ASSIGNMENT 6
ANALYZING A BERT-BASED SENTIMENT CLASSIFIER

CS 478 NATURAL LANGUAGE PROCESSING (FALL 2024)
https://nlp.cs.gmu.edu/course/cs478-fall24/

OUT: Nov 21, 2024

DUE: Dec 11, 2024

TOTAL CREDITS: 100 Points

Your name: Arron Birhanu

Your GID: G01315277

Shareable link to your notebook: Link to my notebook
(Make sure to submit your notebook to Blackboard as well)

Academic Honesty: Please note that students should complete the assignment independently. While you may discuss the assignment with other students, **all work you submit must be your own!** In addition, if you use any external resources for the assignment, you must include references to these resources at the end of this PDF. However, **you are NOT allowed to use AI assistants such as ChatGPT and Claude to complete your assignment, although using them for generic concept understanding is okay.**

Overview and Submission Guideline: In this project, you will perform an analysis on a BERT-based sentiment classifier ("textattack/bert-base-uncased-yelp-polarity") on our sentiment classification dataset using the well-known CheckList tool (https://github.com/marcotcr/checklist). If you haven't, you are suggested to read the original paper publication of this tool:

Ribeiro, Marco Tulio, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. "Beyond Accuracy: Behavioral Testing of NLP Models with CheckList." In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4902-4912. 2020.

You will be using exactly the same dataset as in Assignments 3 and 4. Like before, complete the notebook, save your edits *(make sure to save the cell outputs as well)*, add a shareable link to your notebook on this PDF, and submit your notebook to Blackboard as a backup.

How to fill out this PDF? In most answer blanks, you only need to provide the execution output of your code implementation. There are only a few blanks that ask you to provide your code implementation for ease of grading. When you complete the PDF, submit it to Gradescope.

What and Where to Submit? To summarize, you will submit two items from this assignment:

- A completed PDF compiled from this LaTex source file to Gradescope. A link to your notebook should also be included in this PDF.
- Your completed notebook to Blackboard.

Part 1: Invariance Test (INV) using CheckList (70 Points)

In this part, you will use CheckList to implement two INV tests. INV is a test type where we aim to perturb the original input sentence *without changing its ground-truth label*. By comparing the classifier's performance on the original and the perturbed test examples, we will gain insights into the robustness of the classifier to the small perturbations.

Part 1.1: INV - Punctuations (35 points)

In the first INV test, we will focus on "punctuation perturbation". For example, for the sentence "*John is a very smart person, he lives in Ireland.*", we can remove its period, making the sentence into "*John is a very smart person, he lives in Ireland*", but this perturbation should not change a classifier's prediction on this sentence (unless the task depends on the punctuation, e.g., counting the number of punctuations in a sentence).

In CheckList, the punctuation perturbation is implemented to (1) remove the last punctuation, if any, and (2) add a period at the end, if the last word in the original sentence does not end by a period. For the sentiment classification dataset we use, all sentences have been tokenized, and all of them end with a tokenized punctuation. As a result, when you use CheckList's punctuation perturbation function, you should see it produce three sentences as output: (a) the original sentence, (b) the original sentence with the last tokenized punctuation truncated, and (c) the original sentence with a period attached to the last word.

Q1: Follow the CheckList notebook and implement the punctuation perturbation function (15 points):

Your Code Solution

```
# TODO: complete the assignment for creating
# a set of dev-set examples with perturbed punctuations
# using the Perturb class
ret1 = Perturb.perturb(pdata, Perturb.punctuation)
```

Show the perturbation results for the first three sentences on the dev set (i.e., the output of "ret1.data[:3]"):

Your Answer

```
[["it 's a lovely film with lovely performances by buy and accorsi .",
  "it 's a lovely film with lovely performances by buy and accorsi",
  "it 's a lovely film with lovely performances by buy and accorsi."],
 ["and if you 're not nearly moved to tears by a couple of scenes ,
    you 've got ice water in your veins .",
  "and if you 're not nearly moved to tears by a couple of scenes ,
    you 've got ice water in your veins",
  "and if you 're not nearly moved to tears by a couple of scenes ,
    you 've got ice water in your veins."],
 ['a warm , funny , engaging film .',
  'a warm , funny , engaging film',
  'a warm , funny , engaging film.']]
```

Also show the two statistical numbers (i.e., how many predictions were flipped in total, and how many *initially correct* predictions were flipped):

Your Answer

```
26 examples got their label flipped!
15 examples got their *correct* label flipped!
```

Q2: Read through the printed examples (whose labels were flipped caused by the punctuation perturbation) carefully and include two most interesting ones in the assignment PDF. (10 points)

