Turkish Online Journal of Qualitative Inquiry (TOJQI)
Volume 12, Issue 6, June, 2021: 7590-7601

Flat and Nested Named Entity Recognition: A Review

Vanipriya C H[1], Subhashree M[1], Abhishek S G[1], Rakshitha J[1], Gaurav Gupta[2]

[1]SIR M Visvesvaraya Institute of Technology, Banglore
[2]Yogananda School of Artificial Intelligence Computers and Data Science,
Shoolini University of Biotechnology and Management Sciences, Solan-173229,
Himachal Pradesh, India

Abstract. The developments in Information Technology have significantly improved our lives. Information is at our fingertips; we can reach a wider audience and explore new things with ease. With all these made possible, there comes an issue that needs to be addressed: Information Overload. There is an extensive amount of information out there that has caused the extraction of relevant information arduous. Here is where the role of Named Entity Recognition (NER) comes into play. Identifying proper nouns from text and labeling them with their semantic type that belongs to a predefined list of classes is called NER. Moreover, NER is the initial step for various other applications in Natural Language Processing (NLP). With the continual endeavors in this promising research area breaking new grounds, we discuss some notable works in this field. Further, we present a comprehensive study on nested NER. This study shall help the readers with a quick rundown of some of the major works proposed for flat and nested NER.

Key words: Named Entity Recognition, Nested named entities, Information Extraction, Natural Language Processing, Internet of Things.

1. Introduction

Named entity recognition (NER) is a crucial task of natural language processing (NLP). NER is the course of extracting information from known semi-structured or unstructured data by identifying and tagging (categorizing) the named entities like Organization (ORG), Location (LOC), Person (PER), Timex (TIME and DATE), Monetary, Percent and Miscellaneous (MISC). In other words, it is the process where it captures sentences or paragraphs as input to the algorithm and discovers the named entities that are present in the given text and classifies them. Consider the example: From 1987-1989, Mr. Khan studied at the GITAM University located in Bangalore. In the above example, NER algorithm classifies Mr. Khan as person, GITAM University as organization and Bangalore as location. Named entities are of the two forms: flat named entities and nested named entities. Nested entities will be discussed in detail later on.

NER is one of the ongoing research area for the past 2-3 decades. There is an exponential advancement in detecting named entities but still there are huge disputes in deciding named entities because of deviation in spelling and usage of foreign words. Other challenges in NER include the usage of NEs that are too long or too short(abbreviations) especially in biomedical domain[15] and lack of available resources in specific domains and languages. Moreover, the presence of homonyms and heteronyms in some languages makes it difficult to tag these expressions due to the fact that the sense in which these expressions are used depends on the context. Consider the word ‘minute’. When it is used in the sentence, ‘It took her 50 minutes to reach the Airport.’, the entity ‘50 minutes’ is tagged with the label TIMEX. Whereas in the sentence, ‘This tea contains minute quantities of cardamom in it.’, ‘minute’ being an adjective here is a non-entity tagged with the label OTHER.

Suppose you have an e-book of a novel and would like to know the names of the characters in the novel before reading it, or the historical time period the novel is set in or the places that the plot revolves around. NER is apt for such tasks. NER has a vast range of applications in the real world. For example, NER can be used to find the names of persons, places and organizations in a particular news article or blog, to find the most related research or review paper from a particular journal, to find solutions for customer complaints and appreciably more.

Evaluation metrics are used to evaluate the excellence of any statistical machine learning model and is done by using precision, recall and F1-score.

The General equipped outlook of any named entity recognition can be envisaged using the flow-diagram shown in Figure 1. Structured or unstructured input are pre-processed to clean the data such as sentence segmentation, stemming, lemmatization, POS tagging and feature extraction are carried out. Then model is trained using training data and evaluated on testing data. Finally, the model can be used to locate the named entities.

