CS584 Assignment 1

Vir Mittal

Code

The code for the assignment is available at https://github.com/Vir-Mittal/CS584-Assignment-1

Abstract

This paper analyzes the performance of a rule-based system in extracting symptoms from social media (Reddit) posts for the purpose of data collection. The paper will also discuss errors made by the system and possible improvements that can be made.

Annotation

The first step in developing the rule-based system was to understand the data and produce a sample annotated file that could be tested against the system developed. Approximately 20-25 posts were manually annotated and for each post, the CUIs (Concept Unique Identifier) of the symptoms mentioned in the post were listed using a symptom lexicon. Along with the CUI, a negation flag was also listed, in order to signify if the symptom was negated or not. The annotation file is available at https://github.com/Vir-Mittal/CS584-Assignment-1/blob/master/vir_mittal_annotations_s5.xlsx

Annotation Agreement

The manually annotated files were compared among students to produce IAA (Inter-Annotator Agreement) values which depicted how likely agreement amongst annotators was for the same posts. The values ranged from 77%-100% with 100% depicting complete agreement. The diverse range of values could have been a result of differences in the number of posts compared for each annotator.

In general, a majority of the values were between 80%-90% showing that there was good agreement amongst the annotators. Disagreements could have arisen in cases where a symptom was mentioned multiple times in the same post, sometimes with a positive negation flag, and sometimes with a negative negation flag. In this case, some annotators may have listed the symptom with both flags, while some may have only listed it with one flag. Disagreements could have also arisen when an annotator identified a phrase as the symptom "Other", but a different annotator did not.

Rule-based System

The rule-based system makes use of exact matching along with regular expressions to identify if a symptom is mentioned in the post. Using the symptom lexicon provided, the system iterates over each phrase mentioned in the lexicon and uses regular expressions to check if that same phrase is also mentioned in the post. If it is, then the CUI of that phrase is listed. The system also checks what the few words (=4) before the usage of the phrase are to identify if the phrase (and thus symptom) is negated or not. This is done using String exact matching with a predefined list of negation words. This helps provide the negation flag for the corresponding CUI. Through this process, each post ends up with a corresponding list of CUIs and negation flags for symptoms mentioned in the post. The Python code for the system is available at https://github.com/Vir-Mittal/CS584-Assignment-1/blob/master/vir_mittal_assignment1.py

Evaluation

The system was used on the posts contained in the manually annotated file and the gold standard file provided. The annotated file produced by the system was then compared to the original files with the manual annotations using an evaluation script which calculated the total number of true positives, false positives, and false negatives by comparing the CUI and negation flag lists for each post in the corresponding files. The score metrics produced are displayed in Table 1.

Table 1: Evaluation Scores

Annotation File	Recall	Precision	F1-Score
Manual Annotation File s5	0.69	0.73	0.71
Gold Standard Set	0.65	0.78	0.71

Discussion

The F1-score obtained in both cases was 0.71, indicating that the rule-based system was 71% accurate in producing the annotations. Comparing this to the IAA values (80%-90% agreement) shows that the rule-based system's performance is not as reliable as manual annotation, but it is not far behind.

Closer look at the scores shows that the number of false negatives was much higher than the number of false positives (75 vs 39, for the gold standard set). This shows that a big flaw of the system was that it was unable to recognize some symptoms that were present in the text, mostly because the system relied heavily on the predefined lexicon for matching.

One component of this was that the system was unable to recognize phrases which had symptoms that the lexicon did not have. In this case, manual annotation would list the symptom as "Other" or "C0000000". However, the system did not understand what a symptom phrase looks like and instead relied on exact matching using the lexicon. Thus, it was unable to correctly list "Other" or "C0000000" in any of the posts.

Another error that the system made was that it was unable to recognize short informal phrases that authors of the posts made. An example of this is "sob" which stands for "shortness of breath". Since "sob" was not present in the lexicon, the system was unable to identify it as a symptom.

The system was also unable to correctly identify phrases such as "lack of taste/smell". Using the lexicon, the system could identify "lack of taste", but was unable to capture the context of the entire phrase to recognize "lack of smell" as well. Manual annotation would correctly list both.

On the other hand, false positives occurred as the system identified multiple symptoms within the same phrase. For example, the phrase "chest pains" would be manually annotated as "Chest Pain CUI C0008031". However, the system would iterate over each symptom in the lexicon and attempt to match it to the phrase. Thus, it would find a match for "chest pains" and list "Chest Pain CUI C0008031". It would then find a match for "pains" and list "Body Ache & Pain CUI C0741585" as well. Thus, in the same phrase, the system found two symptoms, but an annotator would correctly only list one.

Improvements

An improvement to the lexicon would be beneficial to the system, however it is not always possible to do this. Improvements can also be made to the system itself and the rules within the system.

Fuzzy matching can be used within the search to help with inexact but similar matches. This would help identify symptoms even if the post had small mistakes in spelling.

Another improvement could be to identify "/" in the sentences and repeat the words which occur before the "/" again, for the word which immediately comes after it. This would split the phrase "lack of taste/smell" into "lack of taste" and "lack of smell", allowing the system to correctly identify both symptoms.

Another improvement targeting false positives could be to remove a phrase from the sentence once a symptom is found to match that phrase. As discussed earlier, the system sometimes detects multiple symptoms within the same phrase. Removing the phrase once a symptom is matched, would mean that each phrase can only be matched with one symptom. This solution might work for a small lexicon, however it would be difficult to implement with a large lexicon. This is because the system would simply match the first matched symptom in the lexicon. For the phrase "chest pains", if "Body Ache Pain CUI C0741585" is listed before "Chest Pain CUI C0008031" in the lexicon, then the system would just match "Body Ache Pain CUI C0741585". This is incorrect as "Chest Pain CUI C0008031" is more detailed and more accurate. The lexicon would have to be pre-sorted according to specificity, with the most specific symptoms such as "Chest Pain CUI C0008031" occuring before general symptoms such as "Body Ache Pain CUI C0741585". This may not always be possible to do, but is still worth mentioning.

Conclusion

In conclusion, the rule-based system is good at identifying symptoms based on phrases listed in the lexicon. However, it is not as good as manual annotation because the system relies heavily on the lexicon and is unable to identify deviations from it. The improvements discussed could help with some general cases that arise across posts.