Machine Learning Fundamentals project
Sentiment Analysis with Hidden Markov Model

Rémi Eyraud, Sri Kalidindi, Thibaud Leteno, Richard Serrano

March the 17^{th} - 2023

Along this project you will be asked to implement a Hidden Markov Model (HMM) algorithm to do sentiment analysis. Some functions' skeletons well be given to you, and you can always refer to your HMM class (slides available on moodle).

1 Logistic

- This project concerns the following master programs : COSI, DSC, IMLEX, MLDM.
- You should work in **groups of 3 or 4 students**. Of course, larger groups imply higher expectations.
- A group can extend to **different master programs**.
- The project will be hand in on the **moodle** platform in the associated repository.
- Due deadline : **April the 30^{th} of 2023** at 23:59.
- This project accounts for 30% of the final grade of the Machine Learning Fundamentals module.

2 Code

The code we provide to you is in the form of a Python Jupyter Notebook `https://jupyter.org/`. If you personal machine is not powerful enough, if you struggle with python libraries or for an easy way of sharing the notebook within your group, an easy workaround is to work from the google colab platform[1] `https://colab.research.google.com/`.

You are expected to hand in one jupyter notebook by group on the moodle platform in the associated repository. Your jupyter notebook should start with a Markdown cell containing every team member's first name, last name, master program, and a percentage of work accomplished for the project. Example of group where everybody would have worked equally :

Aiden Boyle,	COSI,	25%
Dakota Etsitty,	DSC,	25%
Finley Gallach,	IMLEX,	25%
Kai Lanka,	MLDM,	25%

Along with this document is a Jupyter Notebook template with functions skeletons and comments, which you must use to perform an HMM forward-backward training algorithm.

2.1 Data

The dataset used for this project is the NER Dataset from Kaggle `https://www.kaggle.com/datasets/debasisdotcom/name-entity-recognition-ner-dataset`. The dataset purpose is to exercise on the Named Entity Recognition task of Natural Language Processing using Part-Of-Speech Tagging. Even though this dataset allows one to do NER, we keep only the sentences of the dataset and focus on the simple task of sentence likelihood prediction to stay in the scope of this class. Fell free to explore the kaggle repository for a better in-depth of the NER task.

[1] Note for this and future project : If you use google colab and you need a GPU to fasten the computation, think of activating it `Runtime > Change runtime type > Hardware accelerator > GPU`.

2.2 Code expectations

Here are some indications on how to handle the practical (coding) part of this project.

1. Download the `project.zip` file available on moodle.
2. Unzip (in bash : `$ unzip project.zip`) the file to get the following :
   - `MLF_project_HMM.ipynb` : this is the jupyter notebook which you must fill and hand in at the end.
   - `data/dataset.csv` : this is a folder called `data`, containing the dataset.
3. Following the notebook instructions..

The project will ask you 4 parts :

Task 1 Similar to how we built the hidden state transition probability matrix as shown above, you will built the transition probability between the words. With this matrix write a function that can calculate the log likelihood given a sentence.

Task 2 Using the model parameters write the Viterbi algorithm from scratch to calculate the best probable path and compare it with the hmmlearn implementation.

Task 3 Form our own HMM

Task 4 Complete the forward and backward functions in the Baum-Welch algorithm and try it with your formulated HMM.

You will have to hand in this notebook : Do not hesitate to use Markdown cells to keep track of what you are doing along the project. You can use Google Colab to share the notebook and run it more easily within a group.

Fill free to try some extras : you could, for instance, pre-process the data differently than what we offer. Think of explaining anything you try in the notebook (briefly) and in you report. Evaluate the impact of your tries and reports the results in your report.

3 Report expectations

Expected element in your report are :

1. Data description

BONUS Pre-processing

2. How you handled each tasks

BONUS Say a word about the algorithm's complexity (or anything else you want).

3. Further work (what you would do to continue this work further on)
4. Conclusion (think of a critical view of your work)