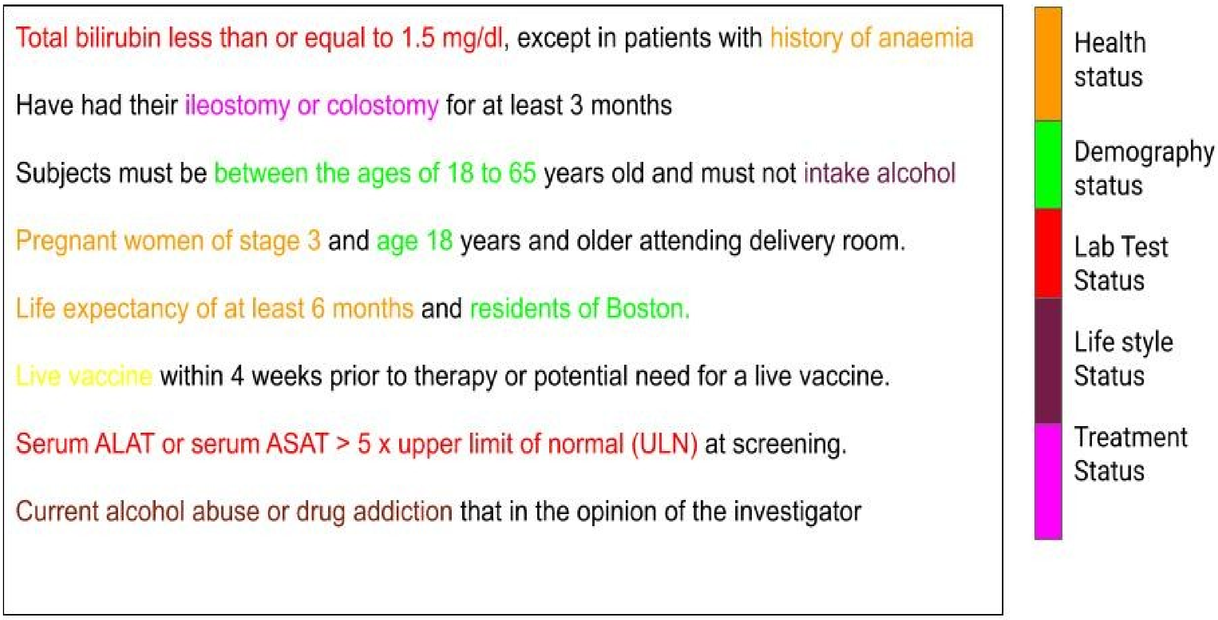
Dataset Details With Examples:

Sample dataset annotations in which there can be multiple aspects:



Dataset Annotation Heuristics:

a. PyMeshSim annotations for collecting the disease names

b. We also take into account the DrugBank Vocabulary to collect the instances of Drugs and chemicals for determining the drugs related to the aspect of “Treatment”.

c. The Lab-Test Results are mostly generated based on rules like pattern identification of “ upper limit of normal”, “greater than or equal to” etc.

d. CliNER Tagger for determining the aspects like “Health-Status”, “Treatment” and “Lab-Tests”. Although CliNER tagger tags the health, treatment and lab-test spans almost correctly, the manual experts verified that some of the spans are not correctly tagged by the tagger based on the some requirements such as:

1. “Life expectancy greater than 50 years” must be tagged as “health status”, which was incorrectly tagged as O.

2. The sentence “Have an implantable device and not having stage III cancer” was not correctly tagged by the tagger. “ Have an implantable device” must belong to the Treatment Class.

Challenges:

We also investigate the effect of bag-of-words pattern-based heuristics in correctly identifying the aspects from the eligibility criteria of the clinical trials. Some of the rules have confidence scores less than 100%. Those rules are enumerated as below:

1. "history of" : "history of back pain" denotes "Health" aspect whereas "history of plastic surgery" denotes "Treatment" aspect, "history of daily exercise" speaks about life-style.

2. "greater than" , "less than" are also ambiguous in nature. "greater than 100 ml" denotes "lab-test" aspect, whereas "age greater than 50" denotes "demography" asoect.

In this way, we finally arrive at a conclusion that, in addition to simple pattern-based heuristics, semantic information is equally helpful to extract out the aspects with higher accuracy. Semantic Knowledge in terms of vector embeddings prove to be useful in our experiments.

Implementation and Hyperparameter Details:

For many sequence labeling tasks it is benefi-cial to have access to both past (left) and future(right) contexts. However, the LSTM's hiddenstatehttakes information only from past, knowing nothing about the future. An elegant solution previouswork is bi-directional LSTM (BiLSTM). The basic idea is to present each se-quence forwards and backwards to two separate hidden states to capture past and future information, respectively. Then the two hidden states are concatenated to form the final output. For sequence labeling (or general structured pre-diction) tasks, it is beneficial to consider the cor-relations between labels in neighborhoods andjointly decode the best chain of labels for a given input sentence. Therefore, we model label sequence jointly using a conditional randomfield (CRF), instead of decoding each label independently. Finally, we construct our neural network model byfeeding the output vectors of Bi-LSTM into a CRF layer. For each word, the character-level representation is computed by the CNN in with character embeddings as inputs. Then the character-level representation vector is concatenated with the word embedding vector to feed into the Bi-LSTM network. Finally, the output vectorsof Bi-LSTM are fed to the CRF layer to jointly de-code the best label sequence. Dropout layers are applied on both the input and output vectors of Bi-LSTM. It is to note that the character embed-dings are initialized with uniform samples with dimension 20.

To mitigate overfitting, we apply the dropout to regularize our model. We also apply dropout on character embeddings before inputting to CNN, and on both the input and out-put vectors of BLSTM. We fix dropout rate at 0.5 for all dropout layers through all the experiments.We obtain significant improvements on model performance after using dropout.