CS447 Literature Review: Discontinuous and Overlapping Named Entity Recognition

Saish Desai,
sbdesai2@illinois.edu

December 11, 2022

Abstract

A standard Named Entity Recognition (NER) system involves detection of entities which are present as a continuous span of words representing any real-world object such as a 'person', 'place', 'organization', etc. However, text from biomedical documents can be irregular and might contain entities with overlapping and discontinuous spans. There are instances wherein the entities share some common words within text or are embedded into each other. There are also cases where a single entity might span across certain part of text in discontinuous chunks. Specific annotation schemes, learning models and decoding algorithms are needed to encompass these cases which go beyond the standard continuous named entities. This literature review discusses the annotation schemes designed to create biomedical corpora and compares the performance of the some novel NER models trained on it for detecting nested, overlapping and discontinuous named entities.

1 Introduction

Named Entity Recognition (NER) is a sub-task of Information Extraction (IE) which identifies mentions of real-world objects from text and categorizes them into some predefined classes such as person, location, organization, etc. Figure 1 depicts entities extracted from a sample sentence using the tool $en_core_web_sm$ from the python library *spaCy*. The phrase **Tim Cook** and **Apple Inc.** are identified as entities and categorized into of type **person** and **organization** respectively. In the biomedical domain the names of the proteins, DNA formulae and diseases can be identified as entities. For example, the sentence - *This case is a paternally transmitted congenital myotonic dystrophy* contains the phrase *congenital myotonic dystrophy* which can be treated as an entity of the type **disease**. IE involves extraction of structured information from unstructured documents and NER is an important part of this pipeline which can further help in tasks such as relation extraction

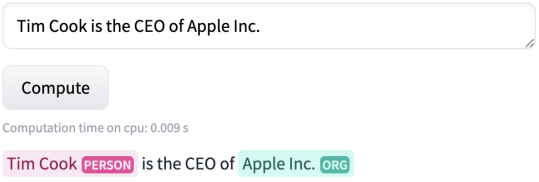

Figure 1: Application of Named Entity Recognition on a sample text

(Marsh and Perzanowski, 1998), event extraction (Hogenboom et al., 2016), question answering (Wang, 2022), etc. There has been a lot of experimentation on extracting named entities from unstructured data with models accurately detecting named entities as continuous span of words. The NER task then can be considered as a sequence labelling task which can be solved using models such as Hidden Markov Models (Rabiner and Juang, 1986) or linear-chain Conditional Random Fields (Lafferty et al., 2001). However, it is possible that entities can nest other entities or be nested by other entities. In biomedical literature, particularly in clinical text, a single entity might span across a sentence as separate discontinuous segments. Existing approaches involving use of standard BIO (*Begin*, *Inside*, *Outside*) format for encoding entities and training the data using a sequence labelling model fail to learn such entities. This further affects the performance of the decoding algorithm used for predicting the entities using the trained model in terms of time complexity and ambiguity of the output. Hence, in this literature review we will be discussing annotation schemes designed to label biomedical corpora and models proposed by (Finkel and Manning, 2009), (Lu and Roth, 2015), (Muis and Lu, 2016) and (Wang and Lu, 2019) and finally comparing their performances in detecting nested, overlapping and discontinuous entities. The review starts with presenting research questions pertaining to motivation behind the topic selection in **Section 2**. To answer this, annotation schemes proposed by all research papers have been discussed followed by their model architecture and performance used for decoding such entities. The **Section 3** gives an overview of the task of Named Entity Recognition (NER). This is followed by **Section 4** which discusses the motivation behind the selection of research papers for answering the research questions within the review. **Section 5** contains a summarized description of all the research papers involved in the review and **Section 6** involves the assessment and analysis of these research papers based on their comparison of annotation scheme, model architecture and performance. **Section 7** summarizes the contributions of each research paper in answering the research questions under study. The review concludes with **Section 8** which consolidates all the comparisons and claims made in the review, followed by the reference section comprising of all the research papers cited.

2 Research Question

In an ideal scenario a NER model should use an annotation scheme that encodes all potential combination of segments which can form an entity and learn to predict a unique output for each entity in a sentence. If a sentence contains nested, discontinuous and overlapping entities, then choosing the relevant segment combination from all possible candidates would take a lot of time even for a small textual data set. With this problem at hand the review tries to answer following research questions : 1) *What annotation schemes have been designed to prepare biomedical corpora for encoding nested, overlapping and discontinuous entities?* 2) *What technical challenges do NER models face while training on such biomedical corpora and decoding entities?* To achieve this, sentences can be parsed with the potential entity labels using syntactic constituency trees as discussed by (Finkel and Manning, 2009) or mapped into compact representations such as mention hypergraphs as presented by (Lu and Roth, 2015), (Muis and Lu, 2016) and (Wang and Lu, 2019) to encode entities within the sentence. Models such as linear-chain CRF (Lafferty et al., 2001) and semi CRF (Sarawagi and Cohen, 2004) can be trained on this annotated data and used to decode expected entities. A neural approach comprising of LSTM architecture for capturing the entity span and selecting relevant span combination as a potential entity without using any external hand-crafted features can also serve as a better candidate model for extracting discontinuous entities. The review focuses on how these models and their respective annotation schemes evolve to overcome the shortcomings of their

predecessors by reducing the time complexity and output prediction ambiguity.

