Medical Named Entity Recognition in Twitter data

Aditi Mavalankar (201201049)

Prameela kavya Thummuru (201301180)

Udbhav Kalra (201505571)

Mentor:

Nikhil Priyatam

Introduction

Named Entity Recognition (NER) is an unsolved problem in the field of Information Retrieval. Huge amounts of data, when processed to extract named entities, such as names of institutions, people, etc., is bound to have errors, due to a variety of factors, some of which include incorrect data entry, inconsistency in the same named entity across documents, false identification of a term or a phrase as a named entity, to name a few.

When we look at this problem in the medical domain, it becomes even more complex in that the terms used in medical jargon are not the same as those used in written English language. This makes the specialized extraction even more difficult, as the rules and features for the medical data are different from those of ordinary data. As a result of this, this remains an open subject for discussion and further improvement, and pharmaceutical companies are constantly on the lookout for such developments that would benefit them, as Twitter is one of the most widely used social media, and thus, medical tweets would help them find out more about different diseases in various parts of the country or world, and also to find out more about their rival companies and their products.

This project aims at parsing named entities and recognizing and classifying medical data into the relevant categories, namely drugs, diseases, symptoms, side-effects, treatment, etc. Twitter data will be the input and based on previous medical data from databases and ontologies, relevant medical terms have to be parsed and classified (medical named entities are recognised and classified based on the category they belong to(ex: drug or a disease or cure etc...)

What is Named Entity Recognition ?

- Identifying proper names in text, and classification into a set of predefined categories of interest.
- It is the cornerstone of Information extraction,, providing a foundation from which it is needed to build complex information extraction systems.
- Named entity recognition from medical texts involves two main tasks:

  (i) identification of entity boundaries in the sentences.

  (ii) entity categorization.

Why do we need Named Entity Recognition?

The social web has expanded in recent years, with people freely expressing their opinions on a variety of topics. When making a decision, more and more people look online for opinions related to whichever they are interested in, and they often base their final decision on the information found.

One of the pre-processing techniques applied for opinion mining is the recognition of named entities of interest, from which related opinions can be identified and assessed as positive or negative.

Problem Statement

- The task of a Medical Name Entity Recognizer is to identify medical entities in text.
- Medical entities can be diseases, drugs, symptoms, side effects, etc.
- Previously, researchers in the field have used hand crafted features to identify medical entities in medical literature.
- We attempt to uncover features that will push the accuracy beyond what has been obtained so far.
- In this work, we wish to extend medical entity recognition on tweets.
- Twitter data is highly disorganized and prone to inconsistencies.
- The task also involves filtering these inconsistencies from the Twitter data to obtain
- We are expected to use NLP toolkits designed for processing tweets along with other medical ontologies (or databases) to exploit a lot of semantic features for this task.

Dataset

- We have a dataset of 1 year of tweets about 4 diseases and 32 drugs.

- A team of domain experts has annotated about 2000 tweets with
  → entities (around 20 types: diseases, drugs, symptoms)
  → relations (around 40 relation types: cures, causes, etc).

Sample 1:

★ Annotation:

T1 Disease 1 7 Asthma

★ Text:

"Asthma plus anxiety plus food is a bad time."

Sample 2:

★ Annotation:

T1 News 0 61 "AstraZeneca's patent on asthma drug invalidated by US court.

★ Text:

"AstraZeneca's patent on asthma drug invalidated by US court. Watch video - http://t.co/WUR7Qg4VM5"

