Universitat Autònoma de Barcelona

Fundamentals of Natural Language
Project Report

DETECTION OF NEGATION AND UNCERTAINTY

Students:
Júlia Garcia, 1630382
Josep Maria Rocafort, 1631378
Nerea Qing Muñoz, 1631552

31st May 2023

Contents

1 Introduction

The primary objective of this project is to enhance the detection of negation and uncertainty cues in medical reports text. The project incorporates both rule-based and deep learning techniques to address this objective.
The rule-based approach involves the creation of predefined linguistic rules and patterns, specifically designed to identify and extract instances of negation and uncertainty cues.
On the other hand, the deep learning approach leverages neural network models. These models are trained using labeled data, in which input text samples are paired with annotations indicating the presence of negation and uncertainty cues, as well as their extents or scopes.
You can access the GitHub repository associated with our report by following this link: Github repository.

2 Dataset description

In our project, we utilized a JSON file containing sample texts and annotations of Negation and Uncertainty. This dataset was divided into two main subsets: a training set and a validation set.
The training set accounted for 70 % of the entire dataset, while the remaining 30% was allocated for the validation set. This division ensured that we had a sufficiently large portion of the data for training our model while also setting aside a separate set of data for evaluating its performance.
The JSON file contained various text samples, which could be sentences or paragraphs, along with corresponding annotations indicating the presence of Negation and Uncertainty in each text. These annotations allowed us to label and classify the instances of negation and uncertainty within the texts accurately.
You can access the dataset at the following link: Github repository.

3 Rule based approach

The detection process was divided into two distinct steps: cue detection and scope detection. Each step will be explained separately in dedicated sections, and the evaluation has been done on the first diagnosis.

3.1 Cues detection:

The detection of cues involves examining the JSON files to identify the labels of NEG and UNC, which indicate the presence of relevant words. These words are extracted and stored in a dictionary along with their corresponding labels.

```
{'keyword : 'no', 'category': NEG}
{'keyword' : 'sospechar', 'category': UNC}...
```

After creating the dictionary with the lemmas of the relevant words and their corresponding labels, we iterate through all the tokens of the input text to check if they appear in the dictionary. If a token is found in the dictionary, it is stored, allowing us to classify it as a cue.

```
for token in doc:
        if token.lemma_ in dict:
negation_indices.add(token.i)
```

3.2 Scopes detection:

To determine the most effective method for scope detection, we employed three different approaches. Each approach was evaluated individually to assess its performance and determine the optimal method.

3.2.1 Fixed-size window

This initial method is the most basic approach. It involves creating a fixed-size window, where the size can be adjusted. In our implementation, we utilized a window size of 4, meaning that the scope is formed by the subsequent four words following the detection of a cue.
Indeed, this basic method may not perform well in all cases. One of the limitations is that it can include too many words within the scope, which might lead to noise or incorrect interpretations.

```
Predicted scope: ['hijos', 'tiene', 'un', 'hermano']
Real scope: ['hijos', 'o', 'o', 'o']
```

Opposite scenarios can also occur where not enough words are considered within the scope. This can result in missing important contextual information or failing to capture the complete meaning of the sentence.

```
Predicted scope: ['alteraciones', 'en', 'el', 'contenido', 'o', 'o']
Real scope: ['alteraciones', 'en', 'el', 'contenido', 'del', '
    pensamiento']
```

Ideal approach if all scopes had the same size, but not in this case. The following figure depicts the quantitative results obtained.

```
True Positives: 27
True Negatives: 0
False Positives: 5
False Negatives: 18
+-----------+-------+
|  Measure  | Score |
+-----------+-------+
| Precision |  0.84 |
|   Recall  |  0.60 |
|  F1-score |  0.70 |
|  Accuracy |  0.54 |
+-----------+-------+
```