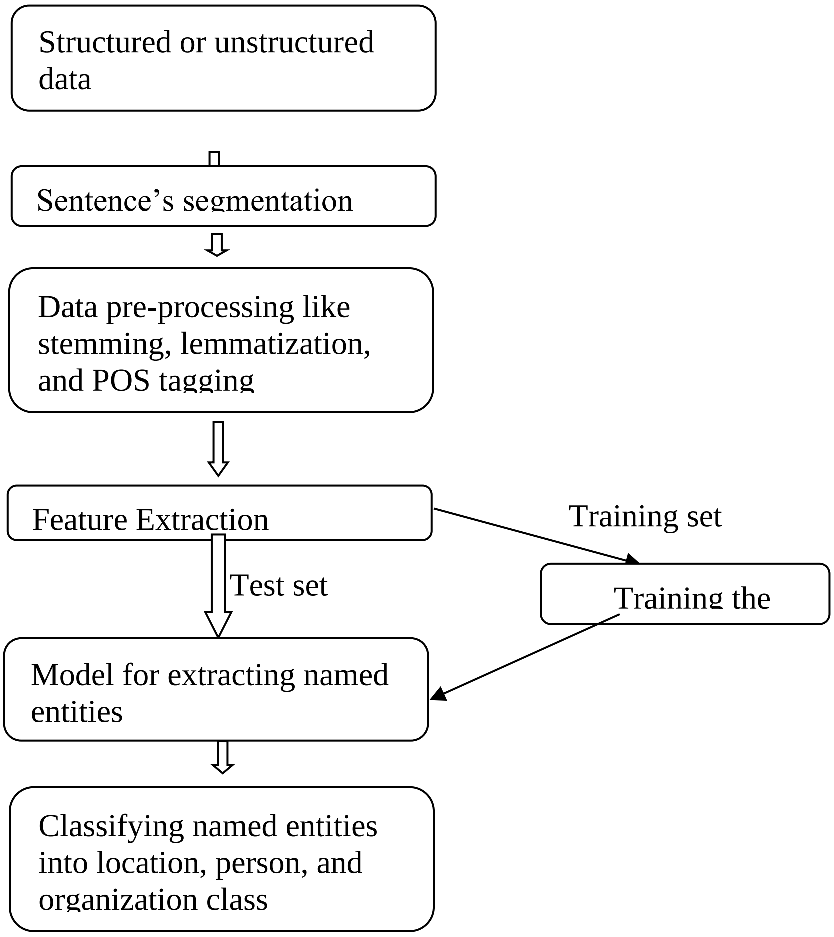


Figure 1: General outfitted Framework of NER [1]

There are many tools that are available to perform NER. Few of them are mentioned in the figure 2. One can choose an NER tool based on the requirements pertaining to the input and output formats, performance of the underlying model, domain, economic viability and the NE classes that the tool can detect.

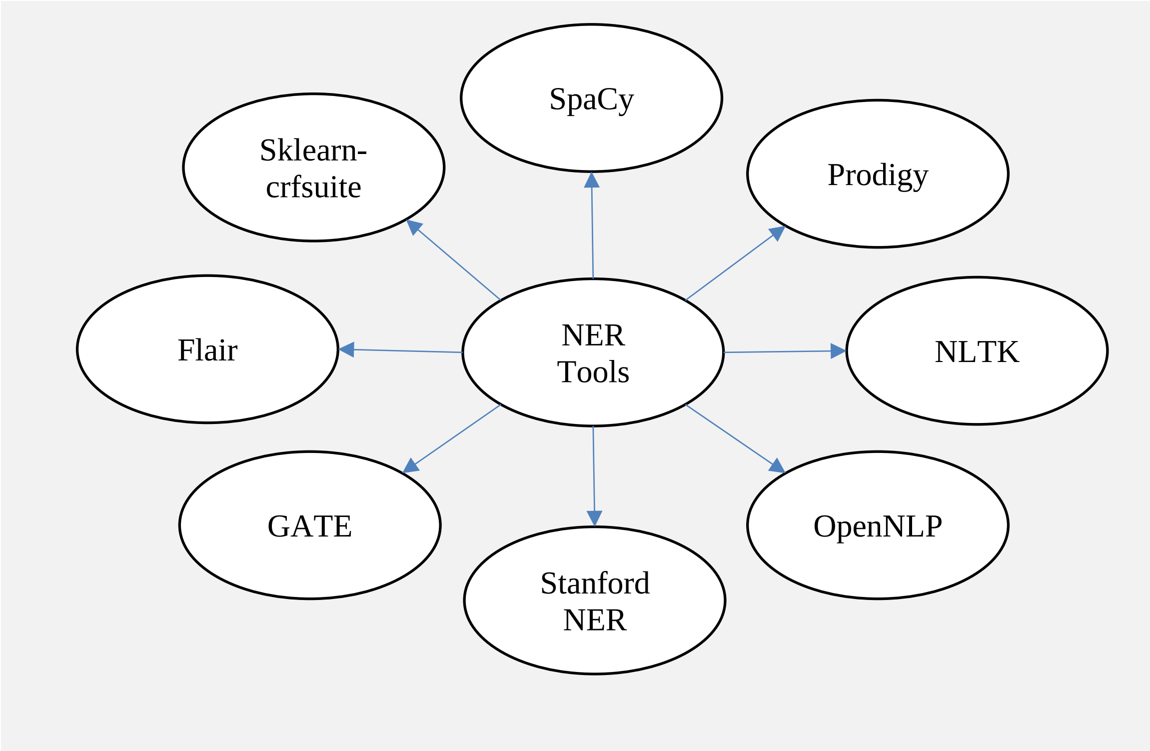


Figure 2: Few NER tools that are available online

2. Literature Review

(Borthwick et al.,2000) [2] This paper describes the system entitled MENE which is a novel statistical model for named entity recognition that exploits maximum entropy framework in its tagging decisions. When this purely statistical system is pooled with hand coded systems, it accomplishes the best score because it utilizes a flexible object based architecture and does not include any hand generated patterns as well. Here, entities are classified into 7 distinct classes i.e., location, percentage, person, monetary amount, time, date and organization. The MENE system is made up of Perl and CPP modules and have 3 crucial elements: features, histories and futures. Binary valued features are trained and proper weight will be assigned to each feature and finally, the process of decoding is carried out as follows:

1. Text is tokenized
2. Calculate the history views of each word by looking up the dictionary
3. In the given text of each article
    a. For each token, query the future objects and history view then check if feature is firing or not and merge proper weight to the feature
    b. Use the viterbi algorithm to uncover the legal path with the highest probability.