Algorithm

1. Parsing and tokenizing tweets.

2. Using training data labels to generate the feature files (for both 1-gram and 5-gram models).

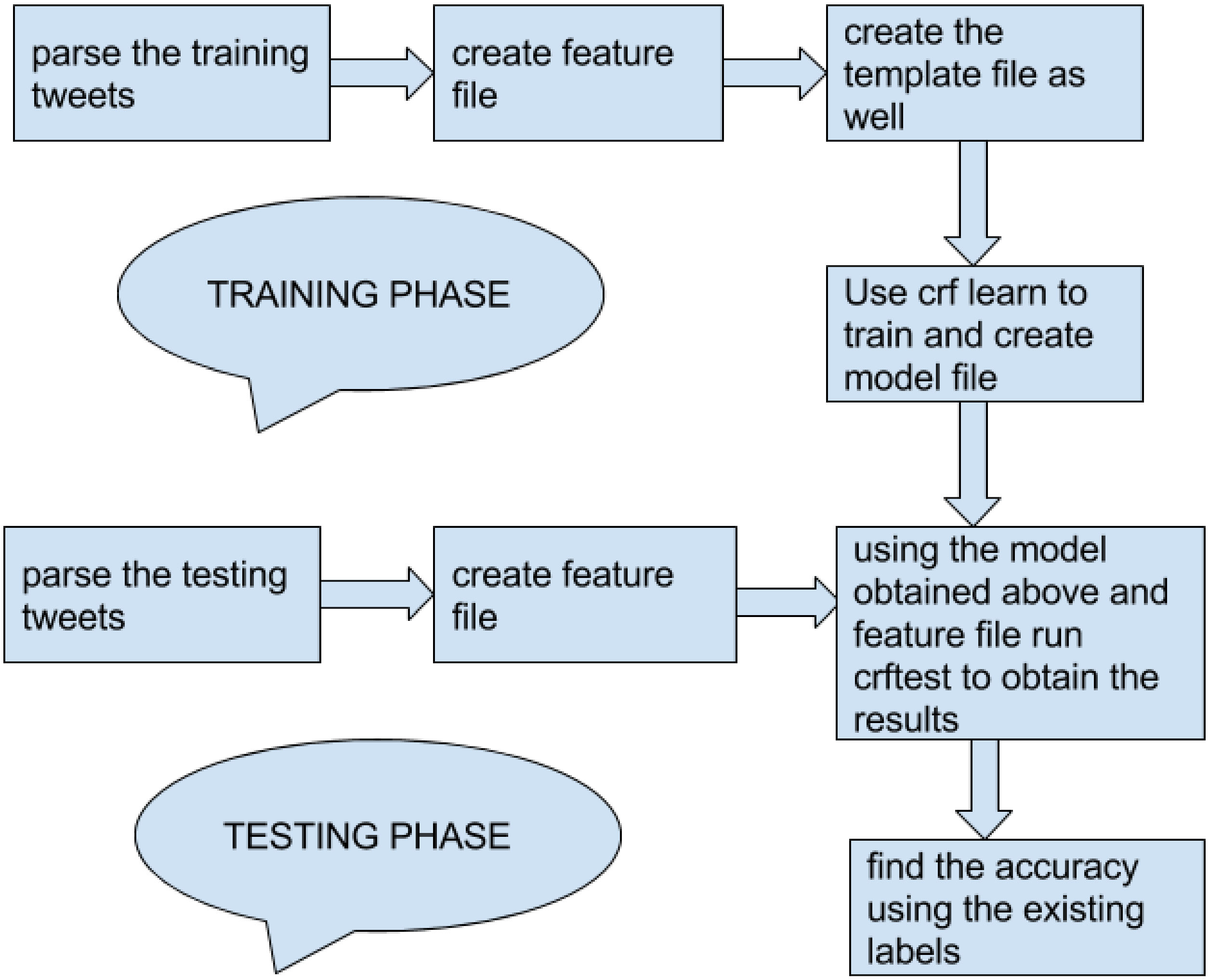
3. Using the output feature file generated in step 2, along with the template file, we use crf_learn command to generate a model file (for both 1-gram and 5-gram).

4. We now generate the feature files for the testing data, excluding the labels.

5. Using the output feature file generated in step 4, along with the template file, we use crf_test command to get the labels for the test data.

6. We compare the predicted labels with the actual test data labels to get the percentage accuracy.

Procedure


parse the training tweets
create feature file
create the template file as well
Use crf learn to train and create model file
TRAINING PHASE
parse the testing tweets
create feature file
using the model obtained above and feature file run crftest to obtain the results
TESTING PHASE
find the accuracy using the existing labels

Procedural Flow

Phase 1

- In phase 1, we used a 1-gram model, as well as a 5-gram model for class prediction. The intent was to see the performance in the most basic feature set used. As a result, none of the properties of the words taken into consideration were included.

- For the 1-gram model, the only feature used was the word itself. This, as is obvious, is not a very useful technique. However, the main intent was not to have a very good performance, but to have a minimum accuracy, with which future performances could be measured.

