NLP Code Test

On this test you will implement different steps in the NLP pipeline: NER and vectorization. In order to evaluate the results we will take into account 3 factors:

- Methods and algorithms.
- Extraction of conclusions and evaluation of results..
- Code quality.

It is mandatory to use Python and the project will be structured in 3 sections:

- Analysis: supported by Jupyter Notebooks or Google Colab, you will provide the data exploration, the tested algorithms and the results with the corresponding explanations.
- Executables: python files with the final solution.
- Answers: ReadMe.md file with the answers to the questions.

We provide all required resources in the following repository: https://bitbucket.org/delectateam/nlptrainingexam/ . The dataset will be the same for the 2 exercises: **reviews.json**.

Once you finish the exercises, zip everything and send it to Roman Boix. (rboix@delectatech.com)

Parte 1: Named-Entity Recognition (NER)

You will find the dataset in the resources folder: **reviews.json** contains the user comments and **entities.json** the labels. The files are related by review_uid field.

Implement a NER system able to detect the following entities using spaCy system:

- **Conceptos** (concepts): Food service entities, mainly nouns.
- **Modificadores** (modifiers): what is said about each concept and expresses polarity (positive, neutral, negative), usually adjectives.

Once the NER is trained, it will be able to recognize this kind of comments:

"La **paella de marisco** era **bastante cara**, pero el **servicio** fue **excelente**."

You will follow all the steps of a ML project, including scoring and results discussion.

Finally, answer the following questions:

1.1. Which metric have you implemented to evaluate the model? Why?

1.2. Does the model have the capacity to find new entities that aren't present on the training set? If so, are they from the domain? Otherwise, what would you do to improve that ability?

Parte 2: Vectorización de comentarios

The dataset will be the same, this time you will only need **reviews.json.**

Implement a vectorial representation of the client reviews. Once the transformation is done, check the capacity to model similarity between comments. Feel free to use examples, graphics and other solutions to analyze the quality of the solution.

Finally, answer the following questions:

2.1. Do vector representations of the sentences hold information about the sequence (order) of words?

2.2. If you had more data and time, how would you improve the vectorization?