The different statistical systems like IsoQuest, Proteus and Manitoba are merged with MENE in different combinations and the joint system outperforms the individual systems. A negative aspect of the MENE system is that it outplays only if the MENE system is united with additional best statistical systems. If not, it will be unsuccessful to furnish good state of the art results.

(Lafferty et al.,2001)[3] Conditional random fields also called as CRFs, is a structure for constructing probabilistic model to fragment and identify sequence data. CRFs provide various edges over hidden markov model and stochastic grammars for such tasks, consisting capability to ease well-built self-sufficient presumption made in those models. Stochastic grammars and HMMs are probabilistic models that are apposite to problems that entails tagging an input sequence. In computer science, HMMs and stochastic grammars have been utilised for a broad range of problems in text and speech refining and information advancement. A conditional model states the chances of feasible label segments given a monitoring segment; therefore, it does not authorise modelling effort on observation, which at test time is fixed anyway. The selected characteristics may constitute features at various levels of granularity of same collective properties of monitoring segment. Non probabilistic local resolution models have also been conventional in division and labelling of words. Because of the theoretical complication of global training, these models are only instructed to reduce in the accuracy of discrete label conclusion presuming that adjacent labels are perfectly selected.

(Chieu and Ng,2003)[4] have discussed about maximum entropy approach for the named entity extraction task. This approach was not only useful for extracting local context that is present within sentences but also useful for extracting global features i.e. extracting incidence of each word within the same document, which will boost the performance of NER extraction. Only a few assumptions were made to estimate the probabilities besides the constraints that were drawn from the training data. The features that are used in maximum entropy framework are of the form binary. This approach classifies the named entities as one of the following tags: B-tag, C-tag, L-tag and U-tag for token in the beginning, token present inside the NE , token being the last word and if the token is a unique word respectively. In the process of testing, the classifier generates inadmissible classes of the type sequences. To purge such sequences we are going to calculate transition probability between word classes. Transition probability is 1 if classes are admissible. Classes are inadmissible if and only if transition probability between the classes is 0. They also introduced two systems namely ME1 and ME2. ME1 system only uses training data, it does not make use of any external knowledge but ME2 makes use of both training data and external features that are obtained from name lists. Name lists that are derived from training data are UNI (Useful unigrams), FWL (Frequent word list), UBI (Useful bigrams), NCS (Useful name class suffixes), SUF (Useful word suffixes) and FUN (Functional words). Some of the local features are rare words, case sequence and lexicon sequence. Global features include acronyms, bigrams, unigrams and many more.

(Mansouri et al.,2008)[5] performs a retrospection on the various approaches to NER namely rule-based, ML based and hybrid NER. The models reviewed in this paper are briefly mentioned in figure 3. This paper considers portability and evaluation metrics as 2 main factors to compare these approaches. It also discusses about the advantages and shortcomings of these methods. Hand-made rule based NER systems, as the name suggests, make use of rules that are based on the grammar and orthography of the language framed by language experts along with dictionaries or gazetteers. Portability is an issue in these methods. Though these models achieve convincing results over a specified domain, they may not perform well in new domains. ML based approaches considers NER as a classification problem and makes use of statistical models. These approaches are divided into 2 categories: supervised learning and unsupervised learning. In supervised learning, the model learns the right distinctions from labelled training data and uses the gained knowledge on the test data. An unsupervised model is provided with unlabelled data and it learns the similarities and differences between the examples on its own. This is not a very common approach in NER. Hybrid NER methods are the fusion of rule based and ML based methods designed with the objective of achieving an improved performance. Portability is an issue in this approach as well. Further, this paper proposes a method for NER termed as fuzzy support vector machine. The drawback of using SVM for NER is that it assigns a named entity a fixed label even in cases where the meaning of the named entity may differ based on the context and may belong to a different class. This problem is overcome by the fuzzy algorithm, which improves precision.