Figure 1: Quantitative Report

3.2.2 Finding patterns: Noun + Adj

The second method we devised builds upon the previous approach. Instead of considering a fixed number of words after a cue, we extend the scope until we encounter a noun followed by an adjective. In other words, the scope encompasses the detected cue and continues until this specific noun-adjective pattern is encountered.
However, this method still faces similar challenges as the previous one. In some cases, it may include excessive words because the actual scope might not adhere to the pattern of a noun followed by an adjective. As a result, the method continues adding words until it identifies such a pattern, potentially leading to an expanded scope beyond what is necessary.

```
Predicted scope: ['hijos', '.', 'tiene', 'un', 'hermano', 'con', 'el', '
    que', 'tiene', 'contacto', 'en', 'barcelona', '.', 'vive', 'en', '
    apartamentos', 'tutelados']
Real scope: ['hijos', 'o', 'o', 'o', 'o', 'o', 'o', 'o', 'o', 'o', 'o',
    'o', 'o', 'o', 'o', 'o', 'o']
```

Indeed, the second method may also fail to include all the necessary words within the scope. Various factors can contribute to this, such as encountering a noun followed by two adjectives instead of one, encountering conjunctions that introduce additional information, or encountering other linguistic structures that deviate from the specific noun-adjective pattern. These possibilities highlight the challenges in accurately determining the scope and the need for further refinement in the approach.

```
Predicted scope: ['alergias', 'mediamentosas', 'o']
Real scope: ['alergias', 'mediamentosas', 'conocidas']
```

Despite the limitations mentioned, the second method can still yield successful results in certain cases.

```
Predicted scope: ['alteraciones', 'en', 'la', 'sensopercepcion', 'ni', 'otras', 'alteraciones', 'dentro', 'de', 'la', 'esfera', 'psicotica']
Real scope: ['alteraciones', 'en', 'la', 'sensopercepcion', 'ni', 'otras', 'alteraciones', 'dentro', 'de', 'la', 'esfera', 'psicotica']
```

The following figure depicts the quantitative results obtained.

```
True Positives: 31
True Negatives: 0
False Positives: 30
False Negatives: 14
+-----------+-------+
|  Measure  | Score |
+-----------+-------+
| Precision |  0.51 |
|   Recall  |  0.69 |
|  F1-score |  0.58 |
|  Accuracy |  0.41 |
+-----------+-------+
```

Figure 2: Quantitative Report

The results are worse than the previous method. The presence of false positives was a significant challenge, leading to a decrease in overall accuracy. This highlights the need for further refinement and optimization to reduce false positives and improve the precision of cue detection and scope identification.

3.2.3 Introduction of Dependency Trees

We incorporated dependency trees using the Spacy Stanza library to enhance the detection of words affected by negations and uncertainty cues. By analyzing the relationships between words in the sentence, we aimed to capture a more comprehensive understanding of the contextual impact.
The quantitative evaluation confirmed the positive impact of incorporating dependency trees.

```
True Negatives: 1
True Positives: 54
False Positives: 5
False Negatives: 28
+-----------+-------+
|  Measure  | Score |
+-----------+-------+
| Precision |  0.92 |
|   Recall  |  0.66 |
|  F1-score |  0.77 |
|  Accuracy |  0.62 |
+-----------+-------+
```

Figure 3: Quantitative Report

The qualitative analysis revealed promising outcomes with the inclusion of dependency trees. The refined scope better captured the intended context, allowing for more accurate identification of words influenced by negations and uncertainty cues. Here we have an example of the performance of the system:

```
no valorables . - tc abdominal : glandula pancreatica de pequeño tamaño , atrofica , con lipomatosis
difusa , sin identificar se lesiones focales ni
dilatacion significativa de el conducto pancreatico .
ureterohidronefrosis bilateral secundaria a globo vesical , observando se una vejiga de paredes trabeculadas
probablemente en relacion a patologia prostatica . evolucio clinica a su llegada a urgencias estable ,
afebril , destacando a la exploracion fisica sequedad mucosa . electrocardiograma en el
que destacan t negativas en di y avl y
d2 sin disponer de ecgs previos y equilibrio
acido-base con acidosis metabolica e hiperglucemia &gt ;
750mg/dl con cetonas altas . bajo la sospecha de cetoacidosis
diabetica se inicia sueroterapia con reposicion de
potasio y perfusion de insulina . analitica que
evidencia minima insuficiencia renal asi como leve
elevacion de troponina i en meseta . se solicita
valoracion por cardiologia que realiza ecoscopia sin evidenciar disfuncion sistolica
aparente . en planta permanece estable . revisando analiticas previas ambulatorias , en marzo se objetivaba
alteracion de glucemia en ayunas ( 190mg/dl ) , sin recibir tratamiento .
se solicita analitica con hba1c de 13.9% y
funcion tiroidal que es normal ; marcadores
tumorales negativos . ampliamos estudio con tc abdominal que descarta patologia tumoral
```