3 Background

Named Entity Recognition (NER) is the process of identifying a chunk of text containing semantic information pertaining to real-world objects and classifying this chunk into a pre-defined category. This term was first coined in the 6th Message Understanding Conference (MUC 6) and it was considered as a part of the Information Extraction pipeline (Grishman and Sundheim, 1996). For example, in the sentence *Time Cook is the CEO of Apple Inc.*, *Time Cook* is an entity of type *PERSON* and *Apple Inc.* is an entity of type *ORG*. Biomedical literature and clinical trial records generally contain information in the form unstructured text. Names of genes, drugs, diseases, proteins, DNA, etc can be extracted as biomedical entities from such text. A standard named entity can be considered as a continuous sequence of words and a sequence labelling task can be used to predict the span and type of an entity within the text. A sequence labelling task involves, giving a sentence from the corpus as an input and generating the corresponding labels for each word in the sentence at the output. Sequence labelling tasks for identifying named entities takes place at span level as opposed to labelling at word level in case of Part-of-Speech tagging (Kupiec, 1992). To achieve NER using a sequencing labelling task we use the **BIO** annotation scheme (Ramshaw and Marcus, 1995). This annotation scheme involves three tags namely - **B** for beginning, **I** for inside and **O** for outside. A word is tagged with label **B** if it is the left most word of an entity. It is tagged with **I** if it is inside the span of an entity and all the words which are not part of any entity are tagged as **O**. If models like HMM (Rabiner and Juang, 1986) or CRF (Lafferty et al., 2001) are trained on such annotations, they can extract entities with continuous spans. A typical supervised machine learning approach for NER starts with an annotation scheme to label each entity from text. This labelled corpus is then used for training a NER model with additional external feature vectors. These features can be used to map a (x,y) pair where x is the input sentence and y represents expected named-entities. These features can be categorized into three main types - Word-level features, List lookup features and Document and corpus features as discussed by (Nadeau and Sekine, 2007). Word-level features capture information pertaining to the word case, punctuation, tense, prefix, suffix, etc. List lookup features check if words from the training data are a part of any curated list such as a ”lexicon” or ”gazetteer”. Finally, Document and corpus features contain the statistical information pertaining to document or corpus as a whole. Each feature has a weight associated with it. The feature weights are learned while training the model on the annotated corpora. The learned model is then used to predict the potential entities from a test sentence using a decoding algorithm. With progress in neural networks, entity recognition can be modelled using a neural approach, involving learning word and span level encoding in absence of external features and classifying these encodings into expected entities. However, entities especially from the biomedical domain can have a complicated structure. As discussed in **Section 1** entities can be overlapping, nested or discontinuous. This reviews talks about approaches which go beyond the standard sequence labelling tasks to capture such instances of named-entities.

4 Motivation

The research papers chosen for the review try to build on the standard named entity extraction approach for recognizing nested, overlapping and discontinuous entities. As an example we will consider the sentence mentioned in Figure 2. Let us consider each labelled span of text as a mention.

The first mention ***hiatal hernia*** is a standard continuous entity spanning over 2 words. The second is a combination two mentions ***lacrerations*** and ***esophagus*** which are two discontinuous segments of the same entity, thus portraying an example of a discontinuous entity. The third and fourth mentions, ***blood in stomach*** and ***lac*** can be considered as two overlapping entities wherein the fourth mention is additionally a discontinuous entity. Finkel and Manning (2009) proposed a model to identify a case of named entity recognition were an entity might be nested in another entity or might embed an entity of smaller span within itself, thus contributing towards extraction of nested named entity. However, it suffers from time complexity issues. Lu and Roth (2015) has tried to use a compact representation for annotating the named entities using mention Hypergraphs and has been successful in achieving a linear time complexity in count of words in the input sentence. Muis and Lu (2016) has inherited the Hypergraph annotation from Lu and Roth (2015) with an additional encoding for words outside the entity spans for ensuring the annotation of discontinuous entities. The models discussed so far are machine learning based models which use additional external features in the training phase. These models face the problem of ambiguity while decoding the correct output from encoded Hypergraphs. To solve this issue, (Wang and Lu, 2019) has come up with a neural approach in two stages for extracting overlapping and discontinuous entities. It starts extraction of all continuous segments to identify continuous and overlapping entities, followed by combining segments of discontinuous entities. In all the papers discussed in the review the performance of the proposed model has been compared with the existing models used for standard named entity recognition.