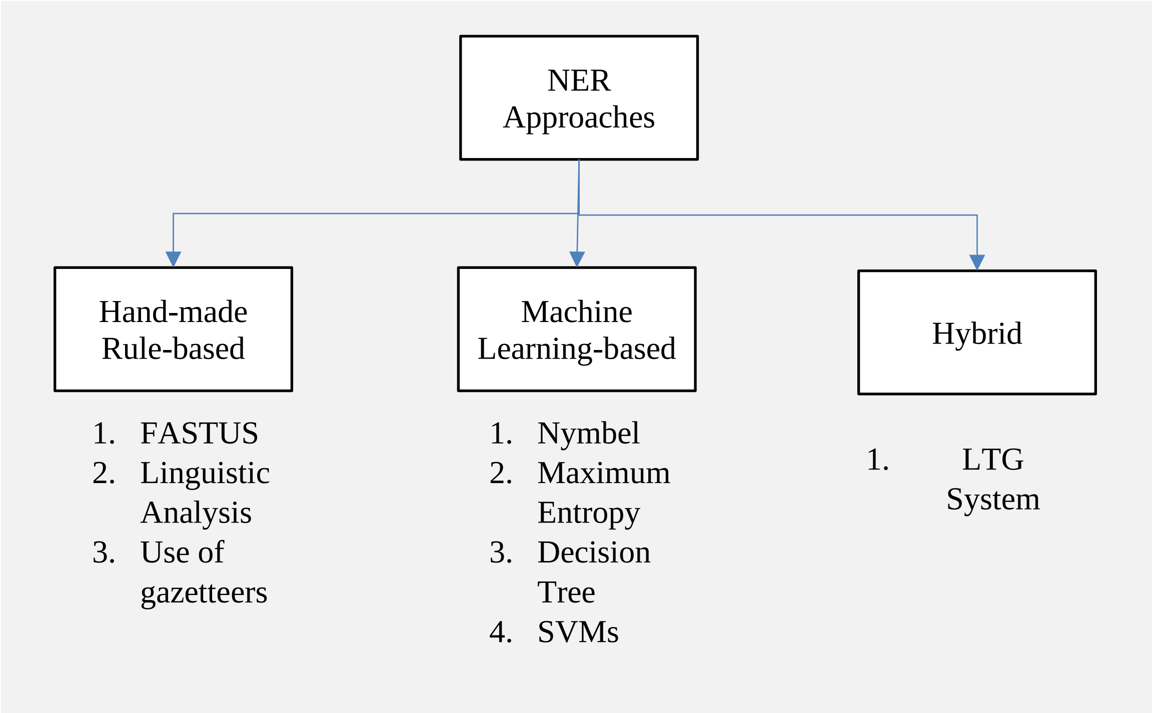


Figure 3: Classification of NER Approaches [5]

(Morwal et al., 2012) [6] proposes a scalable NER system that utilizes the HMM model. This system is very flexible that it accommodates any NE class of our interest. In other words, the predefined categories or NE classes are not fixed. Another advantage is the portability that it offers. This system can be employed for text in any language. This paper also briefly discusses about the limitations in the existing systems for NER in Indian languages and how this system overcomes those limitations. The steps involved in this system are: data preparation, training and testing. In data preparation, the words in the text are tagged based on our knowledge. Next in the training or parameter estimation phase, the states and the values of π, A, B are determined. Finally, we use the obtained values in the previous step in Viterbi algorithm and tag the entities in the testing sentence.

(Ma and Hovy,2016)[7] presents an end-to-end neural network model that utilizes BiLSTM(Bidirectional Long Short Term Memory), CNN(Convolutional Neural Network) and CRF(Conditional Random Fields). Traditional sequence models leverage handmade features based on the grammar and orthography of the language. This affects the portability of these systems. On the other hand, neural networks are not dependent on external sources of knowledge ,instead they learn to recognize patterns automatically. Firstly, character embeddings are given as input to CNNs. CNNs are employed to obtain the character-level representation of each word. Next, BiLSTM is used to obtain the context representation of each word. LSTM is a version of RNNs(Recurrent Neural Networks) which helps in recording long dependencies thereby taking the edge off gradient vanishing problem in RNNs. BiLSTM takes into account both past and future information of the target token unlike LSTM that considers only past context. Finally, a CRF is used on top of BILSTM to predict the sequence of labels for the input sentence.

(Yang et al.,2017) [8] Reranking system is a structure to enhance system execution by spudding more precise features. Neural reranking basically involves the learning of sentence level patterns in the named entity recognition to invoke the named entity mentions. When we get output from the model, for example, ‘Rama was born in Ayodhya’, Rama will be replaced by PER and Ayodhya by LOC. PER denotes the tag persons and LOC denotes locations. The output sentence will be PER was born in LOC, without the named entity mentions. There are possibilities of replacement of the tag with some other entity. Hence, it is infrequently used. The above drawback can be resolved using two dissimilar baselines, they are Discrete and neural CRFs. In discrete CRFs, the binary vector is given as input where as in neural CRFs, the word depictions are rendered as constant vectors. Experiments on both the baselines show remarkable enhancements in reranking by improving execution of named entity recognition. The drawback of using this approach is that the non-entity words are shared by all candidates, because of this it is very hard to differentiate the candidate spans in long sentences.

(Peters et al.,2018) [9] have conferred about deep contextualized word representations which models the characteristics like semantics and syntax that are complex and also how these complex characteristics show a discrepancy across linguistic contexts. They used vectors that are obtained from the bidirectional LSTM, which is trained using large text corpus. For this reason deep contextualized word representations are also known as ELMo representations. ELMo representations are contextual, deep and character based. An ELMo representation improves state of the art performance significantly and also reduces the relative error in range 6-20% in every case. This approach leverages subword units by utilizing character convolutions. Furthermore they have verified that the semantic and syntactic information in the text was efficiently captured by the biLM layers. The overall performance of the system was enhanced when all layers were utilized.

