CSIS 3400 Natural Language Processing
Project 01

Due date: February 13 05:00pm

NOTE:

Instructor may ask the project group or individual members questions about the actual work and contribution after submission. The questions may be very specific (E.g., the lines of code written by a member and the explanation of the codes.) It may be done via email or in-person after the class without advanced notice. Failing to answer the questions satisfactorily will result in mark deduction. Please ensure all group members have roughly even contribution in the project, especially the coding part, and have full understanding on the whole project.

Grouping:

You need to form groups of 3/4 to complete the project in this course. Sign your project group list on the Wiki page in Blackboard.

One of the members in your group, the group leader/captain, is responsible for submitting the project. Marks will be deducted if more than one member of your group submit the same/different project.

Project Description:

In this project you will be working on an unclean dataset that contains information of statements about climate. Each statement may or may not be relevant to a climate action. You task is to predict the relevancy.

Project Submission Requirements

Proj01.ipynb (or a zip file if you have any extra files): In this Jupyter notebook, you have to utilize different cells (code/markdown) to clearly indicate and explain every step. Your Jupyter notebook should include all the markdown texts signifying the steps with correct heading, python code, comments/analysis, and visualizations as stated in the following instruction. Note: You need to create the appropriate markdown headings for each section mentioned below. Codes should have some short comment describing the statement. Adding a markdown cell containing text before specific actions performed is appreciated.
(*Note: Restart the kernel and re-run all cells of the notebook before submission. Substantial marks will be deducted for cell errors.*)

Files Required:

climate_train.csv: This file contains the training set. It has three fields: Row Number, Text and Label. Row Number is the unique incremental ID; Text is the content of the statement; Label is the relevance to climate action (1 is yes, 0 is no).

climate_test.csv: This is the text set. It has the same format as the training set. This label information of the test set must not be used in the training of the model, ZERO marks will be given otherwise.

Part A. Planning

1. **Title, Name and References**
   Create the very first markdown cell in the notebook and include the full name and student ID of your group.
   Create another markdown cell titled References. Add information about any references you used to help complete the project

2. **Planning**
   Examine the dataset carefully. Create another markdown cell. In this cell, describe your plan about how to carry out the NLP pipelines to create classification models. Use appropriate markdown if necessary to have a better formatting and illustrations.

Part B. Basic Model

In the following parts, you need to write python codes, with appropriate comments or markdown cells to explain your work. Lacking explanations will result in mark deduction.

1. **Library import and data loading**
   Import all the required important libraries for Parts A, B and C, and load the provided dataset.

2. **Data exploration, pre-processing and cleansing**
   In this step, you should write codes
   i. to explore the dataset
   ii. convert labels to number if necessary
   iii. to pre-process the dataset by removing digits, stopwords

3. **Feature generation**
   i. In this step, use TF-IDF to represent each data in both training set and test set

4. **Model building**
   i. Use Naïve Bayes (Multinominal) Model to train a model.

5. **Testing and evaluation**
    i. Test the model in test set.
    ii. Use suitable metrics to evaluate the performance of the model

6. **Post analysis**
    i. Research on how to identify the most impactful features (tokens) to classify climate action statement.

Part C. Comparison with other models

1. **Classifier comparison**
    i. Using the same features you found in Step 3 of Part A, training different classifiers: Logistic Regression, Linear SVM, K-Nearest-Neighbour (You need to research on how to do this).
    ii. Evaluate these models in test set
    iii. Compare the performance and make a conclusions

2. **Feature generation comparison**
    i. Identify the best classifier in Part B. Step 1.
    ii. Using different feature generations methods: BoW model, Pre-trained word embedding, followed by the training of this classifier.
    iii. Evaluate these feature generation methods in test set
    iv. Compare the performance and make a conclusions

Part D. Competition

In this part, your group competes with other groups in performance. In the same notebook, you need to write python codes after Part B, with appropriate comments or markdown cells to explain your work. Lacking explanations will result in mark deduction.

1. **Dataset**
    i. You must only use the climate_train.csv in the model training.
    ii. Evaluation must be done on climate_test.csv.

2. **Evaluation**
    i. Micro F1-measure will be used as the evaluation metric to determine which group is the winner.

3. **Method**
    i. You are free to use any data preprocessing, feature generation, classifier in this part.
    ii. Only one final pipeline should be included in the submission. For example, your group may try different methods in another Jupyter notebook or python program. However, your group should only report the best one.

4. Restriction

i. The execution time of this part needs to be within 1 minute. Your group will be disqualified if your codes cannot finish within 1 minute.

Member Contribution

In addition to the proposal, each group needs to submit a peer evaluation matrix. Each cell should be a number between 1 and 4, which reflects how a member thinks the contribution by another member. The evaluation is opened to open to all members of your group (i.e., Every one can see how others grade on you), so that each member knows how to enhance their contribution in the project.

(Hint: You may refer to this link to see how to create a table in a Jupyter Markdown cell.)

Evaluator \ Evaluatee	Member 1	Member 2	Member 3	Member 4
Member 1				
Member 2				
Member 3				
Member 4				

Here is the rubric on how to evaluate your team members:

1 Point: No or very little contribution to the project; cannot deliver artifacts or largely miss the agreed deadline; showing no or very little passion in development.

2 Points: Little contribution to the project with no negative effect to the group; sometimes cannot deliver artifacts or miss the agreed deadline; mainly follow other members' idea and instructions.

3 Points: Fairly large and positive contribution to the project; can handle most of the assigned tasks and deliver artifacts on time;

4 Points: Large and positive contribution to the project; can help members to tackle problems; pro-active and passionate in the development.

D. Project Grading Criteria

The project will be graded on a scale of 20 points.

Criteria		Grading
Project submitted, named properly with all files included in their corresponding folders to Blackboard.		1
Part	**Detail**	
A	Planning for the analysis	2
B.2	Data exploration, pre-processing and cleansing	1
B.3	Feature generation	1

B.4	Model building	1
B.5	Testing and evaluation	1
B.6	Post analysis	1
B	Overall description or explanation	2
C.1	Classifier comparison	3
C.2	Feature generation comparison	2
C	Overall description or explanation	2
D	One best group: 3 Other groups with reasonable effort: 2 Others groups with simple approach: 1 No submission/problematic implementation/invalid approach: 0	3
	Total:	**20**