5 Summary of Discussed Papers

5.1 Nested Named Entity Recognition

5.1.1 Task

Finkel and Manning (2009) proposed a novel model for detecting nested named entities by annotating the sentences containing entities with a nested structure. The nested structure allows the detection of a particular entity with help of entity labels contained in them and the labels which contain such entities. Each sentence is represented as a parse tree. The words in the sentence are the leaves of this tree and and there are phrases within the tree structure which are annotated as entities. These parsed trees are trained on a discriminative constituency parser influenced by (Finkel et al., 2008). The performance of the proposed model is evaluated in terms of F-score and compared with a standard semi-CRF approach (Sarawagi and Cohen, 2004).

EGD showed [***hiatal hernia***]$_1$ and vertical [***laceration***]$_2$ in distal [***esophagus***]$_2$ with [***blood in*** [***stomach***]$_4$]$_3$ and overlying [***lac***]$_4$.

Figure 2: Discontinuous and Overlapping entities encoded in a sentence

5.1.2 Data

The model uses GENIA v.3.02 corpus (Ohta et al., 2002). This corpus consist of 2000 Medline abstracts containing approximately 500k words and is annotated with 36 different kinds of biological entities, and with parts of speech. The data has been split into three sets-namely train, development and test. 90% of the data is used for training and remaining 10% is used for testing. In the development phase, the first half of the data set has been used for training and the next quarter has been used for testing the model. All the entities to be detected can be referred to as biomedical entities. For the proposed model the researchers have collapsed all the entity types from the corpus into 5 major types - Protein, DNA, RNA, Cell line and Cell type.

5.1.3 Annotation

The annotation for the nested NER model is in the form of a parse tree representing each sentence. Each parsed tree consist of a Root node which is connected to the entire sentence and is further divided into phrases. These phrases are depicted as the entities which are the intermediate nodes in the tree and the words are represented as the leaves of the tree. The parsing represents a nested structure of entities. Entity at a particular node might branch further into other nodes which might be individually represented as another entities. Figure 3 shows an example of the parsed tree. The phrase *mouse GM-CSF promoter* is labelled as an entity of type *DNA*, but it also contains the word *GM-CSF* which is labelled as an entity of type *PROT* (label for Protein). Each node is also labelled with the names of its parent and grandparent node labels in order to capture the nested structure. Each word has a non-terminal connected to it which serves as the part of speech tag of that word. This tag enables detection of words within an entity. For example, if a word is labelled with *"verb"* POS tag then it has lesser chances of being associated with any entity tag. The parsing has been influenced from the Probabilistic Context Free Grammar (PCFG) (Manning and Schütze, 1999) parsing, however the tree structure is comparatively flat.

5.1.4 Model

The proposed nested NER model has used a CRF-CFG (Finkel et al., 2008) parser to predict the most likely parsed tree. This model follows the approach of PCFG with a variation of calculating clique potential over sub-trees and combining the results instead of calculating the probabilities over rules for a CFG parser. Each one-level sub-tree within the parse tree is predicted for a given sentence using a defined local clique potential. This sub-tree is basically a rule from the PCFG parsing and a

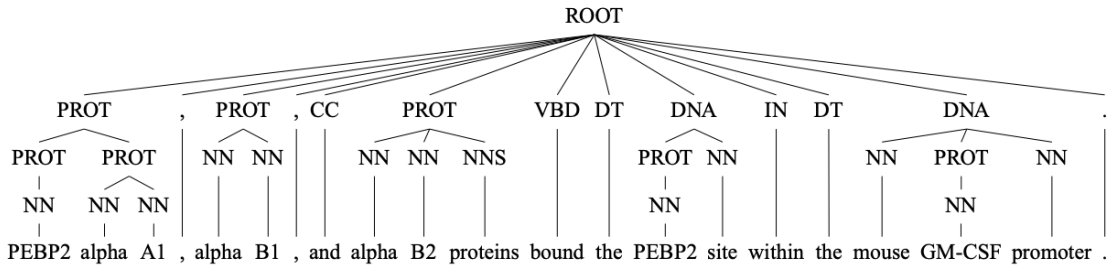

Figure 3: Parsed Tree with nested named entities

set of features with their corresponding weights are trained to generate best possible rules/sub-trees to get the expected parsing for a given sentence. The optimization of the features used for training the parser is achieved by using a stochastic gradient descent technique.

5.1.5 Performance

The performance of the proposed model is compared with that of a semi-CRF model. The proposed model outperforms the semi-CRF model when evaluated on the top-level entities as well on all-entities. The disadvantage of using this method is its runtime. The number of words tagged per second by the proposed model is low as compared to the semi-CRF and linear chain CRF models.

5.2 Joint Mention Extraction and Classification with Mention Hypergraphs

5.2.1 Task

Lu and Roth (2015) introduced an approach using Mention Hypergraphs and Linear-Chain CRF model (Lafferty et al., 2001) to detect nested and overlapping named entities as an addition to traditional NER approach. The model proposed in the paper predicts named entities with linear time complexity with respect to the the number of words in a sentence and the types of entities. Named entity recognition comes with three major challenges. Each entity can be a group of words in a sentence and two entities might overlap each other or might be contained by each other. The annotation schemes and entity extraction algorithm have tried to solve these challenges by making the prediction of entity types efficient and scalable for large data sets especially in the biomedical domain.