(Baevski et al.,2019) [10] have implemented a BiLSTM model which offers considerable performances across diverse language understanding problems. In this approach, firstly each word is ablated and then rest of the text will be predicted. The bidirectional model locates each token that is there in the given training data and also calculates the centre word when LR (left-to-right) and RL (right-to-left) context representations are provided. Procedure to compute the centre word is as follows:

1. Calculating both backward and forward states

2. Merging both the states as one to get the ablated word

To figure out both backward and forward representations they implemented a two tower model consisting of N stacked backward and forward tower blocks which will be operated right-to-left and left-to-right respectively. The block structure of the complete module consists of two towers that are 16 heads multi head self-attention module and FNN (Feed Forward Neural Network). They also performed consistency parsing and NER structural prediction task to improve state of the art performances.

(Luo et al.,2019) [11] this paper discussed about two deficiencies in BiLSTM model : Firstly, input will not utilize entire global information of both sentences and entire document(data set). Secondly, the nature of the constraints is sequential. To overcome the aforementioned deficiencies, they introduced hierarchical contextualized representations which contain two levels of representations: sentences and document level representations. In sentences level, they focused on contributions of different words in a single sentence and in document level, they adopted key-value pair memory network to memorize the document level representation. In the two level hierarchical contextualized representation, each input token is merged with the corresponding BiLSTM's hidden states. This approach uses IntNet-LSTM-CRF model as reference model. Hierarchical contextualized model consists of a decoder, sequence label encoder, document and sentence level encoder. The decoder uses the viterbi algorithm to formulate decisions by considering tough connections between output tags. Sequence labeling encoder is used as decoder representation generator and also memory update component. Sentences and document level encoders are used to generate sentence and document level representations. If confidence score is more, it automatically increases the sentence level representations. This model was evaluated on 3 standard NER datasets, namely, CoNLL-2003 English dataset, CoNLL-2002 Spanish dataset and OntoNotes 5.0. These datasets were divided into 3 subsets, i.e., training dataset, validation (development) dataset and testing dataset as shown in figure 4. Dividing of the dataset into 3 subsets is done due to the following reasons:

1. To know how well our built model will achieve good performance on concealed data
2. It will assist the effectual mapping between the input and the output

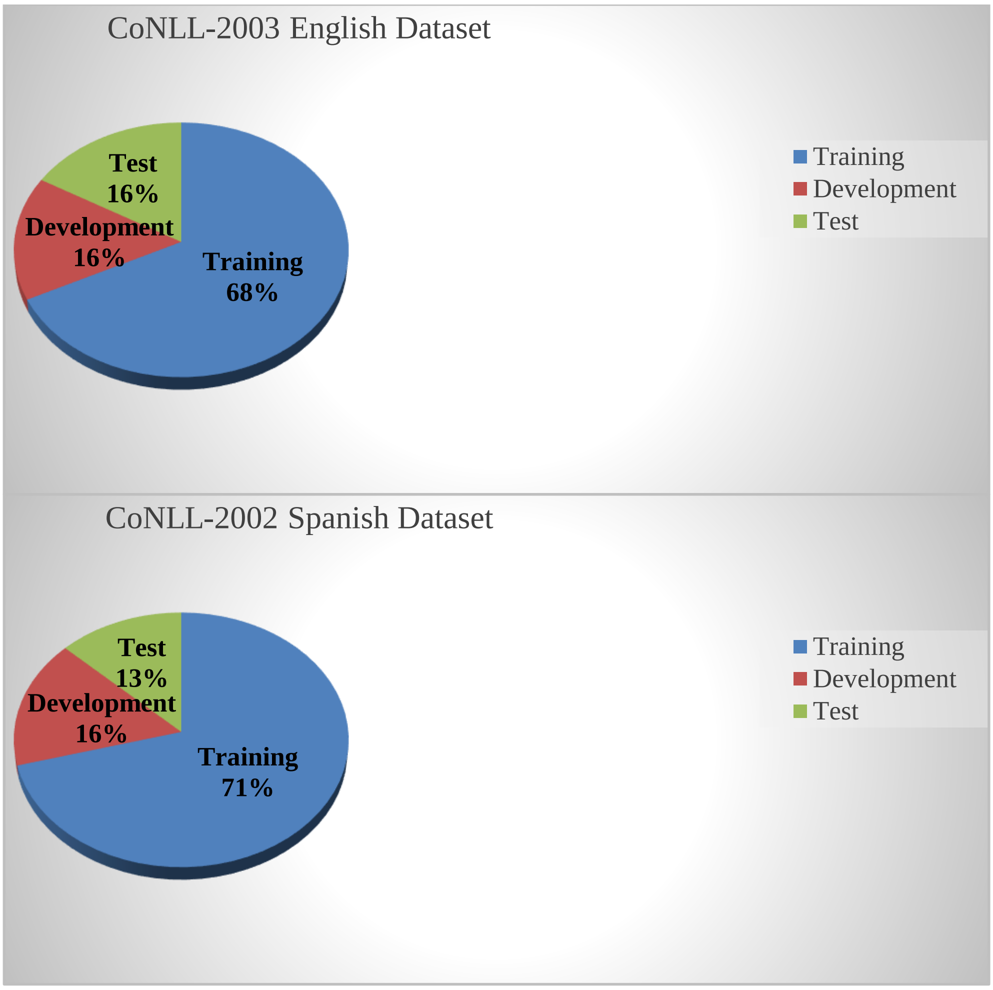


Figure 4: Split of CoNLL-2003 English Dataset and CoNLL-2002 Spanish Dataset into training, development and test sets.

