

Assignment 11

Applied Machine Learning

1. [50 pts] In this assignment, we will use a priori analysis to find phrases, or interesting word patterns, in a novel.

Note that you are free to use any a priori analytics and algorithm library in this assignment.

Use the `nltk` library corpus `gutenberg` API and load the novel 'carroll-alice.txt', which is *Alice in Wonderland* by Lewis Carroll (although his real name was Charles Dodgson). There are 1703 sentences in the novel—which can be represented as 1703 transactions. Use any means you like to parse/extract words and save in a .csv format to be read by Weka framework, similar to the a priori Analysis module. (Hint: Feel free to use `mlxtend` library instead of Weka.)

Hint: Removing stop words and symbols using regular expressions can be helpful:

```
from nltk.corpus import gutenberg, stopwords
Stop_words = stopwords.words('english')
Sentences = gutenberg.sents('carroll-alice.txt')
TermsSentences = []
for terms in Sentences:
    terms = [w for w in terms if w not in Stop_words]
    terms = [w for w in terms if re.search(r'^[a-zA-Z]{2}', w) is not None]
```

If you chose to Weka, use FPGrowth and start with default parameters. Reduce `lowerBoundMinSupport` to reach to a sweet point for the support and avoid exploding the number of rules generated.

Report interesting patterns.

(Example: Some of the frequently occurring phrases are “Mock Turtle”, “White Rabbit”, etc.)

2. [50 pts] In the lecture module, the class `NeuralNetMLP` implements a neural network with a single hidden layer. Make the necessary modifications to upgrade it to a 2-hidden layer neural network. Run it on the MNIST dataset and report its performance.

(Hint: Raschka, Chapter 12)