5.2.2 Data

The authors have tested their model on multiple data sets. However, for the purpose of this review we will be focusing on experiment of the proposed model on the GENIA v.3.02 corpus (Ohta et al., 2002). In this experiment the model performance has been compared with that of (Finkel and Manning, 2009). The authors have inherited the data division scheme from (Finkel and Manning, 2009) with 90% of the data from the corpus dedicated for training and remaining 10% for testing . From the train data 90% has been used to train the model and the remaining 10% has been used as a development set.

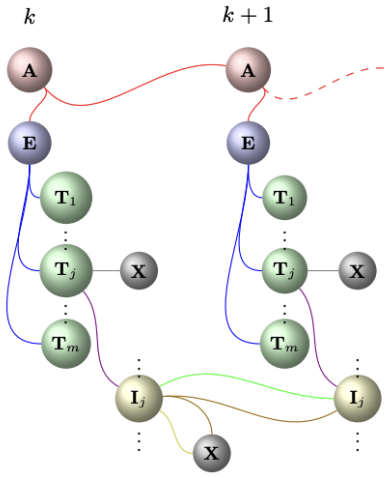

Figure 4: Compact representation of entities using Mention Hypergraphs

5.2.3 Annotation

Lu and Roth (2015) observed that in a sentences of n words, an entity of type t will have a some span $\langle\ b_m,\ e_m,\ t\ \rangle$, where b_m serves as the start index, e_m serves as the end index and t serves as the entity type. There will be $tn(n+1)/2$ entity candidates and $2^{nt(n+1)/2}$ entity combinations, which makes enumerating all the combinations exhaustive even for small values of t and n. Hence, Lu and Roth (2015) have come up with a compact representation of all entities using mention hypergraphs. This is considered as a network of nodes and hyperedges. Each hyperedge has been used to connect a parent node to an ordered list of children nodes. Figure 4 depicts a part of the mention hypergraph proposed by (Lu and Roth, 2015) representing a span starting with the k-th word in a sample sentence. Considering the k-th position, the node $\mathbf{A}_k$ is used to represent a mention which has a left boundary either at k or at $k+1$. The node $\mathbf{E}_k$ is used to represent a mention with left boundary exactly at k. The node $\mathbf{T}_j^k$ is used to represent a mention of type j with current word at position k. Similarly, the node $\mathbf{I}_j^k$ has been used to represent an intermediate part of the mention of type t with current word at position k. The node $\mathbf{X}$ is a called the terminal node which depicts the end of the mention. Hyperedges are used to relay the semantic information from the parent to the child nodes. Mention information from the parent node $\mathbf{A}_k$ will be transferred to node $\mathbf{E}_k$ if the mention starts at k, otherwise it will be relayed to the node $\mathbf{A}_{k+1}$. Similarly tag $\mathbf{I}_j$ connects to $\mathbf{I}_{j+1}$ if the entity span continues and to $\mathbf{X}$ if the span ends. It may also connect to both nodes in case of overlapping entity combinations. With this annotation Lu and Roth (2015) proposes a theorem that *Any combination of mentions in a sentence can be represented with exactly one sub-hypergraph of the complete mention hypergraph.* Thus the annotation can be compactly used to represent nested and overlapping entities.

5.2.4 Model

A log-linear approach has been used for training the model. This is a discriminative model, which is used to predict the best possible output (all possible sub-hypergraphs) for entities within a sentence

given as input using the following equation:

$$p(y|x) = \frac{exp(w^T f(x,y))}{\sum_{y'} exp(w^T f(x,y'))}$$

This conditional probability is described in terms of a feature set **f(x, y)** mapping input-output pairs within the training data. Each feature is associated with a weight **w**. The aim of the training phase is to update the weights such that the model predicts the best possible mention combination as entities for a given test sentence. To achieve this, the training phase constructs an objective to minimize the regularized negative joint log-likelihood of the dataset. The optimization is achieved using gradient descent-based methods such as L-BFGS (Liu and Nocedal, 1989) and the authors of this research have been able to reach a global optimum due to the convex nature of the objective function. Dynamic programming algorithms have been implemented for decoding the entities and their types in terms of the mention sub-hypergraphs using the trained model.

5.2.5 Performance

Performance of the proposed model when trained on the GENIA v.3.02 corpus is compared with that of semi-CRF model (Sarawagi and Cohen, 2004) and a constituency parser proposed by (Finkel and Manning, 2009). F1-score has been used as the metric for comparison. It has been observed that the proposed model outperforms the semi-CRF model but has a lower F1 score as compared to (Finkel and Manning, 2009). However, the model of (Finkel and Manning, 2009) has a cubic time complexity. Thus, the proposed model, which has linear time complexity in the number of words in the sentence is highly scalable for large data sets. Moreover, when trained on the ACE2004 dataset, the authors have observed the model to perform efficiently while dealing with large number of entity types without much increase in the execution time. This is evident from the fact that model has a linear time complexity with respect to the number of entity types. Thus, the use of mention hypergraphs have enabled a compact representation of entities, enabling a model trained on such annotation scheme to predict the nested and overlapping entities efficiently with lower execution time.

