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.

The goal of this task is to find any words that indicate the start of negated (such as "no" or "without") or uncertain (such as "maybe", "perhaps" or modal verbs) clauses and finding the scope of text that is affected by them. You will work on a dataset that contains reports of medical exams written in **Catalan** and **Spanish**.

Data

The dataset you will be working with is a collection of real medical notes that have been manually labelled to find negation cues and negated sequences of words. In Figure 1 you may see an example of one of such notes from the dataset.

The dataset is packaged in a JSON file structured as seen in Figures 2 and 3. Bear in mind that the only information we care about is the text itself and the labelling of the negation/uncertainty words and scopes. Therefore, for each sample you have to collect the "text" key and the "predictions" key. In the latter you can find a list of scopes expressed as character ranges and their corresponding class from the "start", "end" and "labels" keys respectively.

Tasks

The goal of the project is to implement **three** different methods to solve the task of negation detection:

- A handcrafted rule method.
- A Machine Learning method.

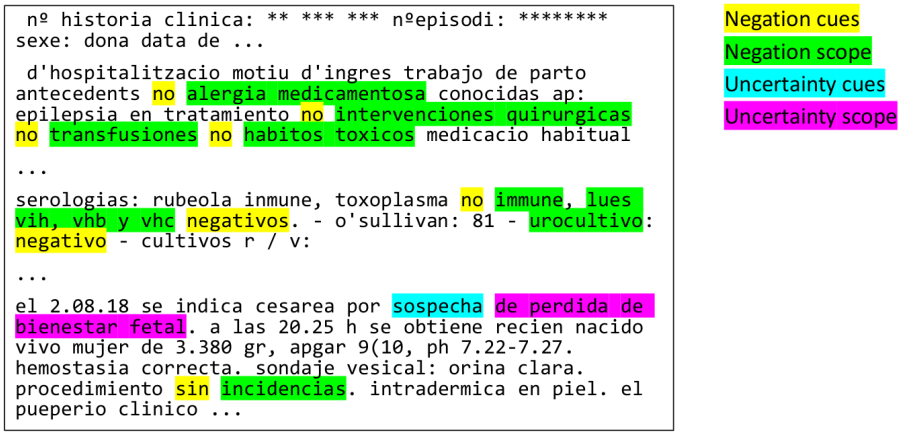

Figure 1: Display of a sample in the dataset with some example cues for negation and uncertainty and their affecting scopes


```
{"data":{"cmbd": "null",
          "id": "19062854",
          "docid": "null",
          "page": "null",
          "paragraph": "null",
          "text":" nº historia clinica:..."},
          "annotations":[],
          "predictions":[{"result":
         [{"value":{"start": 347,
                    "end": 350,
                    "labels": ["NEG"]},
                   "id":"ent0",
                   "from_name":"label",
                   "to_name":"text",
                   "type":"labels"},
         {"value":{"start": 350,
                   "end": 372,
                   "labels": ["NSCO"]},
                   "id":"ent1",
                   "from_name":"label",
                   "to_name":"text",
                   "type":"labels"},
         ...
```


nº historia clinica: ** *** *** nºepisodi: ******** sexe: dona data de naixement: 20.06.1999 edat: 19 anys procedencia domicil/res.soc servei obstetricia data d'ingres 02.08.2018 data d'alta 06.08.2018 11:28:06 ates per **********, *****; teixido troyano, anna informe d'alta d'hospitalitzacio motiu d'ingres trabajo de parto antecedents no alergia medicamentosa conocidas ap ...

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

- A Deep Learning method. For this one we will ask you to at least implement an LSTM-based

one, but you are free to explore any other methods you deem interesting (beware of the computational costs though).

To do so, you will perform a series of tasks.

Literature Review

The first step when trying to solve a task is checking whether somebody has solved it before or if somebody else's ideas are applicable. A good starting point is usually a survey or a review of the field, from which you may find the most relevant papers and the key ideas to follow. We provide you with a survey to read (check the attached pdfs).

- A. Mahany, H. Khaled, N. S. Elmitwally, N. Aljohani, and S. Ghoniemy, "Negation and Speculation in NLP: A Survey, Corpora, Methods, and Applications," Applied Sciences, 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**

Rule-Based methods

- **Basic NegEx algorithm:** W. W. Chapman, W. Bridewell, P. Hanbury, G. F. Cooper, and B. G. Buchanan, "A Simple Algorithm for Identifying Negated Findings and Diseases in Discharge Summaries," Journal of Biomedical Informatics, vol. 34, no. 5, pp. 301–310, Oct. 2001, doi: 10.1006/jbin.2001.1029.

- **Application of NegEx to Spanish:** R. Costumero, F. Lopez, C. Gonzalo-Martín, M. Millan, and E. Menasalvas, "An Approach to Detect Negation on Medical Documents in Spanish," in Brain Informatics and Health, D. Ślezak, A.-H. Tan, J. F. Peters, and L. Schwabe, Eds., Cham: Springer International Publishing, 2014, pp. 366–375. doi: 10.1007/978-3-319-09891-3_34.
- **Application of NegEx to Spanish + some additional ideas:** O. Solarte Pabón, M. Torrente, M. Provencio, A. Rodríguez-Gonzalez, and E. Menasalvas, "Integrating Speculation Detection and Deep Learning to Extract Lung Cancer Diagnosis from Clinical Notes," Applied Sciences, vol. 11, no. 2, Art. no. 2, Jan. 2021, doi: 10.3390/app11020865.