(Liu et al.,2019) [12] When the concept of named entity recognition was introduced, the methods which were used in the beginning to address this problem were the hand-crafted methods. Handcrafted methods had huge drawbacks, therefore there were several experiments conducted thereafter towards improving named entity recognition. One of those methods for improving named entity recognition was by incorporating gazetteers. As we know that the English dictionary consists of English words with their synonyms, gazetteers are the collection of known instances of names of persons, locations and so on. With the help of CONLL2003 and ONTONOTES datasets there were several experiments conducted. With these datasets the test, train and development were described. Adding the gazetteer features to the neural system has shown promising enhancements in the performance. Gazetteers were used as the training data for hybrid semi Markov model also called as HSCRFs where the score level and segment level scoring information were obtained by converting traditional CRFs to semi–Markov CRFs. However, the use of gazetteers comes with certain drawbacks. Consider the example 'WHITE HOUSE' , this can represent the colour of the house or the place where the President resides. Another drawback is that gazetteers consist of rigid vocabulary of words and sub words. In recent days gazetteers are used for BERT model as the training data since this model consists of fixed vocabulary of words. BERT constructs vector of words and sub words. Adding gazetteers into these training data will enhance the model to learn better at the sub word level.

(Yamada et al.,2020)[13] Entities in Luke are represented using the entity aware self-attention mechanism. Contextualised presentation of words and entities are obtained based on the BiLSTM . In the above context of words and entities, each word is represented as an independent token and output from the BiLSTM is the contextualised depiction of them. Self-attention mechanism is a process where tokens are processed based on their attention scores and are classified. Few of the experiments conducted were entity typing, relation classification, NER and cloze style question answering. Entity typing is used for predicting the entity types in a sentence. BERT model used this process where two entities namely the target and mask were processed. In relation classification, the relation between the head and tail in a sentence were determined. Named entity recognition classifies the candidate spans into its corresponding entity type and considered all possible spans in the input sequence. In cloze style question answering, a sentence consists of missing entity and the missing entity must be found.

3. NESTED NER

A nested entity is an entity that is subsumed into another entity. They are also known as embedded entities or cascaded entities or overlapping entities. For the sake of simplicity, the early works on NER honed in on non-embedded NEs but in reality, data might contain nested entities. Consider the following example : <ORG> Geological Survey of <GPE>India</GPE></ORG> Here, ‘Geological Survey of India’ is of type Organisation and ‘India’ is of type Geo-Political Entity. The customary approach of viewing NER as a sequential analysis problem which relies on BIO type tagging is not appropriate when it comes to nested entities because a token may be a part of more than one entity and assigning single label in that case is not pertinent. While the early endeavours on nested NER mostly used rule-based approach fused with supervised ML models, the recent systems use neural networks.

(Li et al.,2020)[14] presents a unified framework that addresses both flat and nested NER. They have reckoned NER as a machine reading comprehension question answering problem. The dataset is converted into a set of triples, with each triple consisting of the query associated with the label, answer span that denotes the entity and the sequence that provides information about the context. The query corresponding to the tag and the input sequence are concatenated along with special tokens and given as input to BERT, which is the model backbone. A context representation matrix is obtained as the output from BERT. Subsequently, two binary classifiers are employed to infer if each token is the start/end index of any named entity or not which is followed by the use of an index matching model that results in the extraction of answer spans that denote entities. This framework also achieves a decent performance when it is tested on a dataset that is different from the one it is trained on. This can be owed to the generality of this approach.

Other major works on nested NER is briefly discussed in table 1 and their results are enumerated in table 2. Figure 5 illustrates the performance of these models on GENIA corpus.

Table 1 : Summary of some major works on nested NER :

Study	**Methods**	**Drawbacks**	**Enhancements(if any)**
Shen et al.,2003[15]	102 rules were included to classify the nested NEs	Substantial feature engineering	Prewritten abbreviation dictionary can be included
Alex et al., 2007[16]	3 techniques: joint labeling, layering and cascading with the latter yielding best results	In cascading, the embedded entity is not identified when the embedded entity and outer entity bear same entity type	Experimenting with other datasets that contain nested NEs
Finkel and Manning,2009[17]	Uses a discriminative parser where each sentence is delineated by a constituency tree	Due to high time complexity, it is challenging to scale to huge datasets	
Lu and Roth.,2015[18]	It presents mention hypergraphs that comprises of 5 types of nodes (A, E, T, I and X)	Spurious structures of hypergraphs	Consideration of efficient algorithms for joint mention
Muis and Lu.,2017[19]	Gaps between words are labelled utilizing mention separators	Compared to mention hypergraphs, it is 50% slower	Application of this approach to other research problems
Ju et al.,2018[20]	A dynamic model where flat NER layers are stacked resulting in the extraction of the entities in inside-out fashion	Experiences error propagation. When an outer entity is spotted first, the inner entity becomes	

