Lab 2 实验报告 / 人工智能基础

马天开 / PB2100030

2024 年 6 月 22 日

Lab 2.1 传统机器学习

Lab 2.1.1 决策树

1. 计算熵、信息增益:

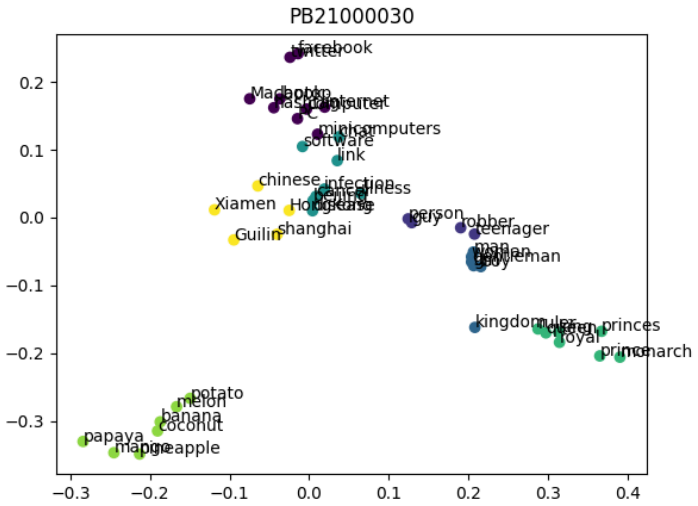
```python
def entropy(self, y: np.ndarray) -> float:
    _, counts = np.unique(y, return_counts=True)
    probabilities = counts / counts.sum()
    return -np.sum(probabilities * np.log2(probabilities))

def information_gain(self, y: np.ndarray, y_left: np.ndarray, y_right: np.ndarray) ->
float:
    weight_left = len(y_left) / len(y)
    weight_right = len(y_right) / len(y)
    return self.entropy(y) - (weight_left * self.entropy(y_left) + weight_right *
self.entropy(y_right))
```

2. 准确率 / Sklearn 提供的决策树模型的准确率:

```
0.9598108747044918
0.9550827423167849
```

Lab 2.1.2 PCA & KMeans

1. 降维 & 聚类后的结果:



Lab 2.2 深度学习

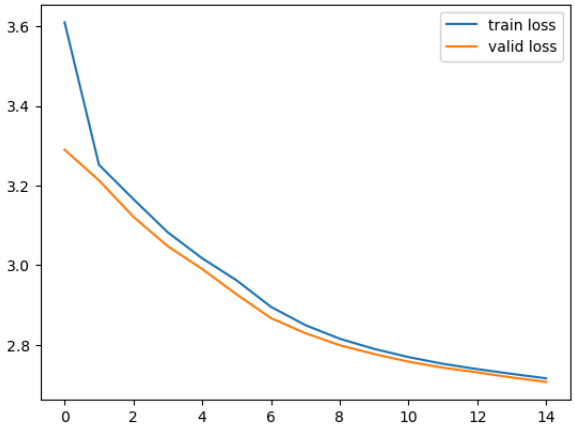
1. 为保证能在 CPU 上运行完整个实验, 做了如下修改:
   1. 仅使用了数据集的 5%:

```
text = text[: int(len(text) / 20)]
```

   2. 减少 epoch, batch size, expert 数量:

```
train_dataloader, val_dataloader = create_dataloader(
    "input.txt",
    tokenizer,
    chunk_size=20,
    batch_size=10,
)
model = SparseMoETransformer(
    vocab_size=len(tokenizer.char2index),
    seq_len=20,
    embed_size=4,
    n_layers=1,
    n_heads=1,
    num_experts=2,
    active_experts=1,
).to(device)
# ...
train_losses, valid_losses = run(
    model, train_dataloader, val_dataloader, device, epochs=15
)
```

2. 训练、测试误差随 epoch 的变化:



3. 生成的文本:

```
I could pick my lance t t the the the th the the th the th the th the th the th the th
the th the th the th the th the t
```

Lab 2.3 对语言模型进行对抗攻击

1. 构造的样本

```
This roaring saf!!!!!!!! fearless Side knocks Kitty!!!!!!!!!!!!!!! habit internetHi
Higsaw Block Ah!!!!!!!!!!!!!!! Cous assembleilled advertisingorth ''
```

2. 对抗攻击后的结果

```
This is great! I love living on the wild side!"

Mikejorie was very surprised. She smiled and said, "
```