5.3 Learning to Recognize Discontiguous Entities

5.3.1 Task

Muis and Lu (2016) built their research on works by (Finkel and Manning, 2009) and (Lu and Roth, 2015) by proposing a model to identify discontinuous entities in textual data in addition to overlapping and nested entities. Entities extracted using traditional NER systems are usually considered as continuous spans of words. Such systems consider two assumptions while extracting entities: 1) Entities cannot overlap each other or may not be nested in each other. 2) A single entity cannot span across the text in discontinuous segments. The models proposed by (Finkel and Manning, 2009) and (Lu and Roth, 2015) can handle cases which violate the first assumption, but cannot handle those that go beyond the second assumption. Such violations have been observed in disorder mentions within clinical text. The proposed model has presented a compact annotation of entities in the form of mention hypergraphs followed from (Lu and Roth, 2015). However, an addition has been made to the annotation structure to accommodate encoding of discontinuous entities within the hypergraph. The proposed model has introduced the idea of model ambiguity with which it is able to quantify the difficulty level of identifying the precise entity output from the encoded hypergraphs.

5.3.2 Data

The data set used for the proposed model contains entities pertaining to disorders mentioned as discontinuous or overlapping segments across clinical text. The data has been a part of the task organized by ShARe/CLEF eHealth Evaluation Lab (SHEL) 2013 (Suominen et al., 2013) and SemEval-2014 (Pradhan et al., 2014), involving extraction of disorder mentions from clinical text. However, the dataset contains a small proportion of discontinuous entities. To understand the behaviour of the model towards discontinuous entities a subset of the data is used for training, such that each sentence in the dataset contains at least one discontinuous entity. The dataset is further split into train, test and validation set following the data split procedure used in SemEval-2014. This data set is labelled as the Large Data set. However, to understand the impact of the size of data on the model performance, the researchers have constructed an additional data set, by generating the train and development dataset from the train data of the Large data set and then using the development data of the Large data set as the test set. The initial research focuses on extraction of one type of entity using the given dataset and later expands to multiple entity types.

Infarctions$_{[\mathbf{BH}]}$ either ***water***$_{[\mathbf{BD}]}$ ***shed***$_{[\mathbf{ID}]}$ or ***embolic***$_{[\mathbf{BD}]}$

Figure 5: Overlapping and Discontinuous Entity example

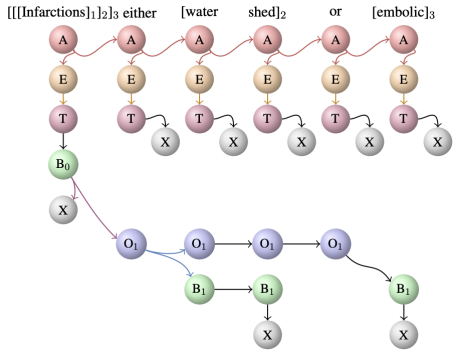

Figure 6: Mention hypergraph representation for discontinuous and overlapping entities

5.3.3 Annotation

Muis and Lu (2016) first talks about the annotation scheme used by (Tang et al.), which is an extension of the standard **BIO** encoding. There are 7 tags namely **B**, **I**, **O**, **BH**, **IH**, **BD** and **ID**. The annotation starts identifying continuous word sequences shared by multiple entities and labelling them using the head tags **BH** and **IH**. This is followed by labelling the remaining discontinuous components using the **BD** and **ID** tags. Other continuous sequences which can from an entity on their own are labelled using the **B**, **I** and **O** tags. This baseline approach has been considered as inefficient, since it does does not encode all the potential entities within a sentence. In cases where two different entities share a common sequence of words, the model fails to decode both entities. Figure 5 shows examples of entities which can be discontinuous and overlapping at the same time. Ideally, ***Infarctions***, ***Infarctions...water shed***, ***Infarctions...embolic*** should be the three entities