Machine Learning methods

- R. Morante and W. Daelemans, "A Metalearning Approach to Processing the Scope of Negation," in Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL-2009), S. Stevenson and X. Carreras, Eds., Boulder, Colorado: Association for Computational Linguistics, Jun. 2009, pp. 21–29. Accessed: Apr. 09, 2024. [Online]. Available: https://aclanthology.org/W09-1105
- M. Enger, E. Velldal, and L. Øvrelid, "An open-source tool for negation detection: a maximum-margin approach," in Proceedings of the Workshop Computational Semantics Beyond Events and Roles, E. Blanco, R. Morante, and R. Saurí, Eds., Valencia, Spain: Association for Computational Linguistics, Apr. 2017, pp. 64–69. doi: 10.18653/v1/W17-1810.
- J. Beltran and M. Gonzalez, "Detection of Negation Cues in Spanish: The CLiC-Neg System," 2019.
- H. Loharja, L. Padró, and J. Turmo Borras, "Negation cues detection using CRF on Spanish product review texts," presented at the NEGES 2018: Workshop on Negation in Spanish: Seville, Spain: September 19-21, 2018: proceedings book, 2018, pp. 49–54. Accessed: Apr. 09, 2024. [Online]. Available: https://upcommons.upc.edu/handle/2117/123065

Deep Learning methods

- **LSTM Methods in Spanish:** H. Fabregat, J. Martinez-Romo, and L. Araujo, "Deep Learning approach for Negation Cues Detection in Spanish".
- **LSTM Methods in Spanish:** H. Fabregat, A. Duque, and L. Araujo, "Extending a Deep Learning Approach for Negation Cues Detection in Spanish," 2019.
- **LSTM Methods in English:** F. Fancellu, A. Lopez, and B. Webber, "Neural Networks For Negation Scope Detection," in Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), K. Erk and N. A. Smith, Eds., Berlin, Germany: Association for Computational Linguistics, Aug. 2016, pp. 495–504. doi: 10.18653/v1/P16-1047.

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. You may choose to do an implementation that is similar to the literature or you can be creative, assert the shortcomings of the existing methods and create one of your own. Bear in mind also that there is a slight change of domain from the literature to this task, so whatever you choose to do you will not be able to do 100% literal translations.

Data pre-processing

Once you have designed the methods you want to employ to solve the task, the next thing you will have to do is to **pre-process** the data. In here you will find some pointers as to where to start and some standard steps you should consider, **but you are free to explore alternatives or to implement the subset of steps you deem necessary.**

- You will see that there are plenty of redacted entities in the text (using sequences of asterisks of variable length):

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

- Catalan and Spanish are mixed together in the prose. Keep that in mind for negation cues.
- There are plenty of misspelled words. Consider correcting the words or lemmatising them.
- Depending on the algorithms you choose to use, you may want to keep or remove punctuation. You may also want to split the text into individual sentences.
- It might be interesting to remove the patient information at the start to avoid false positives.
- When tokenising you should keep the coordinates of the original text for evaluation.

This step is highly dependent on the next ones, and you should motivate your decisions consequently. Do not expect the exact same pre-processing to work for all methods or techniques equally. Provide evaluation results to check whether a certain change is impactful or not (this is the idea of an ablation study).

Rule-Based Method

The first negation/uncertainty detection method you are tasked to implement will be using handcrafted rules. The idea is that you develop a system that

can detect negation by identifying recurrent structures for negated or uncertain clauses. A prototypical implementation of this idea is the **NegEx algorithm**, seen in the references, which identifies negated or uncertain cues by applying regular expressions:

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

UMLS terms are part of a normalised dictionary of medical terms. Prefix and Postfix cues are pre-determined sets of negation cues placed before or after the corresponding medical terms.

You may find an implementation of this method in the following link: https://github.com/chapmanbe/negex/blob/master/negex.python/negex.py.

Your final method can use any of the following:

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

You may find it useful when implementing successive methods to use this method as your "baseline". A baseline is a simple method that can be used as a starting reference point from which to compare more sophisticated methods. If you check scientific literature for Machine Learning methods, it is a very recurrent idea.

Machine Learning Method

The second negation/uncertainty detection method you are tasked to implement will use any kind of Machine Learning algorithm that is not a Deep Neural Network or Deep Learning algorithm. 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 https://github.com/marenger/negtool.

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**: e.g. "Taking into account word will improve performance of the system because "
- **Design an experiment to test your hypothesis**. Force a set of circumstances for which your hypothesis should happen. For instance, if your hypothesis states that a model should become more robust to a specific phenomenon, evaluate the performance of the model on samples containing that specific characteristic. Sometimes the generic metrics you are using for performance might be enough, but perhaps you have to incorporate new measurements that reflect the behaviour of the model in such circumstances. In that case, define them adequately.
- **Analise the results of the experiments and reevaluate your hypothesis if needed**. If your initial hypothesis does not hold when tested, try to assess why and create new hypotheses accordingly. Continue the cycle as many times as necessary (or possible, given the time restrictions).

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.