Example 1

```
Original Sentence: without non-stop techno or the existential
    overtones of a kieslowski morality tale , maelstr m is just
    another winter sleepers .
Perturbed Sentence: without non-stop techno or the existential
    overtones of a kieslowski morality tale , maelstr m is just
    another winter sleepers
Correct Label: 0 (Negative)
Predicted Label for Original Sentence: 1 (Positive)
Predicted Label for the Perturbed Sentence: 0 (Negative)
Your Comment: In this example, the punctuation perturbation (removal
    of the period) caused a label flip. The original sentence
    predicted as positive, likely due to the enthusiastic description,
     while the perturbed sentence, lacking the period, was interpreted
     negatively. This is an interesting case where punctuation removal
     seems to affect the sentiment prediction, suggesting that the
    model might be influenced by sentence-ending punctuation.
```

Example 2

```
Original Sentence:it 's like every bad idea that 's ever gone into an after-school special compiled in one place , minus those daytime programs ' slickness and sophistication -lrb- and who knew they even had any ? -rrb- .
Perturbed Sentence:it 's like every bad idea that 's ever gone into an after-school special compiled in one place , minus those daytime programs ' slickness and sophistication -lrb- and who knew they even had any ? -rrb-
Correct Label: 0
Predicted Label for Original Sentence: 1
Predicted Label for the Perturbed Sentence: 0
Your Comment: The model predicted the original sentence as positive (label = 1), but after perturbation (with the punctuation removed), it correctly predicted the perturbed sentence as negative (label = 0). This seems to be a common trend.
```

Q3: Discussion. Changing or removing the punctuation of a sentence may or may not change the sentence's original semantic meaning. Use your common sense to judge the model's predictions, and discuss: (1) Should all labels NOT be flipped? In other words, do you see cases where the prediction labels are hard to decide or should actually be flipped when the punctuation is changed? (2) Overall, do you think your model is robust to punctuation perturbation? (10 points)

Discussion

```
Some labels should not be flipped, particularly when punctuation changes do not substantially alter the meaning or sentiment. However, some sentences, especially those with contradictions or sarcasm, can lead to ambiguous interpretations where the label might be flipped. The model's response to punctuation changes can be inconsistent, and such cases would require manual review. The model is not fully robust to punctuation perturbation. Although it may perform well in cases where punctuation does not significantly affect sentiment, it struggles in cases where punctuation plays a crucial role in conveying the sentiment, especially with sarcasm, rhetorical questions, or complex sentence structures.
```

Part 1.2: INV - Typos (35 points)

In the second INV test, we will focus on "typos perturbation", i.e., intentionally replacing words with typos in the test examples and observing if the classifier can preserve its prediction. By default, the typos perturbation function in CheckList should return one perturbed sentence each time.

Q4: Follow the CheckList notebook and implement the typos perturbation function (15 points):

Your Code Solution

```
# TODO: complete the assignment for creating
# a set of dev-set examples with perturbed typos
# using the Perturb class
ret2 = Perturb.perturb(data, Perturb.add_typos)
```

Show the perturbation results for the first three sentences on the dev set (i.e., the output of "ret2.data[:3]"):

Your Answer

```
[["it 's a lovely film with lovely performances by buy and accorsi .",
  "it 's a lovely film wiht lovely performances by buy and accorsi ."
    ],
 ["and if you 're not nearly moved to tears by a couple of scenes ,
    you 've got ice water in your veins .",
  "and if you 're not nearly moved to tears by a couple of scene s,
    you 've got ice water in your veins ."],
 ['a warm , funny , engaging film .', 'a warm , funny , negaging film
    .']]
```

Also show the two statistical numbers (i.e., how many predictions were flipped in total, and how many *initially correct* predictions were flipped):

Your Answer

```
94 examples got their label flipped!
62 examples got their *correct* label flipped!
```

Q5: Read through the printed examples (whose labels were flipped caused by the typos perturbation) carefully and include two most interesting ones in the assignment PDF. (10 points)

Example 1

```
Original Sentence: it 's not that kung pow is n't funny some of the
    time -- it just is n't any funnier than bad martial arts movies
    are all by themselves , without all oedekerk 's impish
    augmentation .
Perturbed Sentence: it 's not that kung pow is n't funny some of the
    time -- it jus tis n't any funnier than bad martial arts movies
    are all by themselves , without all oedekerk 's impish
    augmentation .

Correct Label: 0
Predicted Label for Original Sentence: 0
Predicted Label for the Perturbed Sentence: 1
Your Comment: The original sentence conveys a negative sentiment, as
    it critiques the humor of the movie and suggests it does not stand
     out. The slight perturbation ("jus tis") does not alter the
    meaning or sentiment of the sentence. However, the model flipped
    the prediction to positive (1) for the perturbed version, which is
     incorrect.
```