Figure 4: Sample Text

3.3 Conclusions

In conclusion, our approach evolved from basic to more sophisticated methods in order to improve cue detection and scope identification. In this particular dataset, analyzing the relationships between words and utilizing patterns derived from dependency trees yielded the best results. This approach allowed us to capture the context more accurately and achieve higher precision.
However, it's important to note that the effectiveness of the approach can vary depending on the specific dataset. In some cases, the initial basic approaches may yield satisfactory results while requiring fewer computational resources. It is crucial to consider the nature of the data and experiment with different techniques to determine the most suitable approach for the task at hand.

4 Deep Learning Approach

4.1 Character-Based Approach:

The character-based approach was considered as a baseline for our LSTM models, allowing us to explore token complexity from the lowest to higher levels. Although a character-based model may require longer training time, it has the potential to generalize better and handle mixed languages as well as out-of-vocabulary words, including misspellings or mixed language words.

4.1.1 Data Preparation

In the data preparation phase, we extracted texts and annotations from the JSON file to create a dataset. Each text was tokenized at the character level, with each token assigned a corresponding label (NEG, UNC, NSCO, USCO, or O). The dataset was then split into a training set (70%) and a validation set (30%). Finally, we constructed two dataloaders to facilitate efficient handling and processing of the datasets during training and validation.

4.1.2 Model Architecture

For the model, we utilized a straightforward LSTM (Long Short-Term Memory) architecture. The model's structure is as follows:

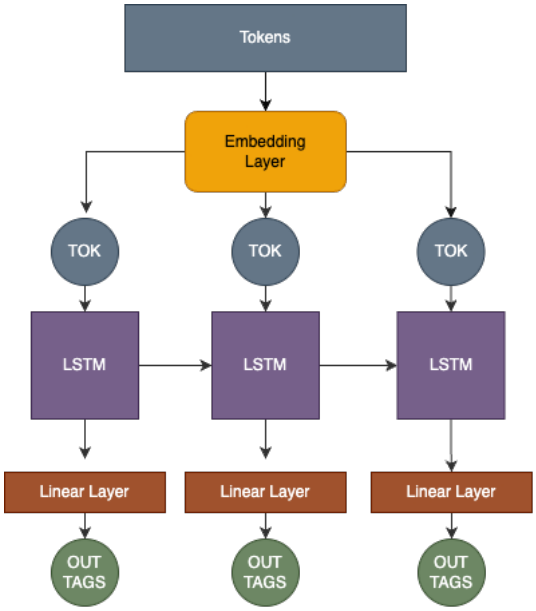

Figure 5: Model Architecture

The same LSTM model architecture was employed for subsequent implementations of this project.

4.1.3 Model Training

In the training process of the character-based model, we iteratively optimized the model's parameters using the chosen dataset. We adjusted various hyperparameters to explore

their impact on the model's performance. Specifically, we experimented with the window of characters seen, hidden size, and character embedding size during the training phase. The goal of this phase was to identify the configuration that yielded the best performance.

4.1.4 Problems and Solutions: Imbalanced Data

Initially, we encountered poor results in our model due to imbalanced data in the training dataset. The prevalence of the "OTHER" label led to most tokens being classified as such, causing difficulties in detecting negation and uncertainty cues and scopes accurately.
After conducting experiments, we found that modifying the loss function yielded improved outcomes. Initially, we applied cross-entropy loss to all output tokens, including the "OTHER" token. However, this imbalance issue caused the model to prioritize classifying most tokens as "OTHER" instead of correctly identifying important ones. To rectify this problem, we made adjustments to the loss function by assigning a higher weight to non-"OTHER" labels.

```
loss_full = criterion(y_pred.transpose(1, 2), y)
loss_specif = criterion_specific(y_pred.transpose(1, 2), y)

loss = loss_full*0.55 + loss_specif*0.45
loss.backward()
```

4.1.5 Model Evaluation

The following figure represents the confusion matrix obtained with the predictions of the validation data, which demonstrates a substantial improvement following the modification of the loss function. While the initial results were not satisfactory, the adjustments made to the loss function significantly enhanced the model's performance.

