three methods for detecting named entities: 1) rule-based 2) statistical and recently 3) neural.

3.1 Rule-based approaches

In the rule-based methods [8], a huge list of named entities such as person names, organizations, countries and so on is prepared and the input text is searched for those entities with the help of some text processing methods like regular expressions or more sophisticated tools like POS taggers (to search for only nouns, for example).

[1] https://www-nlpir.nist.gov/related_projects/muc/proceedings/ne_task.html
[2] https://www.clips.uantwerpen.be/conll2002/ner/
[3] https://www.clips.uantwerpen.be/conll2003/ner/
[4] http://www.cnts.ua.ac.be/conll2003/ner/annotation.txt

3.2 Statistical approaches

In the statistical and neural methods, a model is built by determining features / using word vectors for how to represent the texts for named entities; a set of texts where the named entities are annotated is used as the training set and the words in it are input to a classifier as windows (by its neighbour words); the classifier learns the model with the examples in the training set, then this learned model can be used as named entity recognizer ready to accept new texts whose entities are unknown.

3.3 Neural approaches

Neural architectures for named entity recognition have increased attention by the rise of deep learning model. The first neural model on NER with near classical state-of-the-art performance, to our knowledge[5], was presented by [11] and [12], the latter being more influential as seems by the citation frequency over the papers that we went through. In [11], the authors model a NER system with a linear model but do not present any result, saying that they used a dataset which is non-standard. In [12], the authors give a full sketch of a neural NER system. The model is trained on standard CoNLL03 dataset [1]. In addition to three other useful NLP tasks, namely, POS tagging, chunking and semantic role labelling, the authors tune a convolutional neural network for the task of NER using word embeddings they themselves produced. No other features are used which is quite genuine. They obtained an f-score of 81.47%. However, when they use POS tags or gazetteers as additional features, the result rises up to almost 89%. A concrete discussion is not given but this should be due to the network architecture as we will see in the later neural models. The initial performance (81.47%) is a comparable performance to other works using the same dataset at that time. The all-outperformaning study [13] the authors mention has 89.31% f-score, which employs a linear model based on many hand-crafted features.

In a similar setting to [12]'s work, in [14], the authors employ a NER from scratch approach and produce NER systems for 40 languages without any labelled data; using only Wikipedia texts for word embedding production and Wikipedia entity names as labels. Another neural model similar to Colloberts' work is by [15]. The authors use BiLSTM networks, an extension of recurrent neural models to store relatively long (both forward and backward) dependencies or features in the text. Considering NER as a sequence tagging problem, long-ranging relations of the target sequence item (the word to be labelled) in the sentence, not just its connection to its immediate neighbours, give critical information. Their dataset is the standard CoNLL03 dataset [1]. The features are word embeddings and several orthographic features. They also employ gazetteers. As for word embeddings, they try randomly initialized vectors and the vectors of [12]. The authors try some extensions of BiLSTM, like adding a top CRF (Conditional Random Field) layer or using uni-directional LSTM, as well as trying only CRF. The best model is BiLSTM with CRF, giving an f-score of 90.10% with all the features ([12]'s embeddings, orthographic features in addition to gazetteers). Without gazetteers, the result drops by around 1.3 points. In the paper, we see that 1) CRF affects f-score remarkably (around 3 points). 2) Random embeddings drops the performance about 2-4 points. 3) The decrease in performance when no orthographic features used varies among models. CRF and LSTM depend on those features the most; their f-scores drop by around 8 points and BiLSTM (no CRF layer) depends least on external features; performance drop is only 1.6. Indeed, in this study, NER appears to be the task that relatively more depends on such external features compared to chunking and POS tagging.

In 2015, character embeddings have been introduced as an additional feature for neural NER models by [16]. A convolutional layer combine the features around each character and this is produced as the embedding of that character; thus, subword discriminatory information can be captured especially for morphologically rich languages and for languages with none or ambiguous word boundaries. On Spanish and Portuguese datasets, the authors show that word and character embeddings together outperform models using only word or only character embeddings. It is interesting that word embeddings with additional orthographic features perform almost the same with word+char embeddings on Portuguese dataset. The network architecture is convolutional.