identified by any NER system. However, the baseline approach fails to identify ***Infarctions*** as a separate entity since it is labelled with a **BH** tag which makes it difficult for the model to identify it as a standalone entity. To overcome this drawback and compactly represent discontinuous and overlapping entities from all possible entity combinations (Muis and Lu, 2016) has adopted the mention hypergraph architecture. The proposed mention hypergraph representation is similar to the one introduced in (Lu and Roth, 2015). Nodes and Hyperegdes are used to represent different combinations of entities within a sentence. In addition to the nodes used in (Lu and Roth, 2015), nodes for accommodating the discontinuous entities are added. Node **B** is used to represent the words which are part of the entity and Node **O** is used for representing words outside the entity span. Considering a sequence of words in a sentence node $\mathbf{B}^k_{t,i}$ represents that the k-th word in the sentence is the i-th component of the entity of type t. $\mathbf{O}^k_{t,i}$ represents that the k-th word is the word between *(i-1)*-th and i-th component of the entity of type t. Hyperedges are used to connect a parent node to its respective child nodes. The connection depends upon whether the current word is a part of an entity or outside it. Figure 6 shows the hypergraph representation for sentence in Figure 5. The first entity ***Infarctions*** is a standard continuous entity labelled using node B_0. The second entity and third entities ***Infarctions...water shed*** and ***Infarctions...embolic*** share a common word and also span across two discontinuous segments. These overlapping and discontinuous entities are represented using a combination of **B** and **O** nodes from the first to the last word of the respective entities. Each entity is thus encoded as a part of the mention hypergraph. Thus, a compact representation using the hypergraphs identifies all possible entity candidates which (Tang et al.) fails to detect. This is the first variant of the model termed as the SHARED Model. However, there is an ambiguity which arises in cases where there is an overlap between a continuous and discontinuous entity. A variant of this model called the SPLIT model has been introduced wherein the nodes of the hypergraph are further split based on the number of components present in the entity to which they belong. For example, for the k-th word the node $\mathbf{B}^k_{t,i}$ is now represented as $\mathbf{B}^k_{t,i,j}$. This new nomenclature of the node represents the k-th word as the i-th component of an entity of type t which contains j components in total. Thus, a separate encoding for different sizes of entities can be generated.

5.3.4 Model

The model training has used a log-linear approach to maximize the log likelihood of all the training instances containing the input and output. The aim is to train the model to maximize the probability of the encoded hypergraphs representing entities for a given input text. Here, the input is a the sentence containing entities and the output the corresponds to entity-encoded hypergraph from which the required entities can be decoded. With the presence of overlapping entities along with discontinuous entities ambiguity issues arise while decoding the entity encoded hypergraph. This ambiguity is the result of mapping a single hypergraph encoding to multiple entities. (Muis and Lu, 2016) have presented a proof to quantify the ambiguity for their proposed model and compare it with that of the baseline model (Tang et al.). To quantify the model ambiguity a term called canonical encoding has been used. It represents the correct or the preferred encoding for entities in a sentence. A model predicting large number of canonical encodings for a given text is expected to have lesser ambiguity as compared to the one with lesser encodings. Using this argument (Muis and Lu, 2016) have proved that the proposed model has lesser ambiguity as compared to the baseline model trained using linear-chain CRF. The baseline and proposed models both are trained using external features pertaining to syntactic and semantic information of the text. Some of the features are - neighboring words and their positional information, Part-of-Speech tags of words and their

neighbors, orthographic information (prefix, suffix, capitalization, lemma), etc. The weights associated with these features are updated in the training phrase and used for predicting potential entities in the form of hypergraphs.

5.3.5 Performance

The authors have compared the performance of the baseline model using the two variants (SHARED and SPLIT) of the proposed model. Each of these are trained on the LARGE and SMALL data set. Each dataset has two variants, one with a small proportion of discontinuous entities and other adjusted with equal proportion between continuous and discontinuous entities. The model performance is evaluated in terms of precision, recall and F1 score. The proposed model is able to separately extract continuous and discontinuous entities which overlap, whereas the baseline model fails to extract such entity combinations. Precision can be defined as the ratio of count of correct predictions made by the model and count of all predictions made by the model. The proposed model has a relatively higher precision value, since most predictions made by the model are accurate due to their compact annotation scheme.

5.4 Combining Spans into Entities: A Neural Two-Stage Approach for Recognizing Discontiguous Entities

5.4.1 Task

Models which can detect discontinuous and overlapping entities have been proposed by (Muis and Lu, 2016) and (Tang et al., 2013). However, these models face the problem of ambiguity while decoding entities from their respective encoded structure. To solve this issue (Wang and Lu, 2019) has proposed a neural network based-model using the Long Short Term Memory (LSTM) architecture. This model follows a two-stage approach. The first stage called segment extraction extracts all the continuous and overlapping spans of text which can either be considered as entities on their own or might a part of a larger discontinuous entity. The second stage called segment merging involves learning to combine a set of spans together and using a classifier to identify the correct combination candidate for the resulting discontinuous entity. For example, Wang and Lu (2019) talks about a phrase "*He had blood in his mouth and on his tongue.*" This sentence has two entities each depicting a disorder namely - *blood in his mouth* and *blood...on his tongue*. The two-stage approach will first extract all the continuous and overlapping segments from the text. Thus, the first stage will extract *blood*, *blood in his mouth* and *on his tongue* as the three segments. The second stage will then generate all possible combinations of these segments and use a binary classifier to predict the combinations which form the resulting discontinuous entities, thus leading our expected disorder mentions.

5.4.2 Data

The proposed model has been trained and evaluated on the data from ShARe/CLEF eHealth Evaluation Lab (SHEL) 2013 (Suominen et al., 2013) and SemEval-2014 (Pradhan et al., 2014). The data obtained from these two sources comprise of clinical textual documents containing disorder mentions. Due to unstructured documentation, these mentions have been spread across sentences as overlapping and discontinuous entities which the model has tried to detect. A subset of data has been selected from these sources to ensure that a sufficient proportion of the text contains discontin-

uous and overlapping entities. An assumption has been made to ensure that segments of the same type combine when forming a discontinuous entity.

