**FinTech Unit 12 NLP Homework Grading Rubric**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Criteria** | **Ratings** | | | |
| **Sentiment Analysis**  • Coin with Highest Mean Positive Score determined.  • Coin with Highest Negative Score determined.  • Coin with Highest Positive Score determined. | **35 Points Mastery** • Completed 3 out of 3 requirements • Code runs without error and produces the assigned results • Code accounts for all possible scenario  • Code is free of bugs | **34 > 28 Points Approaching Mastery** • Completed 2 out of 3 of requirements • Code runs without error • Code produces results as expected 80% of the time | **28 > 23 Points Progressing** • Completed fewer than 1 out of 3 requirements • Code runs without error  • Code produces results, but not necessarily the correct results | **23 > 0 Emerging** • Completed 1 or none out of the 3 requirements • No submission • Code runs with error |
| **Natural Language Processing**  **Tokenize** • NLTK used to lower case words, remove punctuation and remove stopwords.  **N-grams**  • NLTK used to produced n-grams. • Top 10 words listed for each Coin.  **Cloud Generation**  • Words Cloud generated for each Coin summarizing news for each Coin. | **25 Points Mastery** • Completed 4 out of 4 requirements • Code runs without error and produces the assigned results • Code accounts for all possible scenario  • Code is free of bugs | **24 > 18 Points Approaching Mastery** • Completed 3 out of 4 of requirements • Code runs without error • Code produces results as expected 80% of the time | **18 > 13 Points Progressing** • Completed 2 out of 4 requirements • Code runs without error  • Code produces results, but not necessarily the correct results | **13 > 0 Emerging** • Completed 1 or none out of the 4 requirements • No submission • Code runs with error |
| **Named Entity Recognition**  • Model build for both coins - Bitcoin and Ethereum - and Tags Visualized using SpaCy. | **10 Points Mastery** • Completed 2 out of 2 requirements • Code runs without error and produces the assigned results • Code accounts for all possible scenario  • Code is free of bugs | **9 > 7 Points Approaching Mastery** • Completed 2 out of 2 of requirements • Code runs without error • Code produces results as expected 80% of the time | **7 > 3 Points Progressing** • Completed 1 out of 2 requirements • Code runs without error  • Code produces results, but not necessarily the correct results | **3 > 0 Emerging** • Completed 1 or none out of the 2 requirements • No submission • Code runs with error |
| **Coding Conventions/Formatting**  • Appropriate header, name, short description at top of the notebook • Imports are at the top of the file, just after any headers or subheads. • Files read in from relative file path • Functions and variable names are descriptive, lowercase, with words separated by underscores • Clean code, no repetition, maintainable and highly reusable code. • Appropriate code wrapping and cell sizes • Appropriate subheads as needed | **10 Points Mastery** | **9 Points Approaching Mastery** | **8 Points Progressing** | **8 > 0 Emerging** |
| **Deployment/Submission**  • Files submitted in personal repo • Appropriate directory structure with correct files needed to run scripts • Appropriate commit messages • Appropriate README | **10 Points Mastery** | **9 Points Approaching Mastery** | **8 Points Progressing** | **8 > 0 Emerging** |
| **Documentation/Comments**  • Code is well commented with concise, relevant comments | **10 Points Mastery** | **9 Points Approaching Mastery** | **8 Points Progressing** | **8 > 0 Emerging** |