Project Instructions

Pau Torras Lei Kang

Introduction

For the project you will implement three different Sequence Tagging techniques to solve the task of Detection of Negation and Uncertainty:

- A rule-based algorithm using basic text processing tools.
- A machine-learning system.
- A Deep Learning system. For this one, we will ask you to at least employ a simple LSTM, but you may optionally choose to go beyond this scope.

Data

nº historia clinica: ** *** *** nºepisodi: ********
sexe: dona data de ...

d'hospitalitzacio motiu d'ingres trabajo de parto
antecedents no alergia medicamentosa conocidas ap:
epilepsia en tratamiento no intervenciones quirurgicas
no transfusiones no habitos toxicos medicacio habitual

...

serologias: rubeola inmune, toxoplasma no immune, lues
vih, vhb y vhc negativos. - o'sullivan: 81 - urocultivo:
negativo - cultivos r / v:

...

el 2.08.18 se indica cesarea por sospecha de perdida de
bienestar fetal. a las 20.25 h se obtiene recien nacido
vivo mujer de 3.380 gr, apgar 9(10, ph 7.22-7.27.
hemostasia correcta. sondaje vesical: orina clara.
procedimiento sin incidencias. intradermica en piel. el
pueperio clinico ...

Negation cues
Negation scope
Uncertainty cues
Uncertainty scope

Figure 1: Display of a sample in the dataset with some example cues for negation and uncertainty and their affecting scopes

Figure 2: Example of the annotation of a negated scope within the dataset

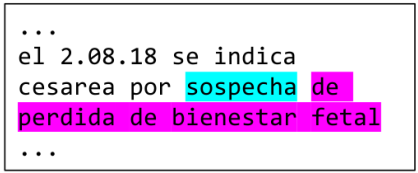

Figure 3: Example of the annotation of an uncertain scope within the dataset

Tasks

Step 1: Literature Review

The first step when trying to solve a task is checking whether somebody has solved it before. **Surveys!**

- A. Mahany, *et al.*, "Negation and Speculation in NLP: A Survey, Corpora, Methods, and Applications," ApplSci, vol. 12, no. 10, Art. no. 10, Jan. 2022, doi: 10.3390/app12105209.

For each type of algorithm we ask you to develop, we provide you with a series of basic references that you may choose to base your methods on. **You can go beyond this scope**

Step 2: Method Design

Once you have an idea of how other people deal with similar tasks as your own, you are in an informed position to design your own method.

- Adapt something from the literature
- Design a system of your own

Step 3: Data pre-processing

This depends a lot on the method! You may have to adapt it depending on what you do afterward.

- Redacted entities:

```
nº historia clinica: ** *** *** nºepisodi: ******** sexe: home [...]
                     == === ===             ========
```

- Catalan and Spanish!
- Misspellings
- Dealing with punctuation
- Redundant or irrelevant information
- Make it evaluatable

Step 4: Implementation (Rule-Based Method)

Baseline idea: **NegEx algorithm**

`<Prefix Cue> WORD{0, 5} <UMLS Terms>`

`<UMLS Terms> WORD{0, 5} <Postfix Cue>`

Your final method can use any of the following:

- Pre-defined list of negation trigger words
- Regular expressions
- Part of Speech tagging
- Syntactic parsing

Step 4: Implementation (Machine Learning Method)

Any Machine Learning algorithm that is not a Deep Neural Network

You may train classifiers based on text features (PoS, lemma, syntactic features, word embeddings, ...) for each of the two sub-tasks (detection of negation/uncertainty signals and detection of the negation/uncertainty scope).

An example method is the one seen in Enger *et al.*, for which you have an implementation in here.

Step 4: Implementation (Deep Learning Method)

The last negation/uncertainty detection method you are tasked to implement will use a Deep Neural Network or Deep Learning algorithm. **The most basic model you should use is an LSTM like the ones seen in the reference section.** You may choose to employ other architectures such as Sequence to Sequence or Transformers. Bear in mind computational costs.

Methodology

You should apply the scientific method for any system you implement.

- **Design a hypothesis**
- **Design an experiment to test your hypothesis**
- **Analise the results of the experiments and reevaluate your hypothesis if needed**

You should **always** back your conclusions with data. Statements must be held by the results of the experiments.

Calendar

The project will be developed **strictly** in **groups of 4**. You will have to write a report, do a final presentation and deliver the code for all implemented methods. You will have three project follow-up sessions, each with an associated delivery.

29/04/2024: First Follow-Up Session (Rule-Based)

You are expected to have accomplished the following:

- Understanding of the problem and the data.
- Finding and reading prior work in the literature of the field for rule-based methods.
- Designing and implementing a rule-based system.
- Evaluating and extracting conclusions for the rule-based system.
- Writing the report including the work until this point.

5 days before this delivery, we will upload the test partition. Before this, you should work only with your own defined train/validation partitions. Results in the report should include test scores as well. You will be provided with a test script.

Deliverables

You have to deliver a first version of the report containing all of this information. We will ask you some questions during the session and give you feedback.

15/05/2024: Second Follow-Up Session (ML-Based)

You are expected to have accomplished the following:

- Finding and reading prior work in the literature of the field for machine learning-based methods.
- Designing and implementing a machine learning-based system.
- Evaluating and extracting conclusions for the machine learning-based system.
- Adding the work until this point to the report.

Deliverables

You have to deliver a second version of the report containing all this new information. We will ask you some questions during the session and give you feedback.

29/05/2024: Final Project Presentation + DL-Based

You are expected to have accomplished the following:

- Finding and reading prior work in the literature of the field for deep learning-based methods.
- Designing and implementing a deep learning-based system.
- Evaluating and extracting conclusions for the deep learning-based system.
- Performing a comparison of all three methods.
- Writing the final version of the report.
- Preparing a 10' presentation of your work.

Deliverables

You will have to deliver the final version of the report, the slides for the presentation and the code you have written for the project within a Git repository.

Grading

The final grades for the project are computed as follows:

$$\text{Grade} = 0.6 \cdot \text{Deliverables} + 0.3 \cdot \text{Presentation} + 0.1 \cdot \text{Individual Eval}$$

- The deliverables grade is obtained from your report submissions. Each report submission gets $1/3$ of the total weight of the mark. Each report submission in structured with 80% given to the report itself and the remaining 20% to the questions we make. The last submission only includes the score for the report itself.
- The presentation grade is obtained from your final presentation. Correctness, tone, respecting time, responses to questions, insights from the project and pace will be considered.
- The Individual eval comes from the contribution of each member to the group.

Tools

It is recommended that you check the following libraries

- NLTK package: Basic LM utilities.
- Spacy: Suite of language-related utilities for Python.
- Scikit-Learn: Collection of machine learning models and utilities.