Hybrid Approach for End-to-End Entity Recognition and Entity Linking using CRFs and Dependency Parsing

Anshik*
Advanced Data Science
ZS Associates Inc.
anshik.anshik@zs.com

Vinit Gela*
Advanced Data Science
ZS Associates Inc.
vinit.gela@zs.com

Sagar Madgi
Advanced Data Science
ZS Associates Inc.
sagar.madgi@zs.com

Abstract

This paper introduces a hybrid approach to entity recognition and entity linking using Conditional Random Fields The approach integrates co-reference resolution system, dependency parsing and intelligent Named Entity Recognition through Linear Chain Conditional Random Fields, to identify family members and map them to their corresponding prevalent disease. The Paper presents the novel approach towards solving the BioCreative/OHNLP Challenge 2018

1 Introduction

Dynamic entity recognition and linking entities with relationship is one of the most challenging problems in Biomedical NLP. A version of this problem was presented as a challenge as part of the BioCreative OHNLP challenge 2018 [1] where we had to identify family members and link them to their respective prevalent diseases using unstructured textual notes of family history (1) We leveraged open-source meSH ontology, coupled with word2vec trained on textual data to train a domain-specific word vector(esp. for diseases) and then segmented into relevant clusters of similar semantic information using K-means. Both the methods increased our coverage for disease recognition. Along with these features we used POS tags, head token for a word, to train our linear-chain CRF with window of size 2.

The paper also shows a novel use case of co-reference resolution in family history text where family history of the patient and its relatives were mixed. Co-reference resolution helps in de-referencing pronouns present in the text to its base form. A heuristic score was also formed to define closeness of two recognized entities based on dependency tree representation of text. Lower the score, higher the closeness and hence chance of association.

[1] https://github.com/ohnlp/

2 Techniques :- An Overview

Interpreting language for a machine is hard, mainly because of two reasons:-

1. Ambiguity and author style:- Since there are multiple ways to describe the same thing, ambiguity is yet to arise. Moreover there are different ways to convey the same meaning.

   (a) Patient's father had cancer
   (b) When diagnosed patient's father was found to have cancer.

   Such variations make the analysis complex for a machine.

2. Lack of contextual information :- machines sometimes are not able to "remember" the information present in the preceding sentence/paragraph, this convolutes interpretation of a sentence which derives meaning from the preceding sentence/paragraph.

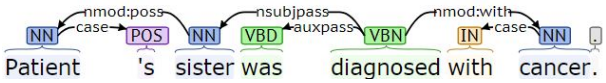
   (a) Patient's father is 68 years of age. He has cancer.

   Then when asked **who has cancer?** a naive system will answer **he**

But now with computational scale to make LSTMs/GRUs (1) work, deriving contextual information is not a major challenge., but then deep learning systems of the likes of LSTMs needs a good amount of data to train on.

Under this section we would like to discuss various key techniques we used in designing our system.

Figure 1: Dependency Tree Example



2.1 Word2Vec and Semantic Clusering using K-Means

Word2Vec is a technique published by Mikolov et.al (3) that represents words in vector spaces, such that the semantic relationship is captured by the vector similarities. We use the Word2Vec model trained on Google News Corpus (3) incrementally on our family history text data to transform words in the data to efficient embeddings. Once we have n-dimensional vector (n=300 in our case) we cluster the word embeddings into groups using KMeans clustering algorithm. Based on some seeded words, we identify which clusters primarily belong to the disease clusters that contain all disease related words. In order to make the process completely automated, we use the disease seed words, tagged as diseases in the training data. For example when we use the word syndrome as a seed word, we see that words as paralysis, sclerosis, palsy, dystrophy, dystonia, granulomatosis also a part of the cluster that syndrome is a part of, and such words are likely to be diseases.

2.2 Dependency Tree Parsing

Dependency-based methods for syntactic parsing have become increasingly popular in natural language processing in recent years.In general, We can remove structural ambiguity in a formal way and that is through Parsing. Dependencies provides a representation of grammatical relations between words in a sentence. [Ref Fig 1.] [2]

We have used dependency parsing to find link between named entities(identified in subtask 1) following which we find the shortest path length between these entities. Multiplying the shortest path length and absolute difference of word token index in the sentence gave us a score. Then those family members having the minimum of the score for a disease was linked to the member.

2.3 Coreference Resolution

Coreference resolution is the task of finding all expressions that refer to the same entity in a text. It is an important step for a lot of higher level NLP tasks that involve natural language understanding such as document summarization, question answering, and information extraction.[3]

We used coreference resolution to do pronoun dereferencing. We did it because of two main reasons.

1. Our Entity linking was based on a score which was a product of shortest path between two entities and absolute difference between their position in the text. For the scoring to hold correct the entities should come out to be close. But due the way clinical notes were present this was not possible in all cases.
   (a) Eg:- Patient father's is years hold. He has two brothers. He is diagnosed with cancer.[Ref Fig 2.]
2. The clinical notes contained family history of not only the patient's family, but also family history of patient's dependents.
   (a) Eg:- Patient's husband is 27 years old. He has two brothers. One of the brother has prostate cancer while the other one in is in good health.[Ref Fig 3.]