Chiu and Nichols also study a similar model [17] but the overall architecture is BiLSTM while the character embeddings are here as well produced by a convolutional layer. They make use of capitalization feature and gazetteers. On CoNLL 2003 English dataset, using word and char embeddings along with gazetteer, they outperform all the known systems, giving an f-score of 91.62%. They also study the effect of different word embeddings and indicate that it effects the performance. There are several following studies on using word and character embeddings for NER, investigating the effect of different architectures, features, languages and datasets [18, 19, 20, 21, 22].

[5] There is a 2003 paper on NER modelled with LSTM [9]; the author reports an f-score of 60.15%. There is one more previous study we could find, from 2010, studying Chinese texts with character-level features training a deep belief network [10].

Some studies use only character embeddings and give f-scores around 80% for different datasets [23, 24]. A comparative study for word and character embeddings on Vietnamese texts can be found in [25].

Not just characters but also morpheme embeddings have also been studied for a better understanding of the problem. In [26], morpheme embeddings are produced in addition to word and character embeddings for Turkish and Czech texts. The system outperforms previous studies on Turkish texts, giving an f-score of 93.59% and is comparable for Czech. An extension is given in [27]. In a similar manner, in [28], the authors claim to use morphological features but they do not employ a parser for it; instead, make use of first and last 3 characters of the word and assume frequent ones as affixes. They give comparable performance to state-of-the-art models for many western languages.

Lastly, we would like to mention a study that produces phrase embeddings for NER [29]. The aim is said to be for better capturing syntactic and semantic relations. The proposed system gives comparable performance on CoNLL 2003 English dataset.

4 Evaluation Methods

The standard for NER is using F-score evaluation (conll)[6].

5 NER in Turkish Texts

The earliest NER studies on Turkish texts are [30], which is based on many hand-crafted features and statistical learning methods; and [31], which employs a language independent machine learning approach.

In [30], a purely rule-based system is presented. The authors extract lists of 1) person names 2) well-known people 3) well-known locations 4) well-known organizations from a large corpus, choosing only nouns from the output of a morphological analyser. And they build pattern bases for 1) location name (X Sokağı) 2) organization names (Y Üniversitesi) 3) temporal and numeric expressions. The tags they consider are ENAMEX, NUMEX and TIMEX. For the detection of named entities in a new text, their system searches for the match of name and pattern bases.

As for performance, the system gives an f-score of 78% on news texts and 55% on historical texts. The authors say that pattern matches for non-entity phrases are common and lead to poor performance. And in [32], this rule-based system is improved with some additional lexical resources.

Recent studies on Turkish NER make use of rather statistical and lately neural approaches. The systems mostly rely on the features mostly found in formal text while social media data or texts from other domains may not carry such features or might be very different stylistically [33] and to overcome such noise or to extend to other domains 1) normalization 2) domain adaptation is suggested [34][2]. Currently, using word embeddings beside named entity features is very popular and said to be overcoming such problems.

In [35], the authors extract various linguistic features for NER and for learning them use Conditional Random Fields, which is one of the top algorithms for sequence tagging [36]. In addition to morphological, lexical and symbolic features extracted, a list of names for entities (gazetteers) is also used. Their training set contains annotated news and tweets. The f-score performance of this system is 91%. Such a system is quite dependent on linguistic tools and language-specific features, which may not be suitable for the cases having unstructured or informal texts. Indeed, neural approaches give comparable / slightly higher results [37].