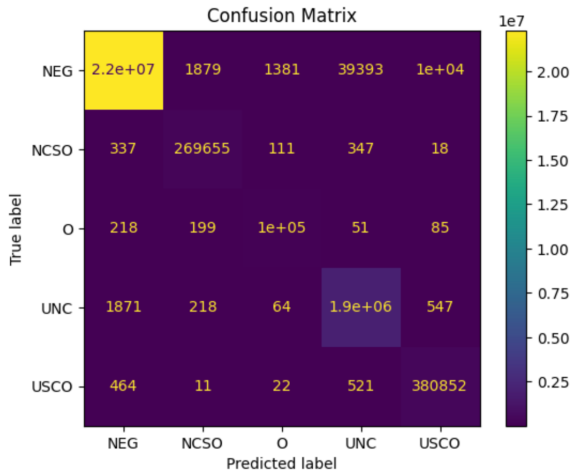

Figure 6: Confusion Matrix

In the qualitative results, we observed that the word "no" played a crucial role in triggering the detection of other tags.

```
NONE USCO UNC NSCO NEG
el paciente no presenta leucoplasia severa en el dorso de la lengua.
```

```
NONE USCO UNC NSCO NEG
el paciente presenta leucoplasia severa en el dorso de la lengua.
```

```
NONE USCO UNC NSCO NEG
aquesta setmana no presentava cap sintoma de febril.
```

4.1.6 Conclusions

Based on these results, it can be concluded that the Char-Based approach did not achieve a high level of success as it failed in numerous cases. In the subsequent sections, we will focus on improving the performance of the Deep Learning model.

4.2 Word-Based Approach:

To try to improve the performance of the algorithm, we decided to explore a different approach. By treating words as individual entities and understanding their relationships within the context of a sentence, we anticipate improved accuracy and a richer understanding of textual content. The process followed in this project is described below:

4.2.1 Data Preparation

In the data preparation phase, a large dataset was created by utilizing the texts and annotations from the JSON file. Each text was tokenized into words and assigned its corresponding label (NEG, UNC, NSCO, USCO, or O). Again, the dataset was split into two subsets: a training set (70%) and a validation set (30%). To efficiently handle and process the datasets during training and validation, two data loaders were implemented.

4.2.2 Model Training

For model training, the same model architecture as described in the section below (Character-Based) was utilized. The training process involved feeding the data from the training set through the data loader, optimizing the model parameters using a specified loss function and optimizer, and iteratively updating the model's weights through backpropagation. The goal was to train the model to accurately predict negation and uncertainty labels based on the input text.

4.2.3 Model Evaluation

To assess the performance of the trained models, we conducted an evaluation using the validation set. We calculated performance metrics such as accuracy, precision, recall, and F1 score to measure the model's effectiveness in correctly classifying negation and uncertainty instances.

Additionally, we generated a confusion matrix to provide a detailed overview of the model's predictions, including true positives, false positives, true negatives, and false negatives.

Here are some qualitative findings from our evaluation of the validation set that demonstrate how the model is able to detect the majority of cues and scopes accurately.

Accuracy: 97.49%
Classification Report:

	precision	recall	f1-score	support
NEG	0.90	0.98	0.94	3500
NSCO	0.75	0.98	0.85	10571
O	1.00	0.98	0.99	157076
UNC	0.93	0.83	0.88	497
USCO	0.86	0.88	0.87	1399
accuracy			0.97	173043
macro avg	0.89	0.93	0.90	173043
weighted avg	0.98	0.97	0.98	173043

