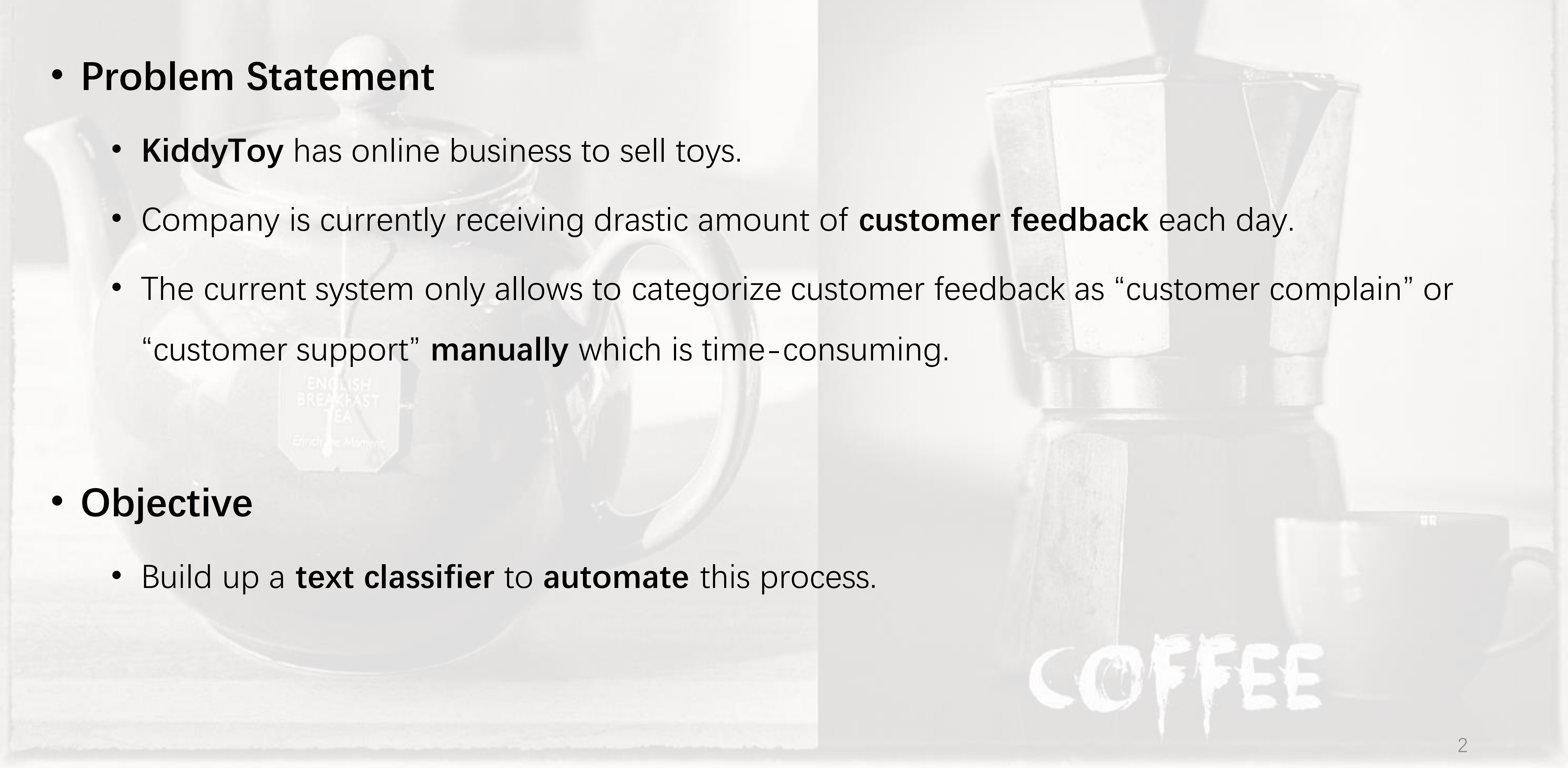
Project 3:

Text Classification

Yang Li

Introduction



- **Problem Statement**
  - **KiddyToy** has online business to sell toys.
  - Company is currently receiving drastic amount of **customer feedback** each day.
  - The current system only allows to categorize customer feedback as “customer complain” or “customer support” **manually** which is time-consuming.