In [33], the neural network based approach of [12], which in both studies called "NLP from Scratch" approach, is applied for Turkish texts. First, the embedding vectors are produced, using the Skip-Gram approach [38], from a huge amount of texts (a huge news corpus combined with a dump of Turkish Wikipedia). As for preprocessing prior to embedding production, the urls and numbers are converted to their respective symbols; all the words are lowercased; they indicate that dismissing capitalization is proven to be more effective on social media data. In the learning phase, the authors build a window-based neural network which makes the assumption that the tag of the target word depends on the neighbouring words [12]. This system performs much better than the similar Turkish systems on the social media dataset but not on the news texts. Among the named entities that the system can detect, the authors say there are misspelled, asciified (ü->u) or extended words so text normalization phase seems not very critical once high quality embeddings are used.

Another study that uses embeddings is [3] where in addition to word vectors, some contextual linguistic and stylistic features are also used and for learning a window-based perceptron is applied. The domain is tweets for which NER is notably more difficult. The f-score performance of the system is 56% when trained on annotated news texts and 48% when trained only on annotated tweets.

---

[6] `http://www.cnts.ua.ac.be/conll2002/ner/bin/conlleval.txt`

Paper	Dataset	Learning approach / algorithm	Labels	Performance	Availability
[14]	Unannotated wikipedia	Embeddings and NN	Person, location, organization	None for not having annotated data	Yes
[33]	News, tweets, forum, speech	Embeddings and NN	Person, location, organization	83% for news, 57% for twitter (f-score)	No
[3]	News, tweets	Embeddings and perceptron	Person, location, organization	56% (f-score)	Yes (both data and executables)
[35]	News, tweets	CRFs	Person, location, organization, date, time, money, percentage	91% f-score	No, only online for one-time use.

Table 1: Sketch of the most recent Turkish NERs.

6 NER for Various Languages

We mentioned studies on western languages and Turkish. However, NER efforts are high for less known or under-resources languages. These studies present fruitful discussions for the difficulties of NER, which gets out by the different properties of the languages. There are studies on Latin [39], Kurdish [40, 41], Vietnamese [25], Armenian [42], Arabic [43], Tajik [41] and many more that we couldn't mention here.

7 Current Challenges and Promising Approaches

Primary challenges are studied in [4] live on available models. Beside, cross-corpus and cross-domain experiments, the authors indicate the importance of named entity diversity and detection of unseen named entities, generalization capacity of the systems. Thus, the systems should intensively be tested on different corpora, texts from different domains, types.

Entities change over time and capturing those is an important issue, as pointed out in [44]. Tag ambiguity is an important problem [45]. Parsing locational entities is another issue in this regard; recent studies seriously concentrate on it [46, 47].

Dealing with resource scarcity is another challenge [48, 49] as encountered in many other NLP studies.

References

[1] E. F. Tjong Kim Sang and F. De Meulder, "Introduction to the conll-2003 shared task: Language-independent named entity recognition," in *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pp. 142–147, Association for Computational Linguistics, 2003.

[2] L. Derczynski, K. Bontcheva, and I. Roberts, "Broad twitter corpus: A diverse named entity recognition resource," in *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pp. 1169–1179, 2016.

[3] E. O. Okur, H. Demir, and A. Özgür, "Named entity recognition on twitter for turkish using semi-supervised learning with word embeddings," *CoRR*, vol. abs/1810.08732, 2016.

[4] I. Augenstein, L. Derczynski, and K. Bontcheva, "Generalisation in named entity recognition: A quantitative analysis," *CoRR*, vol. abs/1701.02877, 2017.

[5] K. Fort, M. Ehrmann, and A. Nazarenko, "Towards a methodology for named entities annotation," in *Proceedings of the Third Linguistic Annotation Workshop*, pp. 142–145, Association for Computational Linguistics, 2009.

[6] K. Komiya, M. Suzuki, T. Iwakura, M. Sasaki, and H. Shinnou, "Comparison of annotating methods for named entity corpora," in *Proceedings of the 10th Linguistic Annotation Workshop held in conjunction with ACL 2016 (LAW-X 2016)*, pp. 59–67, 2016.

[7] X. Wang, C. Yang, and R. Guan, "A comparative study for biomedical named entity recognition," *International Journal of Machine Learning and Cybernetics*, vol. 9, no. 3, pp. 373–382, 2018.