5.4.3 Annotation

The entities have been annotated using segmental hypergraphs, and the proposed model has directly adapted this annotation approach from (Wang and Lu, 2018). This annotation scheme is similar to (Lu and Roth, 2015) as it encodes continuous and overlapping spans of words from the input text. Each segment has been uniquely represented using nodes and hyperedges. A segment combination has been further represented using a hyperpath, which is a combination of hyperedges.

5.4.4 Model

The proposed model has used a two-stage approach, starting with extraction of all the continuous segments of word. Two LSTM (Graves and Schmidhuber, 2005) based encoders have been used to represent word-level and segment-level encoding. The first encoder converts input words into word embedding and the second encoder generates encoding for a span of words. Each span of words is termed as a segment. For the first stage called segment extraction, sentence $\mathbf{x}$ is given as an input and a set $\mathbf{s}$ of continuous segments is generated as the output. Each segment is represented as an hyperpath which is a group of hyperedges within a hypergraph. The second stage called segment merging involves merging relevant segments into entities. An additional LSTM encoder has been used to learn the interaction between segments from the first stage. All the segments identified in the first stage are used as features for a binary classifier to identify whether a segment combination forms a potential discontinuous entity.

5.4.5 Performance

The performance of the proposed model is compared with models trained using the annotation schemes from (Tang et al.) and (Muis and Lu, 2016) which are treated as the two baseline models. The performance of these models are compared using F1-score and it is observed that the proposed model outperforms the baseline models. Two-stage approach of the proposed model overcomes the problem of ambiguity while extracting discontinuous entities which (Muis and Lu, 2016) fail to solve. Adopting a neural network-based approach ensures that the training process does not require other external features followed by the previously discussed models. Thus, features carrying semantic and syntactic information need not be handcrafted and used in the modelling process.

6 Assessment

Named Entity Recognition (NER) started as a sequence labelling task. However, irregularities in textual data led to implementation of models going beyond the traditional NER approach for encompassing nested, overlapping and discontinuous entities. There has been a slow progress in development of models used for detecting such entity types due to technological limitations. With different structures and types of entities coming up due to increasing content generation a given sentence might contain many potential combination of spans out of which only a handful of combinations might be treated as relevant entities. The research papers discussed in the literature review have tried to come up with annotation schemes to select and present relevant entity mentions in textual data. Since, such irregularities are more pronounced in biomedical data the proposed models

have evaluated entities when trained on biomedical data such as clinical reports, literature and observations written by doctors. One major drawback which all the proposed models face is that they are supervised learning algorithms and require labelled data for training. A well-defined annotation scheme is needed to identify each of entity and the annotator needs to be consistent throughout the annotation process. The researchers for each model have either designed their own annotation schemes or referred to existing schemes. In addition to these schemes the machine learning based models discussed in (Finkel and Manning, 2009), (Lu and Roth, 2015) and (Muis and Lu, 2016) require external features for training the model. These features need to be tested empirically by observing the change in the model performance. There is no fixed rule for selecting the features. The deep learning approach followed (Wang and Lu, 2019) has a better performance. However, the model is like a black box. All the results are based on how well the model is trained. But, it is very hard to decipher the reason for a specific behaviour of the model. All these model have been mostly trained on data from the biomedical domain. Thus, these can considered as potential tools for BioNER.

7 Research Paper contribution

All the research papers discussed in the literature review have designed annotation schemes and models for predicting entities by going beyond the traditional NER approach. Each research paper tries to improve the model to identify the specific cases of named-entities based on the drawbacks observed in the previous models. With textual data generated in large volumes from different sources, new combinations of entities have come up making it difficult for extracting entities using the traditional sequence labelling approach. The authors publishing their research through biomedical literature tend to be very particular in following a specific nomenclature for mentioning entities involved in their work. However, text data from clinical reports might not always contain well-documented data as these are just noted as observations by the doctors or researchers as part of patient diagnosis or any medical experiment. Here the writers of such reports take the liberty of writing the medical observations are per their convenience and understanding. For example, the phrase "*productive cough with white or bloody sputum* has two discontinuous mentions *cough..white..sputum* and *cough..bloody..sputum*, which overlap with each other by sharing the word *cough*. In such cases it is necessary to look beyond an entity as continuous sequence of words and identify relevant combination of segments which can form the required entities hidden in the text. Finkel and Manning (2009) has tried to come up with an approach to deal with nested entities by developing a labelled corpus using a parse tree. With overlapping and discontinuous entities mentioned in clinical text, the entity extraction process becomes time-consuming due to extraction of relevant entities from all possible entity combinations. The compact hypergraph representation used for annotation in (Lu and Roth, 2015) and (Muis and Lu, 2016) have made it possible for the corresponding proposed models to encode the relevant span combinations as entities with linear time complexity. Training process for the proposed models involves coming up with hand-crafted external features which map the input text to the encoded output structures. This is an empirical process requiring a lot experimentation to decide the relevant features which can give us the expected results. Wang and Lu (2019) has used a neural network-based approach which bypasses the feature selection process and uses the relation between spans of words within text as features. Thus, no additional external indicators providing information about the syntax and semantics of the text are needed in the training and decoding process.