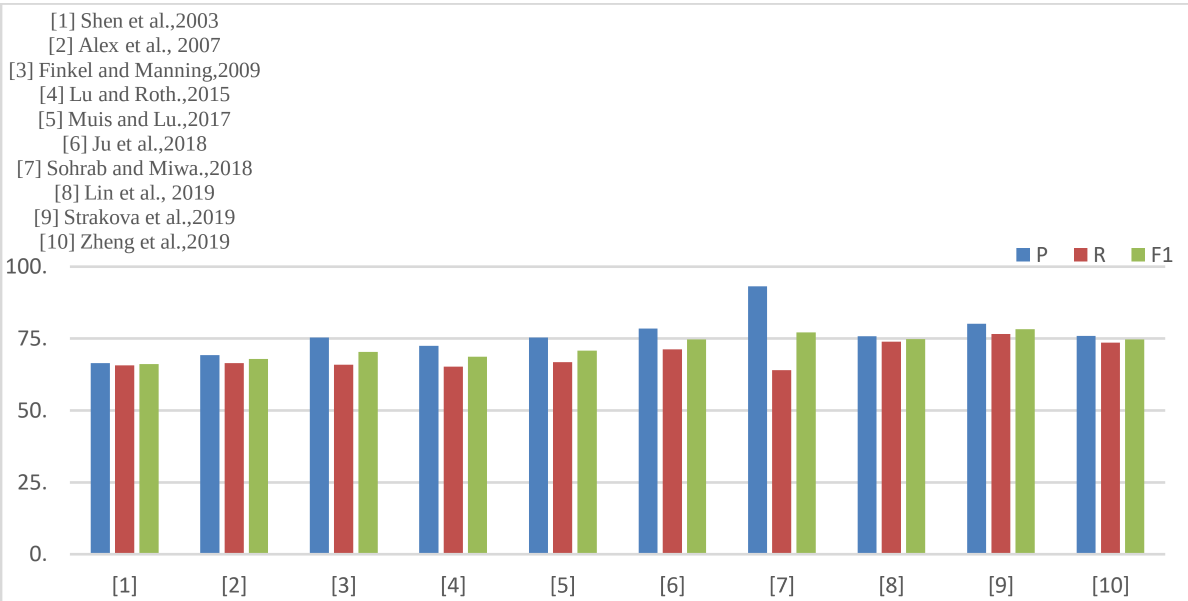
		undetectable	
Sohrab and Miwa.,2018[21]	A neural exhaustive model that considers all possible spans utilizing a BiLSTM layer	Few non-entities are extracted due to ignoring explicit boundary information	Phrase level dictionary can be utilized
Lin et al., 2019[22]	It presents ARNs (Anchor Region Networks) which make use of head words to identify the entity types	Misrecognition of entity boundaries can be caused by the presence of postpositive attributive	Syntactic knowledge could be used to eliminate few error cases
Strakova et al.,2019[23]	It proposes two approaches. The first approach utilizes multilabels and in second approach, nested NER is reckoned as seq2seq task.	Rise in NE classes in the first approach	
Zheng et al.,2019[24]	It makes use of entity boundaries to predict labels and multitask learning to improve performance	It failed to grasp the dependencies among nested entities	Enhancement of boundary detection unit

Table 2 : Results obtained from various nested NER models :

Study	**Datasets**	**P**	**R**	**F1**
Shen et al.,2003	GENIA V3.0	66.5	65.7	66.1
	GENIA V1.1	63.8	61.3	62.5
Alex et al., 2007 (cascading)	GENIA	69.3	66.5	67.9
	EPPI	73.1	68.1	70.5
Finkel and Manning,2009	GENIA V3.02	75.39	65.90	70.33
	AnCora Catalan	78.09	68.23	72.83
	AnCora Spanish	62.38	66.87	64.55
Lu and Roth.,2015	ACE 2004	70.0	56.9	62.8
	ACE 2005	66.3	59.2	62.5
	GENIA	72.5	65.2	68.7
Muis and Lu.,2017 (EDGE)	ACE 2004	72.7	58.0	64.5
	ACE 2005	69.1	58.1	63.1
	GENIA	75.4	66.8	70.8

Ju et al.,2018	GENIA	78.5	71.3	74.7
	ACE 2005	74.2	70.3	72.2
Sohrab and Miwa.,2018	GENIA	93.2	64.0	77.1
Lin et al., 2019	ACE 2005	76.2	73.6	74.9
	GENIA	75.8	73.9	74.8
	KBP 2017	77.7	71.8	74.6
Strakova et al.,2019	ACE 2005	83.48	85.21	84.33
	GENIA	80.11	76.60	78.31
Zheng et al.,2019	GENIA	75.9	73.6	74.7
	GermEval	74.5	69.1	71.7

Figure 5 : Comparison of performance of various nested NER models on GENIA dataset:



3. Conclusion

With the continual endeavours in this promising research area breaking new grounds, we have discussed some notable works on Named Entity Recognition in this paper. We have also provided a comprehensive study on nested NER. The key task of named entity recognition is to identify text and classify those entities accordingly. NER has played many key roles in the past 15 years and has been developing since then. The foremost is to enable the machines to understand what unstructured sequence of words mean. There are many applications with respect to NER. Few of them are question and answering system, machine translation system, relation extraction and so on.