[8] D. Nadeau and S. Sekine, "A survey of named entity recognition and classification," *Lingvisticae Investigationes*, vol. 30, 01 2007.

[9] J. Hammerton, "Named entity recognition with long short-term memory," in *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pp. 172–175, Association for Computational Linguistics, 2003.

[10] Y. Chen, Y. Ouyang, W. Li, D. Zheng, and T. Zhao, "Using deep belief nets for chinese named entity categorization," in *Proceedings of the 2010 Named Entities Workshop*, NEWS '10, (Stroudsburg, PA, USA), pp. 102–109, Association for Computational Linguistics, 2010.

[11] R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in *Proceedings of the 25th international conference on Machine learning*, pp. 160–167, ACM, 2008.

[12] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch," *J. Mach. Learn. Res.*, vol. 12, pp. 2493–2537, Nov. 2011.

[13] R. K. Ando and T. Zhang, "A framework for learning predictive structures from multiple tasks and unlabeled data," *Journal of Machine Learning Research*, vol. 6, no. Nov, pp. 1817–1853, 2005.

[14] R. Al-Rfou, V. Kulkarni, B. Perozzi, and S. Skiena, "POLYGLOT-NER: massive multilingual named entity recognition," *CoRR*, vol. abs/1410.3791, 2014.

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

[16] C. dos Santos and V. Guimarães, "Boosting named entity recognition with neural character embeddings," in *Proceedings of the Fifth Named Entity Workshop*, pp. 25–33, Association for Computational Linguistics, 2015.

[17] J. Chiu and E. Nichols, "Named entity recognition with bidirectional lstm-cnns," *Transactions of the Association for Computational Linguistics*, vol. 4, pp. 357–370, 2016.

[18] G. Luo, X. Huang, C.-Y. Lin, and Z. Nie, "Joint entity recognition and disambiguation," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 879–888, 2015.

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

[20] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer, "Neural architectures for named entity recognition," *CoRR*, vol. abs/1603.01360, 2016.

[21] A. Bharadwaj, D. Mortensen, C. Dyer, and J. Carbonell, "Phonologically aware neural model for named entity recognition in low resource transfer settings," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 1462–1472, 2016.

[22] F. Dernoncourt, J. Y. Lee, and P. Szolovits, "Neuroner: an easy-to-use program for named-entity recognition based on neural networks," *arXiv preprint arXiv:1705.05487*, 2017.

[23] O. Kuru, O. A. Can, and D. Yuret, "Charner: Character-level named entity recognition," in *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pp. 911–921, 2016.

[24] D. Gillick, C. Brunk, O. Vinyals, and A. Subramanya, "Multilingual language processing from bytes," *arXiv preprint arXiv:1512.00103*, 2015.

[25] T.-H. Pham and P. Le-Hong, "End-to-end recurrent neural network models for vietnamese named entity recognition: Word-level vs. character-level," in *International Conference of the Pacific Association for Computational Linguistics*, pp. 219–232, Springer, 2017.

[26] O. Gungor, E. Yildiz, S. Uskudarli, and T. Gungor, "Morphological embeddings for named entity recognition in morphologically rich languages," *arXiv preprint arXiv:1706.00506*, 2017.

[27] O. Güngör, S. Üsküdarlı, and T. Güngör, "Improving named entity recognition by jointly learning to disambiguate morphological tags," *arXiv preprint arXiv:1807.06683*, 2018.

[28] V. Yadav, R. Sharp, and S. Bethard, "Deep affix features improve neural named entity recognizers," in *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pp. 167–172, 2018.

[29] A. Passos, V. Kumar, and A. McCallum, "Lexicon infused phrase embeddings for named entity resolution," *arXiv preprint arXiv:1404.5367*, 2014.

[30] G. TÜR, D. HAKKANİ-TÜR, and K. OFLAZER, "A statistical information extraction system for turkish," *Natural Language Engineering*, vol. 9, no. 2, p. 181–210, 2003.