8 Conclusion

In this review we discussed cases where standard NER systems fail to use their sequence tagging algorithms in detecting entities which are nested, overlapping and discontinuous. We discussed the annotation schemes introduced in each research paper and described the additions made to each annotation scheme for improving the model performance in capturing cases of named entities which the preceding models fail to introduce. We briefly discussed the training and inference algorithms used by each proposed model and described the performance of these models using evaluation metrics mentioned in each research paper.

References

Jenny Finkel, Alex Kleeman, and Christopher Manning. 2008. Efficient, feature-based, conditional random field parsing. pages 959–967.

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*, pages 141–150, Singapore. Association for Computational Linguistics.

Alex Graves and Jürgen Schmidhuber. 2005. Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural Networks*, 18(5):602–610. IJCNN 2005.

Ralph Grishman and Beth Sundheim. 1996. Message Understanding Conference- 6: A brief history. In *COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics*.

Frederik Hogenboom, Flavius Frasincar, Uzay Kaymak, Franciska de Jong, and Emiel Caron. 2016. A survey of event extraction methods from text for decision support systems. *Decision Support Systems*, 85:12–22.

Julian Kupiec. 1992. Robust part-of-speech tagging using a hidden markov model. *Computer Speech Language*, 6(3):225–242.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, page 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Dong C. Liu and Jorge Nocedal. 1989. On the limited memory bfgs method for large scale optimization. *Mathematical Programming*, 45(1-3):503–528.

Wei Lu and Dan Roth. 2015. Joint mention extraction and classification with mention hypergraphs. pages 857–867.

Christopher D. Manning and Hinrich Schütze. 1999. *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, Massachusetts.

Elaine Marsh and Dennis Perzanowski. 1998. MUC-7 evaluation of IE technology: Overview of results. In *Seventh Message Understanding Conference (MUC-7): Proceedings of a Conference Held in Fairfax, Virginia, April 29 - May 1, 1998*.

Aldrian Obaja Muis and Wei Lu. 2016. Learning to recognize discontiguous entities. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 75–84, Austin, Texas. Association for Computational Linguistics.

David Nadeau and Satoshi Sekine. 2007. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30.

Tomoko Ohta, Yuka Tateisi, and Jin-Dong Kim. 2002. The genia corpus: An annotated research abstract corpus in molecular biology domain. In *Proceedings of the Second International Conference on Human Language Technology Research*, HLT '02, page 82–86, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Sameer Pradhan, Noémie Elhadad, Wendy Chapman, Suresh Manandhar, and Guergana Savova. 2014. SemEval-2014 task 7: Analysis of clinical text. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 54–62, Dublin, Ireland. Association for Computational Linguistics.

L. Rabiner and B. Juang. 1986. An introduction to hidden markov models. *IEEE ASSP Magazine*, 3(1):4–16.

Lance A. Ramshaw and Mitchell P. Marcus. 1995. Text chunking using transformation-based learning.

Sunita Sarawagi and William W Cohen. 2004. Semi-markov conditional random fields for information extraction. In *Advances in Neural Information Processing Systems*, volume 17. MIT Press.

Hanna Suominen, Sanna Salanterä, Sumithra Velupillai, Wendy W. Chapman, Guergana Savova, Noemie Elhadad, Sameer Pradhan, Brett R. South, Danielle L. Mowery, Gareth J. F. Jones, Johannes Leveling, Liadh Kelly, Lorraine Goeuriot, David Martinez, and Guido Zuccon. 2013. Overview of the share/clef ehealth evaluation lab 2013. In *Information Access Evaluation. Multilinguality, Multimodality, and Visualization*, pages 212–231, Berlin, Heidelberg. Springer Berlin Heidelberg.

Buzhou Tang, Hongxin Cao, Yonghui Wu, Min Jiang, and Wang Qi. 2013. Recognizing clinical entities in hospital discharge summaries using structural support vector machines with word representation features. *BMC Medical Informatics and Decision Making*, 13.

Buzhou Tang, Hongxin Cao, Yonghui Wu, Min Jiang, and Hua Xu. Recognizing clinical entities in hospital discharge summaries using structural support vector machines with word representation features.

Bailin Wang and Wei Lu. 2018. Neural segmental hypergraphs for overlapping mention recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 204–214, Brussels, Belgium. Association for Computational Linguistics.

Bailin Wang and Wei Lu. 2019. Combining spans into entities: A neural two-stage approach for recognizing discontiguous entities. *CoRR*, abs/1909.00930.

Zhen Wang. 2022. Modern question answering datasets and benchmarks: A survey.