**Project COMP 262: Natural Language processing and recommender systems**

**Introduction**

Throughout this two-phase project assignment, each team needs to construct “a sentiment analysis model for products based on customers’ textual reviews,” using both a Lexicon approach and a machine learning approach.

First phase involves uploading the data, cleaning it up, pre-processing the data in order to create a textual representation, and finally, building and testing the Lexicon classifier. In the second phase, the team needs to construct the same procedure using a Machine learning approach and compare the results of each approach. Lastly, a study of how to utilize the same review data to construct a recommender system is required.

The project would be governed by a set of deliverables per phase and there are certain check points with the professor, as illustrated in the project timetable key-milestones section.

deliverables will be evaluated based on rubric illustrated in the Rubric section.

A project plan should be built by the team and updated on a weekly basis, in addition, a simple log of all team meetings should be maintained. Both should be submitted with final project documentation and code as appendices to the project report.

At the end of each phase, the team needs to present their work to the class.

Grading is both at the team level and at the individual level.

**Data sets**

We will use the Amazon product review datasets available at: <https://nijianmo.github.io/amazon/index.html> previously <http://jmcauley.ucsd.edu/data/amazon/> we will use the small review subsets referenced as the **k-core** for phase #1 and the **full set** for phase #2.

Each team will tackle any one of the dataset, as follows:

#1: Amazon Fashion

#2: All Beauty

#3: Appliences

#4: Industrial and Scientific

#5: Gift cards

#6: Software

Please reference the publishers of these datasets, in your report.

**Deliverables:**

Phase #1

1. Dataset data exploration: List the main finding of the dataset. Be thorough and creative. For example, look at:
   1. Counts, averages
   2. Distribution of the number of reviews across products
   3. Distribution of the number of reviews per product
   4. Distribution of reviews per user
   5. Review lengths and outliers
   6. Analyze lengths
   7. Check for duplicates
2. Text basic pre-processing:
   1. Label your data based on the value of “rating of the product” i.e. as follows:
      1. Ratings 4,5: Positive
      2. Rating 3: Neutral
      3. Ratings 1,2: Negative
   2. Chose the appropriate columns for your sentiment analyzer. (Give this some thought) and mention in your report why you chose each column.
   3. Check for outliers
3. Study the below three Lexicons packages and choose two for model building justify why you chose them:
   * 1. Valence Aware Dictionary and Sentiment Reasoner (VADR) you can find out more information here: <https://github.com/cjhutto/vaderSentiment>
     2. TextBlob you can find out more information here: <https://textblob.readthedocs.io/en/dev/quickstart.html>
     3. SENTIWORDNET you can find more information here: <http://nmis.isti.cnr.it/sebastiani/Publications/LREC10.pdf>
4. Pre-process your text as needed, justify each pre-processing step for each model you chose. (Note: take into account the findings of step #3 above)
5. Randomly select 1000 reviews from your dataset.
6. Modeling (Sentiment Analysis) Lexicon approach:
   1. Build two sentiment analysis models using the labeled pre-processed data for both the lexicons packages the team selected in step #3 above.
7. Validate the results of both models and provide a comparision table.
8. Presentation: Check project presentation requirements.
9. Project report: Check project report requirements/ phase #1 (Make sure you refernce your work)
10. Submit documented code.

Phase #2

1. Modeling (Sentiment Analysis) Machine Learning approach:
   1. Select a subset of the original data minimum 2000 reviews, check point 14 below as you select the subset.
   2. Carry out data exploration on the subset and pre-processing and justify each step of preprocessing.
   3. Represent your text using one of the text represtations discussed in the course, make sure to note in your report why you chose that representation.
   4. Split the data into 70% for training and 30% for testing,—Use stratified splitting based on the rating value field.
   5. Build two sentiment analysis models using 70% of the data. Choose two of the following Machine Learning algorithms to build and fine tune your models:
      1. Logistic Regression
      2. SVM
      3. Naïve Bayes
      4. Gradient Boosting
      5. MLP
2. Note the results of the training process in your report.
3. Testing: Test out the two models using the 30% test data note the accuracy, precision, recall, confusion matrix and F1 score in your report.
4. Design an experiment to compare the test results of the Lexicon model versus the two machine learning models:
   1. Prepare the data: Here you will need to create a situation where you compare apples to apples, so whatever you used in the Lexicon should be the test data for your machine learing model, this step requires good design.
   2. Run both models on the same data and compare the results using appropriate matrics.
5. Review the attached paper “Recommender systems based on user reviews: the state of the art”, can also be accessed at the centennial library. Examining the options presented in the paper carryout the following:
   1. Explain how you can enhance the rating values of your data using the review data.
   2. Choose one of the suggested options, provide diagrams and pseudo-code.
   3. Implement the suggestion on your dataset.(Code needs to be provided) record the results in your report.
6. Select 10 reveiws with lengths more than 100 words, using a LLM model summarize the results into a 50 word. Note the results of the first two into your report. (use Hugging Face models and host them locally)
7. Select one review that carries a question nature, using a LLM model to automatically create a response as if it were from a service representatitive. Note the results in your report. (use Hugging Face models and host them locally)