[31] S. Cucerzan and D. Yarowsky, "Language independent named entity recognition combining morphological and contextual evidence," in *1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, 1999.

[32] D. Küçük and A. Yazici, "A hybrid named entity recognizer for turkish," *Expert Syst. Appl.*, vol. 39, pp. 2733–2742, 02 2012.

[33] K. D. Onal and P. Karagoz, "Named entity recognition from scratch on social media," in *Proceedings of the 6th International Conference on Mining Ubiquitous and Social Environments - Volume 1521*, MUSE'15, (Aachen, Germany, Germany), pp. 2–17, CEUR-WS.org, 2015.

[34] J. Eisenstein, "What to do about bad language on the internet," in *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 359–369, Association for Computational Linguistics, 2013.

[35] G. Akın Şeker and G. Eryiğit, "Extending a crf-based named entity recognition model for turkish well formed text and user generated content1," *Semantic Web*, vol. 8, pp. 1–18, 01 2017.

[36] C. Sutton and A. McCallum, "An introduction to conditional random fields," *Foundations and Trends® in Machine Learning*, vol. 4, no. 4, pp. 267–373, 2012.

[37] H. Demir and A. Özgür, "Improving named entity recognition for morphologically rich languages using word embeddings," in *Proceedings - 2014 13th International Conference on Machine Learning and Applications, ICMLA 2014*, 12 2014.

[38] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *CoRR*, vol. abs/1301.3781, 2013.

[39] A. Erdmann, C. Brown, B. Joseph, M. Janse, P. Ajaka, M. Elsner, and M.-C. de Marneffe, "Challenges and solutions for latin named entity recognition," in *Proceedings of the Workshop on Language Technology Resources and Tools for Digital Humanities (LT4DH)*, pp. 85–93, 2016.

[40] H. Hassani, "A method for proper noun extraction in kurdish," in *OASIcs-OpenAccess Series in Informatics*, vol. 56, Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2017.

[41] P. Littell, K. Goyal, D. R. Mortensen, A. Little, C. Dyer, and L. Levin, "Named entity recognition for linguistic rapid response in low-resource languages: Sorani kurdish and tajik," in *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pp. 998–1006, 2016.

[42] T. Ghukasyan, G. Davtyan, K. Avetisyan, and I. Andrianov, "pioner: Datasets and baselines for armenian named entity recognition," *arXiv preprint arXiv:1810.08699*, 2018.

[43] K. Shaalan, "A survey of arabic named entity recognition and classification," *Computational Linguistics*, vol. 40, no. 2, pp. 469–510, 2014.

[44] H. Fromreide, D. Hovy, and A. Søgaard, "Crowdsourcing and annotating ner for twitter# drift.," in *LREC*, pp. 2544–2547, 2014.

[45] M. Marrero, J. Urbano, S. Sánchez-Cuadrado, J. Morato, and J. M. Gómez-Berbís, "Named entity recognition: fallacies, challenges and opportunities," *Computer Standards & Interfaces*, vol. 35, no. 5, pp. 482–489, 2013.

[46] M. Gritta, M. T. Pilehvar, N. Limsopatham, and N. Collier, "What's missing in geographical parsing?," *Language Resources and Evaluation*, vol. 52, pp. 603–623, Jun 2018.

[47] M. Gritta, M. T. Pilehvar, and N. Collier, "A pragmatic guide to geoparsing evaluation," *arXiv preprint arXiv:1810.12368*, 2018.

[48] B. Zhang, D. Lu, X. Pan, Y. Lin, H. Abudukelimu, H. Ji, and K. Knight, "Embracing non-traditional linguistic resources for low-resource language name tagging," in *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, vol. 1, pp. 362–372, 2017.

[49] J. Foley, S. M. Sarwar, and J. Allan, "Named entity recognition with extremely limited data," *arXiv preprint arXiv:1806.04411*, 2018.