- For the 5-gram model, along with the word itself, its two successors and two predecessors were also used as features. This is a slight build-up on the 1-gram basic model, as a few more features are taken into account. However, those features are also words, and none of their attributes are taken into account, as a result of which not a great leap in accuracy or performance against any database is observed.

- The idea here is to see how inclusion of additional features can improve performance. However, it is also worthy of being noted that more features does not necessarily imply a better performance. It is very likely that the inclusion of unnecessary and noisy features will reduce the performance of the classifier, and give a lesser accuracy. However, in this case, the inclusion of two successors and two predecessors works to our advantage.

Phase 2

- In this phase, we used a variety of features in the given paper, and have also used a few other features, selected after careful experimentation with the other set of features that could be used.

- The features that we used are as under:

    - Word features : The word itself, two words before and three words after, along with their lemmas

    - Morphosyntactic features : POS tags of the word itself, two words before and three words after.

    - Semantic features : Semantic category of the word, provided by Metamap+

    - Other features : Next noun, previous verb

    - Additional features : Previous adjective, next verb

- There also exist orthographic features, such as:

    - The word contains -, +, &, etc.

    - The word is a number, letter, punctuation, etc.

- ➢ The word is in uppercase, capitalized, etc.
- ➢ Prefixes of different length (from 1 to 4)
- ➢ Suffixes of different length (from 1 to 4)

However, we did not use these features as our domain included tweets, which rarely follow the general rules of language. Also, hyphens, back slashes and other such punctuations would also be present in hyperlinks, which are very common in Twitter data, as a result of which the resulting classification would be very poor. In order to avoid these situations, we have eliminated the orthographic features for this particular dataset.

Our observations regarding these features are as follows. On adding the two extra features, i.e. the previous adjective and the next verb, we found a significant change in the accuracy. We will explain the case with the adjective.

Adjectives are a very good indication of the term and its classification. In the medical domain, they are especially helpful as they assist in the classification of medical named entities. For instance, ‘mild fever’ has a very suggestive adjective ‘mild’ that can help the classifier classify the term ‘fever’. As a result, POS tags of surrounding words also play a very important role in identifying the class since this can incorporate grammatical features in particular to language, this also helps us a lot in classifying the named entities.

Toolkits used

1. CRF++

   In our case, named entity extraction system is modeled with a CRF model. Theoretically, the CRF models are best performing models for sequential labeling.

   CRFs are a type of discriminative undirected probabilistic graphical model. It is used to encode known relationships between observations and construct consistent interpretations.

2. Metamap+

   Identify Unified Medical Language System (UMLS) Metathesaurus concepts in text.

   MetaMap uses a knowledge-intensive approach based on symbolic, natural-language processing (NLP) and computational-linguistic techniques.

3. NLTK

   NLTK is a suite of libraries and programs for symbolic and statistical natural language processing (NLP) for the Python.

Challenges involved

i. Relevance

- ❏ Identifying relevant content from tweet (all tweets containing the keyword 'asthma' are not about the disease 'asthma')
- ❏ Number of information to be extracted -
  Medication name, reason, context, cause. In order to do that, a robust system is needed to process all these information quickly and relevantly.

ii. Noise

- ❏ Filtering noise from the tweets
- ❏ The diverse and noisy style of user-generated social media text presents serious challenges in terms of performance also.

iii Misspellings:

- ➢ Examples:
  - ❏ “Mitral stenosis is not present and definate mitral regurgitation is not seen.”
  - ❏ “Sclarea anicteric.”
  - ❏ “ventilator dependent respiratoy failore”

iv Unusual part-of-speech combinations:

❑ **Adjective without noun modified:**

Example:

“Head, eyes, ears, nose, and throat examination revealed normocephalic and atraumatic.”

v. Variety of communication styles:

❑ Examples:

❑ “He was also tachycardic”

❑ “He had a fingerstick of 142.”

iv. Others

❑ Entity linking for exploiting semantic features from ontologies (UMLS, MetaMap).

❑ Learning distributed representations for medical tweets.

❑ Information available in various languages - example NER research has been carried out on medical notes written in english but not on chinese.

- ❏ Number of information metrics to be extracted-
  Medication name, dosage, mode of administration, frequency, duration, reason, and context.
  A robust system is needed to process all this information quickly.

- ❏ Identifying complementary features and algorithm (ex. Based on the test results, it was found that combining word segmentation and section information procedures as results.