**Timetable – key milestones**

|  |  |
| --- | --- |
| Milestone | Week # |
| Project teams assembled, and datasets assigned | 3 |
| Check point # 1 “Data exploration & pre-processing” progress “phase #1 upto step #4” | 6 |
| Presentation & submission phase #1 | 7 |
| Check point #2 progress on modelling | 12 |
| Presentation & submission phase #2 | 14 |

**Peer-evaluation**

With every phase submission, each team member should fill in the peer evaluation form and submit it to the assessment box named "Peer evaluation Phase X", where X is 1 or 2. This form is confidential, and only the professor will access it.

In summary, this form is to express what each team member has worked on and how the team member views the contribution of the rest of the team members. If all team members have contributed equally, then give all a rate of 100%, if a team member did not contribute then give a 0%, finally, if a team member contributed but not to the level of the team agreement, then a score between 1% to 99%.

Any team member who does not submit the form before the deadline will lose 10% of the 100%.

Any team member who fills the form incorrectly will lose 10% of the 100%.

**Project Report requirements:**

1. Cover page
2. Table of contents
3. Detailed results of dataset exploration & conclusions for each phase.
4. Dataset pre-processing steps with explanation and justification of choices.
5. Text representation model with explanation and justification (Only in phase #2)
6. **Models**; per model clarify:
   1. Assumptions/Heuristics/algorithms/packages used
   2. Explain each model, how it works
   3. Fine tunning steps
7. Training results summary, only phase #2.
8. Testing results summary.
9. Make sure all that has been requested in the steps of phase #1 and phase #2 are presented in a professional way in your report.
10. Final conclusion.
11. Assumptions.
12. References.
13. Appendix 1: Project plan.
14. Appendix 2: Meeting register, simple table showing date and time of each meeting, who attended, subjects discussed and assignments.

Note: phase #2 deliverables are appended to the phase #1 report (i.e. Only one report for the whole project).

**Presentations requirements:**

1. All team members need to participate equally. (any team member who does not atten will lose presentation mark)
2. Present working code.
3. Present power point summarizing key points related to the project.

**Rubric**

|  |  |
| --- | --- |
| Phase #1 |  |
| Data set exploration | 20% |
| Text pre-processing | 20% |
| Modelling | 5% |
| Testing | 5% |
| Project report | 20% |
| Presentation | 30% |
|  |  |
|  | 100% |