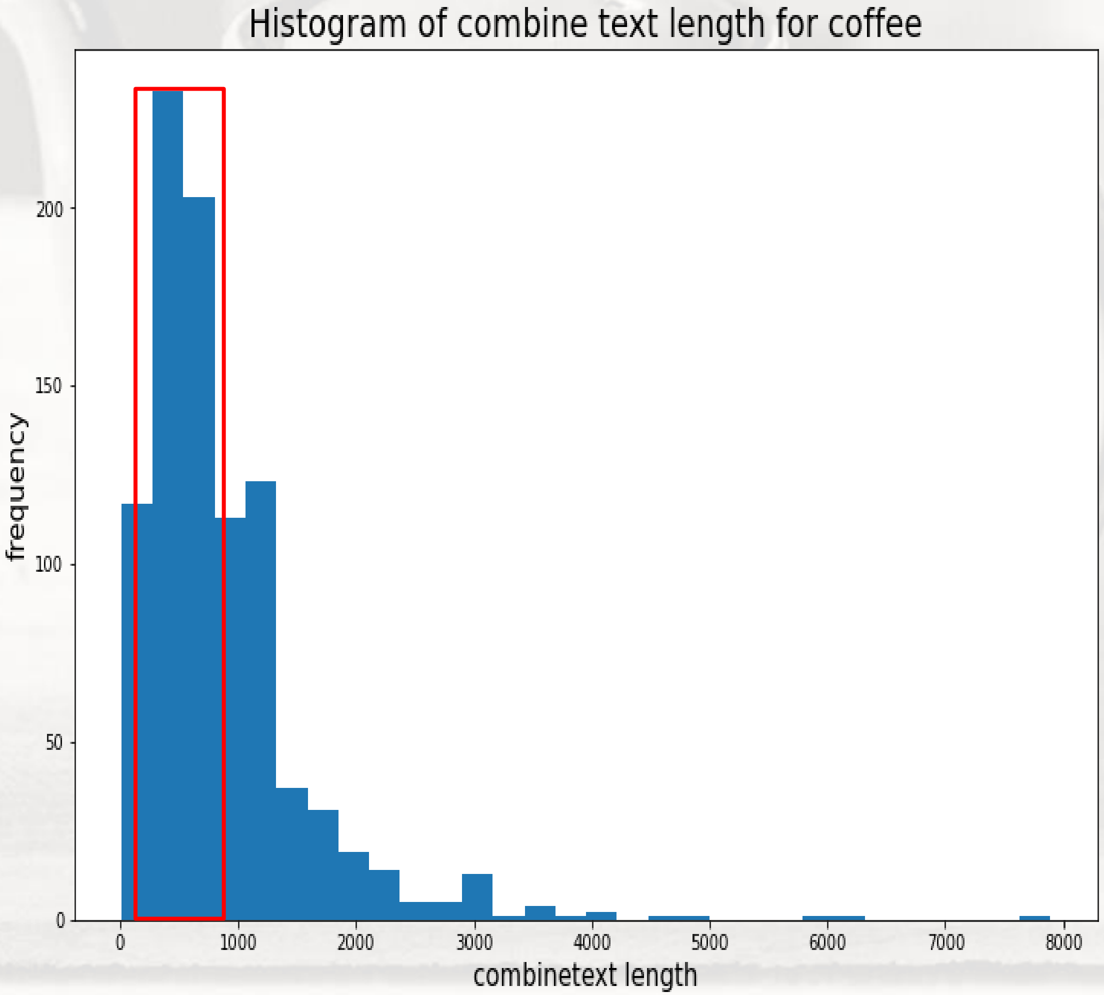
- **Objective**
  - Build up a **text classifier** to **automate** this process.