Example 2

```
Original Sentence: no telegraphing is too obvious or simplistic for
    this movie .
Perturbed Sentence: no telegraphingi s too obvious or simplistic for
    this movie .
Correct Label: 0
Predicted Label for Original Sentence: 0
Predicted Label for the Perturbed Sentence: 1
Your Comment: The original sentence expresses a negative sentiment by
    criticizing the movie's lack of subtlety. The perturbation ("
    telegraphingi s") is a minor typo that does not change the
    sentence's meaning or sentiment. Despite this, the model
    incorrectly flipped the prediction to positive (1) for the
    perturbed version. This misclassification highlights the models
     lack of robustness to small textual perturbations that have no
    bearing on the semantic content.
```

Q6: Discussion. Overall, do you think your model is robust to typos? Discuss any other findings, e.g., for cases where the model predictions got flipped, do the typos exist more commonly in nouns or verbs, or in words with other part-of-speech properties? (10 points)

Discussion

```
The model requires improved robustness to textual noise, particularly
    typos, to enhance its reliability. Strategies like adversarial
    training with typo-augmented datasets or better semantic modeling
    could help mitigate these issues. Additionally, focusing on
    strengthening the model's ability to infer meaning from context,
    rather than relying on surface patterns, could improve performance
    .
```

Part 2: Explore A Different Test or Analysis (30 points)

In this section, you will then explore any one test or analysis you find interesting and apply it to the same BERT sentiment classifier. For example, here are a few analyses you can consider:

- Trying another INV perturbation category (e.g., named entity change) or a different CheckList test type (MFT or DIR);
- Identifying potential ethical problems (e.g., gender or racial bias) by testing the sentiment classifier on test exmaples created by yourself;
- Testing the multilingual capabilities of the sentiment classifier (for this you should switch to a multilingual BERT sentiment classifier from the model hub);
- Visualizing the BERT attention of this classifier (using BertViz or other tools): does the model attend to the right contents when making a prediction (i.e., "right for the right reason")?

Please include your code implementation in the Notebook, and describe your procedure, result, and findings below. **You can feel free to extend or remove the answer blocks per your need, as long as all the three questions are answered.**

Q7: Describe your test or analysis. What do you test or analyze, why does it matter, and how do you do it? (15 points)

Your Description

```
What? We analyzed the robustness of a sentiment classification model
    to perturbations involving changes to names and genders in neutral
     sentences. Specifically, we replaced names with various
    alternatives (e.g., swapping "John" with "Robert" or "Emma") and
    observed whether the predicted sentiment labels were affected.

Why? Models can show bias or instability when exposed to changes in
    specific parts of a sentence, such as names or pronouns. If the
    model flips its sentiment predictions solely based on a name
    change without any semantic difference, it may indicate a lack of
    robustness or inherent bias.

How:
Dataset: We used three neutral sentences with varying subjects and
    professions.
Perturbation: We applied a name and gender perturbation using the
    Perturb.change_names method to generate multiple variations of
    each sentence.
Prediction Comparison: Sentiment predictions for the original and
    perturbed sentences were compared to check for any label flips.
```

Q8: What results did you observe? Make sure to show a few concrete examples! (10 points)

Your Description

```
We observed two components to this experiment.
No Label Flips: Across all perturbed versions, the predicted sentiment labels remained consistent with the original labels ("neutral").
Model Robustness: The model demonstrated stability in handling name and gender changes without altering its sentiment predictions.

Examples-

Example 1:
Original: "Amina is an artist, and she feels good about her progress."
Perturbed Versions:
"Emma is an artist, and she feels good about her progress."
"Jessica is an artist, and she feels good about her progress."
Prediction: The label remained "neutral" for all versions.

Example 2:
Original: "John is a software engineer, and this is a fine day."
Perturbed Versions:
"Robert is a software engineer, and this is a fine day."
"Luis is a software engineer, and this is a fine day."
Prediction: The label remained "neutral" for all versions.
```

Q9: Discuss the results and your findings. (5 points)

Discussion

```
Pros:
Stability: The model demonstrated strong robustness by maintaining consistent predictions despite name and gender changes.
Fairness: The models neutrality to names suggests an absence of bias towards specific names or genders in the given context.

Cons:
Scope of Testing: The test was limited to neutral sentences. Further evaluation is needed on sentences with positive or negative sentiment to check for potential sensitivity to name or gender changes in sentiment-laden contexts.
Potential Biases: While no bias was detected in this test, real-world examples with culturally or emotionally charged names might produce different results.

These results indicate that the model likely prioritizes semantic content over lackluster changes like names. However, continuous testing across diverse scenarios leads to robustness and fairness.
```

Post-Assignment Questions (Required)

Q-P1: Did you use any resources (e.g., online tutorials, blog posts, etc.) while completing this assignment? If you did, please list them below:

Your Answer

I did look at the classes Perturb and Editor to understand the parameters that create BERT classifier benchmarks from our dev.txt file.

Q-P2: Did you use AI assistants (e.g., ChatGPT) while completing this assignment? If you did, please describe how it was used. Note that you are NOT allowed to use the AI assistant's code output as yours, but it is okay to use it to assist you in generic concept understanding.

Your Answer

None at all.