|  |  |
| --- | --- |
| Phase #2 |  |
| Data set exploration | 10% |
| Text pre-processing | 10% |
| Training and modelling | 10% |
| Testing | 10% |
| Review score enhacement | 15% |
| LLM | 10% |
| Project report | 15% |
| Presentation | 20% |
|  |  |
|  | 100% |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Evaluation criteria | **Not acceptable** | **Below**  **Average** | **Average** | **Competent** | **Excellent** |
|  | **0% - 24%** | **25%-49%** | **50-69%** | **70%-83%** | **84%-100%** |
| Dataset data exploration  Phase #1  Phase #2 | Data exploration completely missing or what is submitted is below 30% with no relationship analysis. | Only 50%-60% of dataset attributes have been explored or exploration not complete on # of missing values, only a few relationships are captured, minimum visualizations. | Only 60%-70% of dataset attributes have been explored or exploration not complete on # of missing values not all relationships are captured. | Most dataset attributes columns have been explored and a complete description of each attribute value meaning has been reported in addition to exploring some relationships between attributes and presented a few visualizations. | All dataset attributes columns have been explored and a complete description of each attribute value/meaning/distribution has been reported in addition to exploring all relationships between attributes supported by a complete set of visualizations. |
| Text basic pre-processing  Phase #1  Phase #2 | Data not pre-processed  No comments explaining code. | Some major errors in the data model.  Issues with sampling labelling. Outliers not addressed. Normalization not implemented as needed.  Minor comments are implemented. | Some errors in the data model.  Issues with sampling labelling. Outliers not addressed. Normalization not implemented as needed.  Some code is correctly commented. | Correct sampling, labeling and splitting of data.  Data outliers are cleaned up as needed, normalization/standardization is implemented as needed. Appropriate text pre-processing is implemented  Selection and build of the data model not justified.  selected attributes.  Majority of code is correctly commented. | Correct sampling, labeling and splitting of data.  Data outliers are cleaned up as needed, normalization/standardization is implemented as needed. Appropriate text pre-processing is implemented.  Logical selection/merging and justification of selected attributes.  All code is correctly commented. |
| Text representation  Phase #2 | Missed to represent the text completely. | Shows some thinking and reasoning but text representation not suitable for the nature data/task. | Text representation model can work but not the best for the nature of the data/task. | Suitable text representation without justification clearly explained | Suitable text representation with justification clearly explained. |
| Modelling  Phase #1  Phase #2 | Majority of Models are not implemented. | Some models are implemented with errors. | Majority of models are implemented but not with optimal hyperparameters. And minium justification provided. | All models are implemented correctly but not with optimal hyperparameters. | All models are implemented correctly and an explanation justified. |
| Testing  Phase #1  Phase #2 | No model evaluations  conducted | Some metrics are generated for each model, with no comparisons/conclusions presented. | Some metrics are generated for each model, with minimum comparisons presented with partial conclusions. | All metrics are generated for each model and a comprehensive comparison presented with partial conclusions. | All metrics are generated for each model and a comprehensive comparison presented with clear conclusions. |
| Enhace reviews  Phase #2 | Nothing or pratials that make no sense in relation to the request. | Wrong strategy selected and justified, pseudo code not noted in report. Code implemented with issues and results not explained clearly. | Correct strategy selected and justified, pseudo code not noted in report. Code implemented with issues and results not explained clearly. | Correct strategy selected and justified, pseudo code noted in report. Code implemented correctly but results not explained clearly. | Correct strategy selected and justified, pseudo code noted in report. Code implemented correctly and results explained clearly. |
| LLM Phase #2 | Nothing or pratials that make no sense in relation to the request | No thought for modell selection for the task, not much thought on selection of hyper parameters with minimum justification provided. Prompt engineering best practices not taken into account. Results not presented | No thought for modell selection for the task, not much thought on selection of hyper parameters with minimum justification provided. Prompt engineering best practices not taken into account. Results presented partially | Best models chosen for the task, not much thought on selection of hyper parameters with minimum justification provided. Prompt engineering best practices taken into account. Results presented partially. | Best models chosen for the task, reasonable selection of hyper parameters with justification provided. Prompt engineering best practices taken into account. Results presented professionally. |
| Project report  Phase #1  Phase #2 | Writing lacks logical organization. It shows no coherence, and ideas lack unity.  Missing most conclusions or assumptions or references  Serious errors. No transitions.  Format is very messy. | Writing lacks logical organization. It shows some coherence but ideas lack unity. Serious errors.  Missing many conclusions or assumptions or references  Format needs attention, some major errors. | Writing is coherent and logically organized. Some points remain misplaced.  Missing many conclusions or assumptions or references  Format is neat but has some assembly errors. | Writing is coherent and logically organized, with transitions used between ideas and paragraphs to create coherence. The overall unity of ideas is present.  Missing some conclusions or assumptions or references. Format is neat and correctly assembled. | Writing shows a high degree of attention to logic and reasoning of all points. Unity clearly leads the reader to the conclusion.  Covers all deliverable results. Covers all assumptions and conclusions.  Includes references.  Format is neat and correctly assembled with a professional look. |
| Presentations  Phase #1  Phase #2 | Very weak, no mention of the code changes. Execution of code not demonstrated. Some team members do not participate. | Some parts of the code changes are presented.  Execution of code partially demonstrated. Some team members do not participate. | All code presented but without explaining why. Some parts of the code are not working and have errors. Some team members do not participate. | A comprehensive view of all code demonstrated presented with an explanation, exceeding the time limit. Working code demonstrated. All team members participated but without equal participation. Some team members are not confident of their input. | A comprehensive view of all code demonstrated in working condition with explanation, within the time limit. All team members participate equally and are confident in their responses. |