- ❏ Clinical staff only use practical processing systems. An estimate is that the acceptable accuracy is over 95% which leaves a 5-10% gap from the current processing systems.

Accuracy

Phase 1:

- 1-gram model : 62.23%
- 5-grams model : 75.10%

Phase 2:

- 91.97%

Analysis

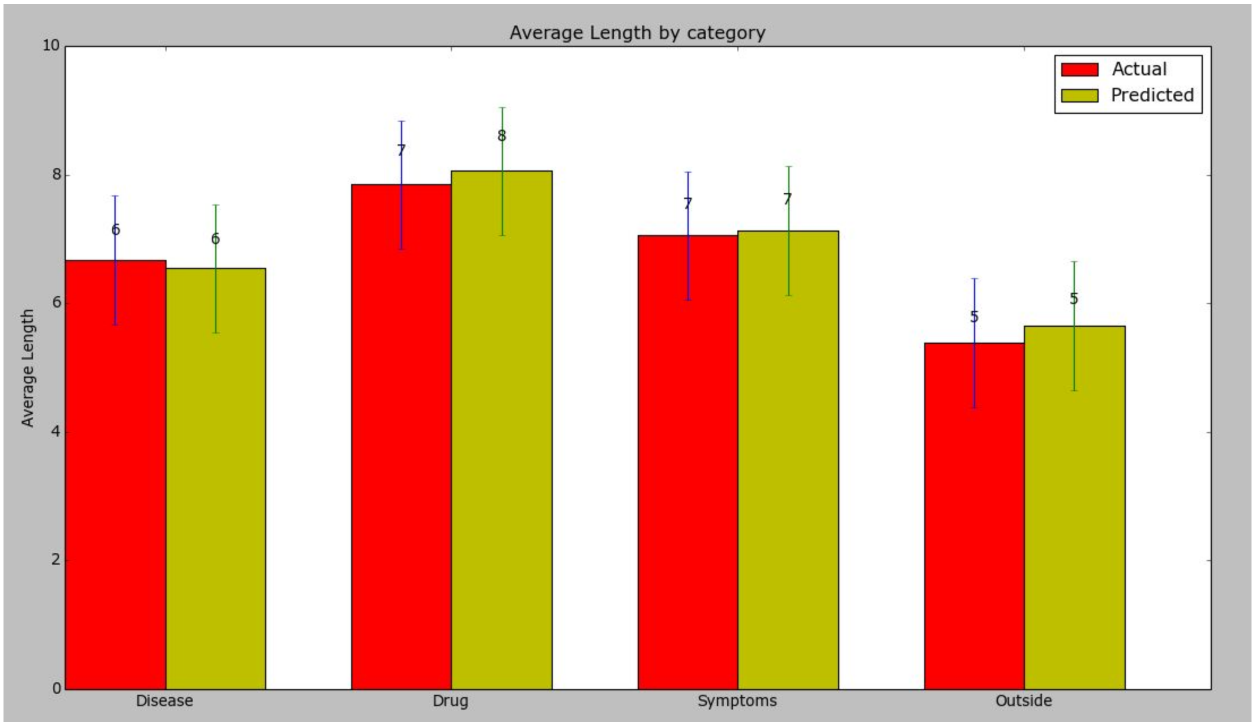
If we closely analyse the corpus and the output of our process, we have a great variation in terms of lengths and the occurrences of certain features like:

- ❏ Average length of:
  - ❏ Categories (including Diseases, Drugs, Symptoms)
- ❏ Occurrences of :
  - ❏ POS Tag features (verbs, nouns, adjectives, noun-plural, preposition)

1. Average Length Analysis:

In this case, we can judge on basis of the average length, that for example if we have a word with length "8" and a word with length "4", it is likely to be a drug name or a symptom not disease name.

