Objective:

You are given a set of news data and the objective is to perform NER on the data to derive a particular security for which the article is.

Description:

These news articles may refer to one or more companies. The goal is to find out the primary company which the article is referring to.

Rules:

a. If article title contain company name or security name that takes precedence over others
b. If article contains one security or company's name multiple times, the company or security mentioned the most would win.
c. If article contains several securities, the company which was mentioned first should win.

Examples:

Input:
Apple Inc and Google started working on COVID-19 tracing application. AAPL stock went up by 1%.

Output:
security: ['AAPL']
securities: ['AAPL', 'GOOG']

Input:
Apple, Amazon and Microsoft are reporting earnings after market close on April 30th.

Output:
security: ['AAPL']
securities: ['AAPL', 'AMZN', 'MSFT']

Input:
Amazon (NASD: AMZN) bought Whole Foods Market (NYSE: WFM) for undisclosed amount.

Output:
security: ['AMZN']
securities: ['AMZN', 'WFM']

References:

https://medium.com/intro-to-artificial-intelligence/entity-extraction-using-deep-learning-8014acac6bb8

https://appliedmachinelearning.blog/2019/04/01/training-deep-learning-based-named-entity-recognition-from-scratch-disease-extraction-hackathon/

Submission:

A well-written Python code with sample input, relevant data, test cases and output model. It would be great, if you provide test cases which proves that your system works.

The code should containing following parts:

1. Model generation
2. Model save
3. Model retrieval and prediction or inference function
4. Train | Test split and accuracy

Points:

The points will be counted based on

1. Clarity of solution - https://github.com/zedr/clean-code-python
2. Well-written code - https://www.ibm.com/developerworks/aix/library/au-cleancode/index.html
3. Test-driven development - https://medium.freecodecamp.org/learning-to-test-with-python-997ace2d8abe
4. Markdown document - https://www.markdownguide.org/getting-started/
5. Breakdown of code into modules - https://programminghistorian.org/en/lessons/code-reuse-and-modularity
6. Shell script to run code - https://medium.com/tech-tajawal/writing-shell-scripts-the-beginners-guide-4778e2c4f609