

Discussion (Apr 8) Project 1 Classifier agent.

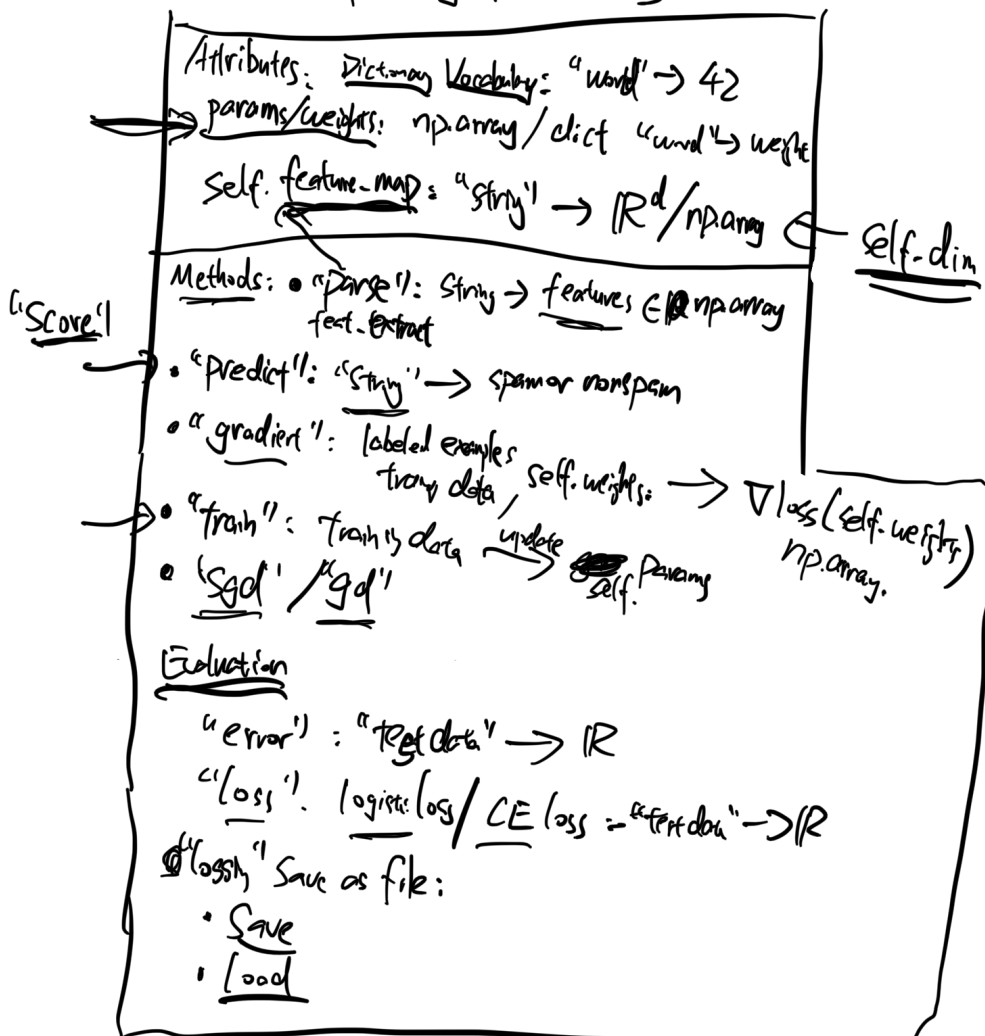
Plan:

- ① Designing a "classifier" class
- ② "Bag of Word" Feature
- ③ "tf-idf" feature
- ④ "Jax" for autograd

The "Zen" of
python programming

"Vectorization"

"Classifier_agent" class



"feature_extractor" class

BoW features:

70k+
el-items in dictionary

Vocab: ["Dear", "Prof", "this", "is", "...", "i"]

Sentence: "this is great")

BoW = $\begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \leftarrow 75\text{th}$

Sparse CSS array
(index, value)
{
()

$$TF \cdot IDF = TF \times IDF$$

Term frequency, Inverse Document Freq.

$$TF = \frac{BoW}{\sum_{i=1}^d BoW(i)}$$

Exp: $\begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}$

$$IDF(i) = \log \frac{\text{Total \# of Documents}}{\text{\# of times Word 'i' appeared in a Document}}$$