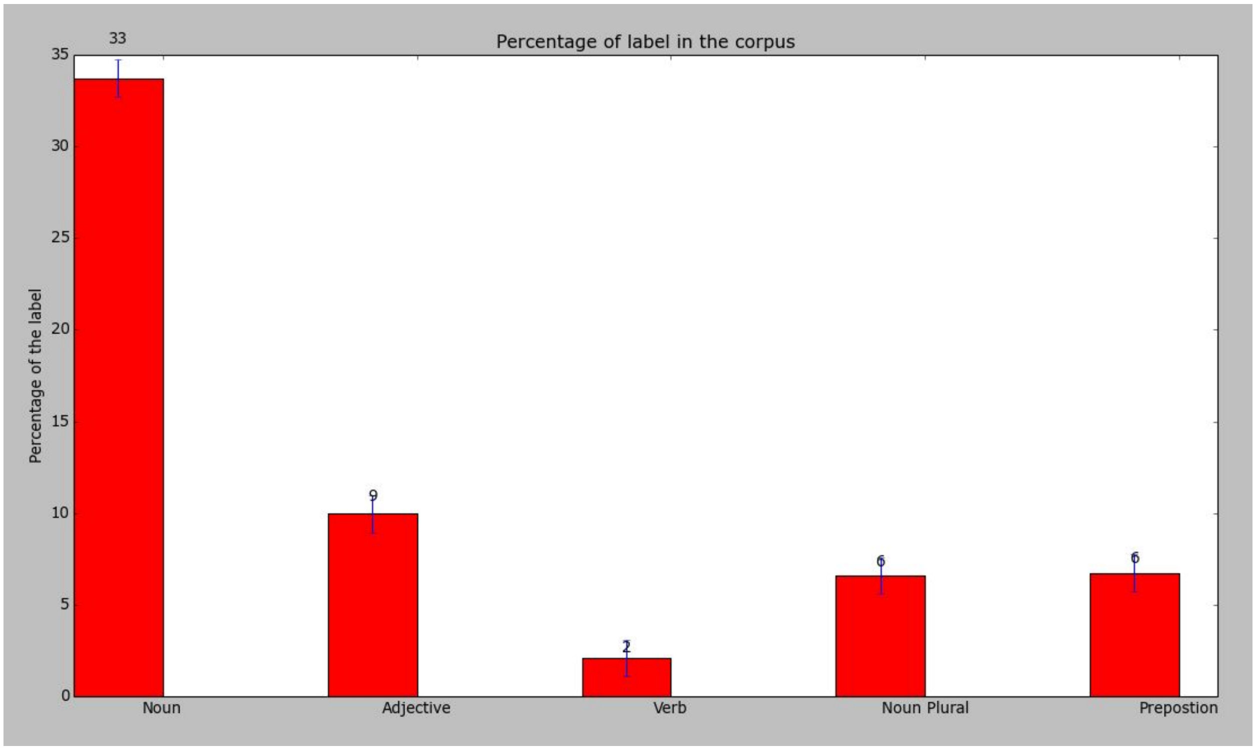
Though, this is not a good assumption but this feature can be added to increase the prediction state.



2. Percentage of POS tag features:

In this case, we can say that the probability of a word to be a “noun” is higher than any other feature (for this corpus).

This can be used to predict the type of the nature of the word present in the corpus. But, as tweets have a lot of noise, this feature wouldn’t help much in terms of prediction of the category but according to the study, this gives some idea about the occurrence of various features.



Applications

- The results of analyzing such data will be used by pharmaceutical companies to boost their sales and also procure knowledge about sales of drugs manufactured by other companies pertaining to the same disease.

- These results will also be beneficial in getting an estimate of the presence of any disease in a particular region and its prevalence.

Confusion Matrix

659	0	0	0	0	0	0	0
0	194	1	7	11	0	1	282
0	0	12	0	0	0	0	73
0	16	0	9	1	0	0	53
0	13	0	0	52	0	5	124
0	0	0	0	0	0	0	34
0	1	0	2	7	0	12	60
0	111	22	12	35	2	10	9184

Project URLs

- Git web page:

  http://prameelakavya.github.io./

- Git repository for code:

  https://github.com/prameelakavya/ire_major_project_medicalNER_code/

- Video URL:

  https://www.youtube.com/watch?v=fjALuLdnxwM&feature=youtu.be

- Slideshare URL:

  https://www.slideshare.net/secret/ySiKibYZ1CilgF

References

- Medical Entity Recognition: A comparison of semantic and statistical methods
- Enhancing clinical concept extraction with distributional semantics
- CRF++ documentation: https://taku910.github.io/crfpp/
- Metamap+ documentation: https://metamap.nlm.nih.gov/
- Scikit-learn documentation: http://scikit-learn.org/stable/documentation.html