Manolache George Vlad

Balagiu Darian

Balan Petru

Micu Hontan Valentin

# DELIVERY 1 REPORT

RULE-BASED MODEL

## HYPOTHESIS

Our hypothesis is based on the next statements:

1. Negation and uncertainty cues are easier to find than medical terms, so we will start out search with the former
2. NEG and UNC amount to much more constrained sets of words than NSCO and USCO, so we can expect better precision on the former
3. Scopes have unpredictable length and are a lot more likely to extend on various words
4. We need to generate a corpus that contains medical terms that are likely to appear in documents

## CORE ALGORITHM

All of the approaches that we have tried share the same starting steps:

1. Create two sets for Negation and Uncertainty cues called NEG and UNC
2. Parse the documents from the training set. Parse the Ground Truth NEG and UNC labels and add the cues to the two sets created above.
3. The search will be based on the cues: we will create two regexes by putting together all the words from NEG and UNC using “|” operations. We will match the regexes on each text and for each match found, we will then employ a particular algorithm to find the corresponding scope.

NOTE: This approach is exactly the opposite to NegEx, in which they were searching for medical terms and then verified if they are inside a negation/uncertainty cues’ scope.

## ANALYZING THE DATA

We start by creating lists of highlighted words (NEG, UNC, USCO, NSCO) for the training set and test set in order to compare similarities between them. These are results:

|  | NEG | UNC | NSCO | USCO |
| --- | --- | --- | --- | --- |
| WORDS FROM TEST ALSO IN TRAINING | 1118/1132 | 118/131 | 599/1074 | 20/129 |

Table 1

Judging by this table, we can expect high precision for both of the cues, medium precision for NCSO and bad for UCSO. That is because, as mentioned in the hypothesis, cues are part of a much more constrained set of words than scopes.

There are some words from NEG that are also found in UNC (such as some occasional “no”, “not”). Keeping these words in UNC will cause a lot of noise because the GT rarely categorizes them as UNC instead of NEG. That is why we are going to remove words from UNC that are also in NEG.

### APPROACH 1: CUTEXT Medical Terminology Parser

CUTEXT is a tool used for extracting medical terms from a corpus. Because of the lack of terminology in the medical field (and also our inability to read Catalan to extract useful information from the documents) we tried creating our own medical corpus. The resulting medical terms generated a set called CUTEXT\_SCOPES, then a regex was created by piping together (“|”) all the terms from it, and for each cue found, the regex was being matched to the next or previous 5 words. This was successful, however in the end the results were not great:

|  | NEG | NCSO | UNC | USCO |
| --- | --- | --- | --- | --- |
| PRECISION | 92.3% | 15% | 56% | 8% |
| RECALL | 98% | 17% | 77% | 11% |
| F1 | 95% | 16% | 65% | 9% |

The reason for that is because there was not enough similarity between our personally created corpus and the test set.

### APPROACH 2: BASELINE

Our BASELINE approach starts by following the core algorithm mentioned above to create the NEG and UNC sets for cue prediction and uses the same idea to create NCSO and UCSO sets.

If we take a look at Table 1, we can expect this approach to yield an average performance on NCSO predictions of around 56% (599/1074).

It is worth noting that both the CUTEXT and Baseline approach work with the same NEG and UNC cues.

1. Create a set called ALL\_SCOPES that combines NCSO and UCSO (our search is going to be done for the cues, so there is no reason to separate the scopes) and a regex pattern that pipes all of the scopes together
2. Whenever we match a cue, use the scope regex pattern above on the previous 5 words or next 5 words

We are aware that 44% of NCSO scopes and 85% of UCSO will not be found because they were not learnt from the training set, but this is just a Baseline approach. We will try to find optimizations to solve this problem later.

After running the BASELINE model on the training set and test set provided, we obtained the following results:

|  | NEG | NCSO | UNC | USCO |
| --- | --- | --- | --- | --- |
| PRECISION | 91.1% | 43.3% | 53.3% | 8% |
| RECALL | 97% | 48.5% | 73.3% | 12% |
| F1 | 94% | 45.7% | 61.7% | 10% |

The NEG prediction performs very well and rises to the level of expectation.

The NCSO prediction does not reach its maximum potential of this approach (56%) and that is because we only took into consideration that the scope will be after the cue. This is a viable optimization that we can undertake.

The UNC does no rise to the expectations and that is because of the words that we removed some useful words from the set (because they were also in NEG and hence the model would not be able to distinguish between the two, especially if it is only rule-based)

### APPROACH 3: BASELINE + CUTEXT

This was done by combining the NCSO and UCSO sets of the previous two methods. The following results are obtained:

|  | NEG | NCSO | UNC | USCO |
| --- | --- | --- | --- | --- |
| PRECISION | 92.5% | 51.7% | 57% | 21% |
| RECALL | 97% | 57% | 77% | 28.7% |
| F1 | 94.7% | 54.2% | 66% | 24% |

Improvements can be seen in all areas.