515 HOMEWORK 5:
ONLINE REVIEW CLASSIFICATION

Your assignment is to create a tool that trains several machine learning models to perform the task of classifying online reviews. Some of these online reviews refer to hazardous products, so these machine learning models will help to identify the most serious product complaints.

The dataset is available at https://dgoldberg.sdsu.edu/515/appliance_reviews.json and contains approximately 1,000 reviews, approximately half of which refer to safety hazards. The data is formatted as a JSON array. An example of the formatting is below:


```
[
    {"Review": "I was really surprised how quickly it was
    shipped. Ordered it on Sunday and it was delivered the
    following Friday. I couldn't be happier with my steamer.
    Works perfectly for any type of vegetable or rice. Easy to
    use and clean up after.", "Stars": 5, "Safety hazard": 0},

    ...

    {"Review": "On the 2nd time that I used it, one of the
    PLASTIC clips holding the upper plate broke and the VERY
    HOT plate came loose falling.  Fortunately, I was standing
    there at the time it fell.  This was a very dangerous
    situation and could have ruined my kitchen counter and
    caused a fire.  Heating element behind the plate was
    exposed.  Using a cheap plastic clip to hold a heavy
    cooking plate in place is just plain stupid engineering.  I
    expected better from a Cuisinart product.", "Stars": 1,
    "Safety hazard": 1}
                                                            ]
```


The purpose of the machine learning models is to predict the “Safety hazard” field, which is already formatted as a 0 or 1. A value of 1 indicates that the review refers to a safety hazard; a value of 0 indicates that the review does not refer to a safety hazard. However, to transform the reviews into a format usable by the machine learning models, use the following four variables:

- Length: the number of characters in the review.
- Stars: the star rating assigned to the review.
- Polarity: the positive or negative emotive content in the review.
- Subjectivity: the opinionated or factual content in the review.

Train decision tree, k-nearest neighbors, and neural network machine learning models. You may choose an appropriate training/test split. Report the accuracy values from all three machine learning models and save a joblib file from the most accurate model. The printout of your code may be brief. For example:

```
Decision tree accuracy: 0.81
k-nearest neighbors accuracy: 0.74
Neural network accuracy: 0.86
Neural network model performed best; saved to model.joblib.
```

Some considerations as you write your program:

- Consider the possibility that, when loading the dataset, some connection issue occurs (that is, a status code other than 200). Ensure that your code handles this case and provides the user with a helpful printout if it does occur.

- It is possible that your accuracy numbers may differ slightly from the values above due to differences in your training/test split; if your values are generally close, though, then this is not a problem.

- When training your neural network model, you may see a warning message "Maximum iterations reached and the optimization hasn't converged yet." This message means that the neural network model would have preferred to work with a larger dataset, but it does not actually cause an error. However, *optionally*, if you would like to turn this message off, then you can do so using the following:

  ```
  import warnings
  warnings.filterwarnings("ignore")
  ```

- Ensure that your output is crisp, professional, and well-formatted. For example, ensure that you have used spaces appropriately and checked your spelling.

- Adding comments in your code is encouraged. At minimum, please use a comment at the start of your code to describe its basic functionality. In addition, for any functions or classes in your code, write appropriate docstrings. Ensure that your code would be as understandable as possible for a programmer working with your code for the first time.