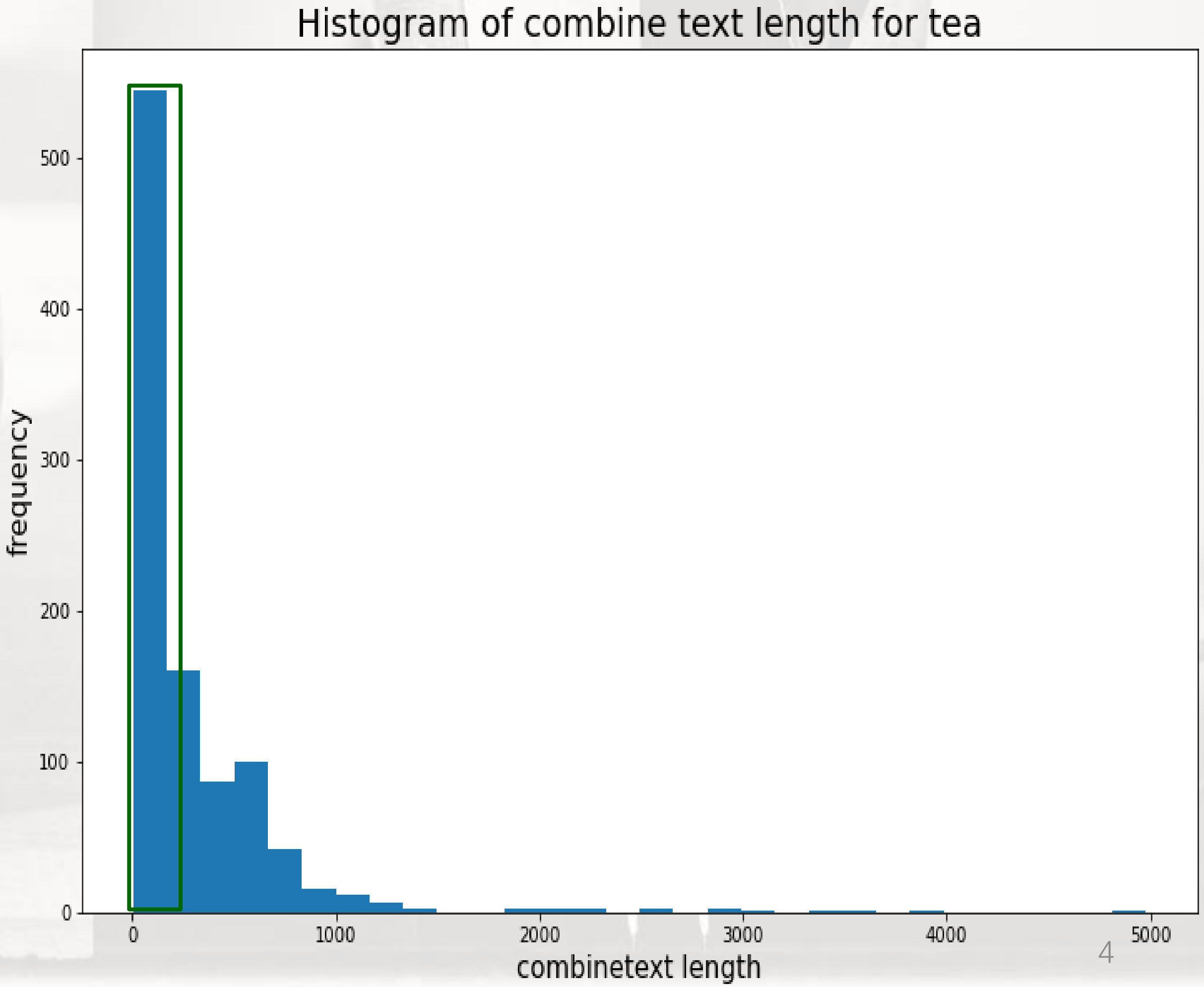
Dataset Description

- **Data Extraction**
  - Reddit API
- **Topics Selected**
  - Coffee and Tea
- **Data clean**
  - 500+ rows out of 900+ rows for Tea subreddit had no ‘selftext’
  - No missing ‘selftext’ for 900+ Coffee posts
  - concatenate ‘title’ and ‘selftext’