3 System Description

This section contains about our system design and setup of different components discussed under Section 2 and how they are used together (2)

Data Processing:- The data provided in the hackathon was present in text files, with each text file accompanied with a xml file detailing the entities and the relations between them. There were also some metadata present for the entities. Metadata varied with type of data like **FamilyMember** and **Observation** had different metadata. Some of the files did not have utf-8 character encoding, this was handled during system input design. All the text was processed in lowercase without any stemming or lemmatization operations on tokens present in the sentences.

Ontology and Word2Vec Clusters :-
We used meSH ontologies[4] to identify whether a word is a identified as a disease as per the ontology class. In addition to this, to capture more words, that meSH might have missed out, we used

[2] https://nlp.stanford.edu/software/stanford-dependencies.html

[3] https://nlp.stanford.edu/projects/coref.shtml

[4] https://meshb.nlm.nih.gov/

Figure 2: Co-reference Resolution Study:- Case 1, Coref

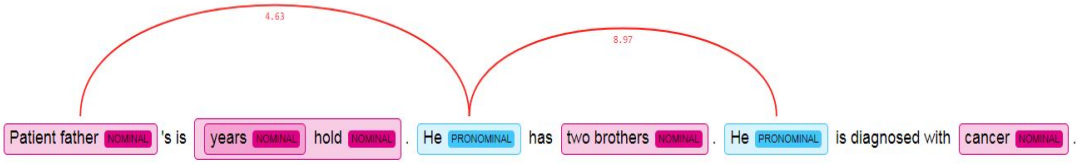
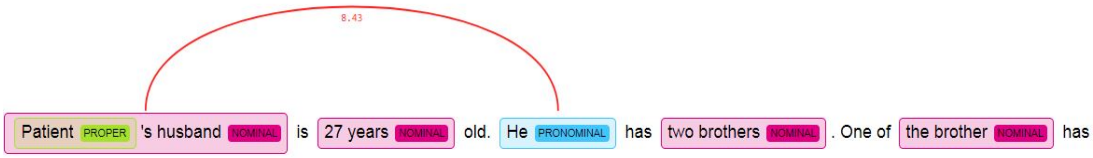


Figure 3: Co-reference Resolution Study:- Case 2, Coref Chaining



word2vec clustering, and identified potential disease clusters. Thereby we add two additional features

1. Belongs to meSH disease class

2. Belongs to w2v disease clusters

Named Entity Recognition :- We used a linear chain CRF (4) to do Name Entity Recoginition. We were predicting for two entities Observations and FamilyMember.

Features for CRF :- Besides creating features using ontologies and w2v clustering, We created features such as head word and dependency type for a token when seen in a dependency tree. We also added POS tags for token in consideration. All these features for a token were then taken with a window of size 2 and trained on a linear CRF to predict the entities

Note that side of family for a Family Member was decided after searching for ”Maternal” or ”Paternal” in a size 2 window around the identified word.if found the appropriate side was added for the Family Member otherwise NA was added. Outputs from CRF was used for Entity Linking, explained below.

Entity Linking :- Before doing entity linking we replaced pronouns like he or she with its main mentions using coreference algorithm. In some of the cases as discussed in case 2 of co-reference resolution. We removed all sentences which were a part of co-reference chaining. Following which we had a text, which had the family member and its disease mention as near as it could be. After this we ran dependency parser on the text to get the output containing dependencies amongst word tokens. While using dependency parser, we did not go ahead with a full text present in the doc, We tokenized the text into meaningful sentences and then proceeded with parsing.We also converted multi token entities and replaced spaces with _. For eg :- breast cancer was replaced with breast_cancer.

What we took from the dependency parsing were two things

1. Length of shortest path.

2. Absolute difference between node id of the tokens present in sentence.

Using the identified metrics from the dependency tree, we multiplied them.For a disease the family member with a minimum score was linked to it.

Living Status Score :- Living status score was assigned after NER for family member. Following which in a 3 size window around the recognized family member following words were searched for

1. Healthy :- ”healthy”,”well”,”health”

2. Death :- ”died”,”dead”,”death”,”passed”

3. Alive :- ”alive”,”living”

For cases tagged as death a score of 0 was assigned whereas for alive a link to disease was checked if true a score of 2 otherwise 4. Healthy family members were also given a score of 4.

4 Results

The results are summarized below. The results are on hold-out set (85:15 split)

Table 1: Family History Information Extraction

	Precision	Recall	F1
Subtask1	0.91	0.88	0.89
Subtask2	0.76	0.68	0.72

5 Conclusion

For a small sample size, use of advanced techniques such as LSTM, was proving to be less accurate, hence a combination of entity recogition using CRFs and entity linking using traditional natural language processing methods like Dependency Parsing and Co-reference resolution was adapted.

References

1) Sepp Hochreiter and Jurgen Schmidhuber. 1997.long short term memory, 9(8):17351780. Institute of BioInformatics.

2) Sijia Liu, Majid Rastegar Mojarad, Yanshan Wang, Liwei Wang, Feichen Shen, Hongfang Liu. Overview of BioCreative/OHNLP 2018 Family History Extraction Task. BioCreative 2018 Workshop Proceedings

3) Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space

4) John, L., Andrew, M., Fernando, P.: Conditional random fields: probabilistic models for segmenting and labeling sequence data. In: International Conference on Machine Learning, pp. 282289 (2001)