4. References

1. https://blog.vsoftconsulting.com/blog/understanding-named-entity-recognition-pre-trained-models

2. Borthwick, Andrew & Sterling, John & Agichtein, Eugene & Grishman, Ralph. (2000). Exploiting Diverse Knowledge Sources via Maximum Entropy in Named Entity Recognition.
3. John Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data
4. Chieu, Hai Leong & Ng, Hwee. (2003). Named Entity Recognition with a Maximum Entropy Approach. 10.3115/1119176.1119199.
5. Mansouri, Alireza & Affendey, Lilly & Mamat, Ali. (2008). Named Entity Recognition Approaches. Int J Comp Sci Netw Sec. 8.
6. Morwal, Sudha & Jahan, Nusrat & Chopra, Deepti. (2012). Named Entity Recognition using Hidden Markov Model (HMM). International Journal on Natural Language Computing. 1. 15-23. 10.5121/ijnlc.2012.1402.
7. Xuezhe Ma and Eduard Hovy. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. arXiv preprint arXiv:1603.01354.
8. Yang, Jie & Zhang, Yue & Dong, Fei. (2017). Neural Reranking for Named Entity Recognition.
9. Peters, Matthew & Neumann, Mark & Iyyer, Mohit & Gardner, Matt & Clark, Christopher & Lee, Kenton & Zettlemoyer, Luke. (2018). Deep Contextualized Word Representations. 2227-2237. 10.18653/v1/N18-1202.
10. Baevski, Alexei & Edunov, Sergey & Liu, Yinhan & Zettlemoyer, Luke & Auli, Michael. (2019). Cloze-driven Pretraining of Self-attention Networks.
11. Luo, Ying & Xiao, Fengshun & Zhao, Hai. (2019). Hierarchical Contextualized Representation for Named Entity Recognition.
12. Liu, Tianyu & Yao, Jin-Ge & Lin, Chin-Yew. (2019). Towards Improving Neural Named Entity Recognition with Gazetteers. 5301-5307. 10.18653/v1/P19-1524.
13. Yamada, Ikuya & Asai, Akari & Shindo, Hiroyuki & Takeda, Hideaki & Matsumoto, Yuji. (2020). LUKE: Deep Contextualized Entity Representations with Entity-aware Self-attention.
14. Li, Xiaoya & Feng, Jingrong & Meng, Yuxian & Han, Qinghong & Wu, Fei & Li, Jiwei. (2020). A Unified MRC Framework for Named Entity Recognition. 5849-5859. 10.18653/v1/2020.acl-main.519.
15. Dan Shen, Jie Zhang, Guodong Zhou, Jian Su, and Chew-Lim Tan. 2003. Effective adaptation of a hidden markov model-based named entity recognizer for biomedical domain. In Proceedings of the ACL 2003 workshop on Natural language processing in biomedicine-Volume 13, pages 49–56. Association for Computational Linguistics.
16. Beatrice Alex, Barry Haddow, and Claire Grover. 2007. Recognising nested named entities in biomedical text. In Proceedings of the Workshop on BioNLP 2007: Biological, Translational, and Clinical Language Processing, pages 65–72. Association for Computational Linguistics
17. Jenny Rose Finkel and Christopher D Manning. 2009. Nested named entity recognition. In Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1, pages 141–150. Association for Computational Linguistics.
18. Wei Lu and Dan Roth. 2015. Joint mention extraction and classification with mention hypergraphs. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pages 857–867.
19. Aldrian Obaja Muis and Wei Lu. 2017. Labeling gaps between words: Recognizing overlapping mentions with mention separators. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 2608–2618, Copenhagen, Denmark. Association for Computational Linguistic
20. Meizhi Ju, Makoto Miwa, and Sophia Ananiadou. 2018. A neural layered model for nested named entity recognition. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pages 1446–1459.
21. Mohammad Golam Sohrab and Makoto Miwa. 2018. Deep exhaustive model for nested named entity recognition. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 2843–2849.
22. Lin, Hongyu & Lu, Yaojie & Han, Xianpei & Sun, Le. (2019). Sequence-to-Nuggets: Nested Entity Mention Detection via Anchor-Region Networks.
23. Strakova, Jana & Straka, Milan & Hajič, Jan. (2019). Neural Architectures for Nested NER through Linearization.
24. Zheng, Changmeng & Cai, Yi & Xu, Jingyun & Leung, Ho-fung & Xu, Guandong. (2019). A Boundary-aware Neural Model for Nested Named Entity Recognition. 357-366. 10.18653/v1/D19-1034.
25. Vanipriya, C. H., Malladi, S., & Gupta, G. (2021). "Artificial intelligence enabled plant emotion xpresser in the development hydroponics system." *Materials Today: Proceedings*.

26. P. Baglat, A. W. Salehi, A. Gupta, and G. Gupta, "Multiple Machine Learning Models for Detection of Alzheimer's Disease Using OASIS Dataset," in *International Working Conference on Transfer and Diffusion of IT*, 2020, pp. 614-622: Springer.