Figure 7: Evaluation Report

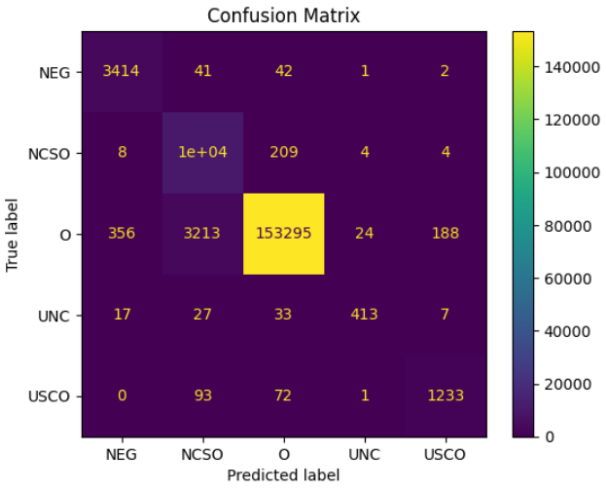

Figure 8: Confusion Matrix

4.2.4 Conclusions

Based on the observed results, our Word-Based approach proved successful in accurately identifying cues and scopes, allowing us to achieve our goal with high precision. Furthermore, we effectively addressed the challenges encountered during the project, particularly the issue of imbalanced data. Through the implementation of appropriate loss functions, we were able to successfully resolve this challenge.

4.3 Fine-tuning BERT

Initially, we had high hopes for fine-tuning BERT, considering its remarkable performance in class for POS and NER tagging tasks. However, our results fell short of expectations. We suspect that the imbalance between the "OTHER" tag and other relevant tags might be the primary underlying cause. Despite attempting the modified loss function, the results remained disappointingly poor.
Similar to the character-based approach, we constructed a dataset where each token was assigned one of the following tags: 'OTHER', 'USCO', 'UNC', 'NSCO', or 'NEG'.
The model itself is a simple wrapper for 'bert-base-multilingual-cased'. It incorporates a basic linear layer in the output, which translates the last hidden state of BERT into the desired tag output size. Dropout regularization is also applied to enhance performance.

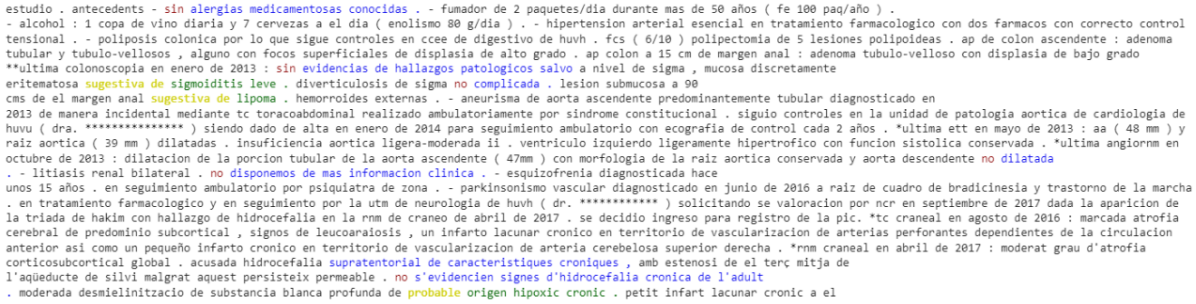

Figure 9: Sample text

```python
class BERT_Tagger(nn.Module):
                    bert,
                    output_dim,
                    dropout):
        super().__init__()
        self.bert = bert
        embedding_dim = bert.config.to_dict()['hidden_size']
        self.fc = nn.Linear(embedding_dim,output_dim)
        self.dropout = nn.Dropout(dropout)

    def forward(self, tokens):
        bert_out = self.bert(tokens)["last_hidden_state"]
        predictions = self.fc(bert_out)
        return predictions
```

Despite the initial promise, fine-tuning BERT did not yield the desired results. It is evident that further investigation and alternative strategies are required to overcome the challenges posed by the imbalance issue and enhance the model's performance.

4.4 Final Evaluation

During our evaluation of various approaches, we concluded that simpler models generally showcased better overall performance. The rule-based approach, despite its simplicity, exhibited commendable computational efficiency and speed. However, it struggled to capture complex patterns that deep learning (DL) approaches excelled at. Among the DL methods, we found that the word-based approach, combined with a basic LSTM, yielded remarkably good results.
While we acknowledge that given more time and effort, the BERT model could have potentially learned intricate patterns and outperformed the word-based approach, this would have come at the expense of increased computational intensity and resource requirements during training.
This project has provided us with valuable insights, highlighting the effectiveness of simple expert systems utilizing rule-based approaches. It served as a notable departure from our primary focus on AI throughout the semester, enlightening us about alternative methodologies that can be equally effective in specific scenarios.