Exploratory Data Analysis

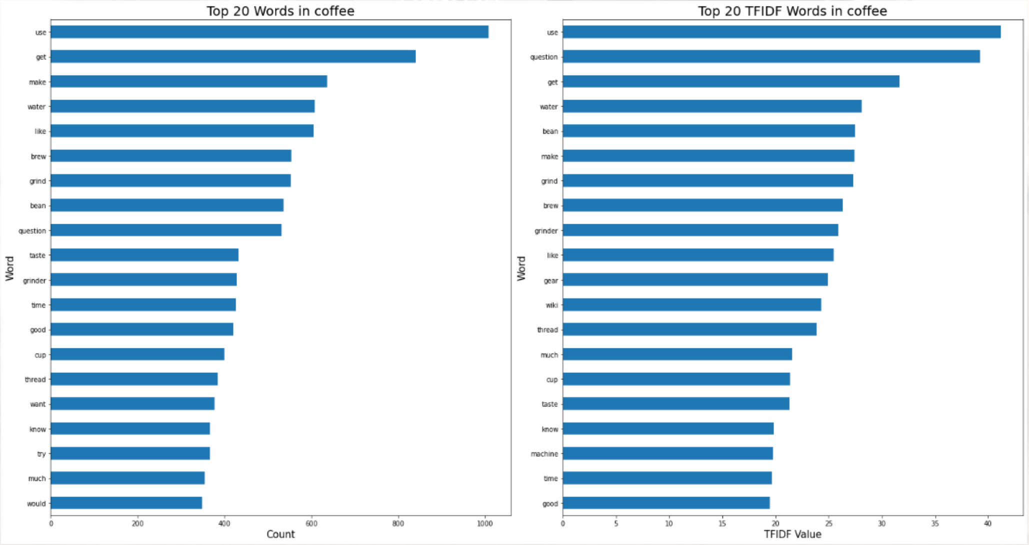
- Overall, tea posts have shorter text length than coffee posts.





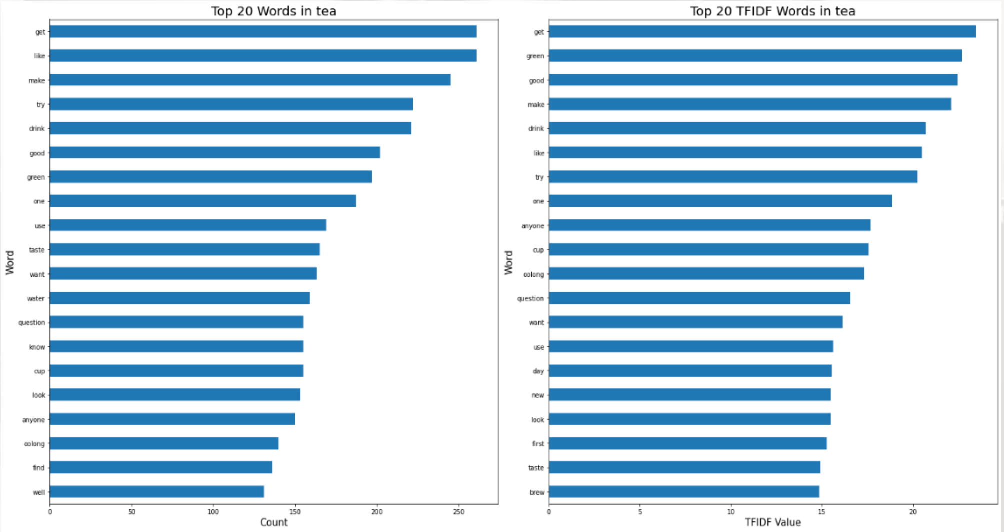
Preprocessing – bag of words

- Coffee
  - **water**, grind, bean, make, **cup**, **taste**, machine, **brew**, time, gear …
- Tea
  - **cup**, oolong, green, want, **taste**, **water**, **brew**, black …



Top 20 Words in coffee

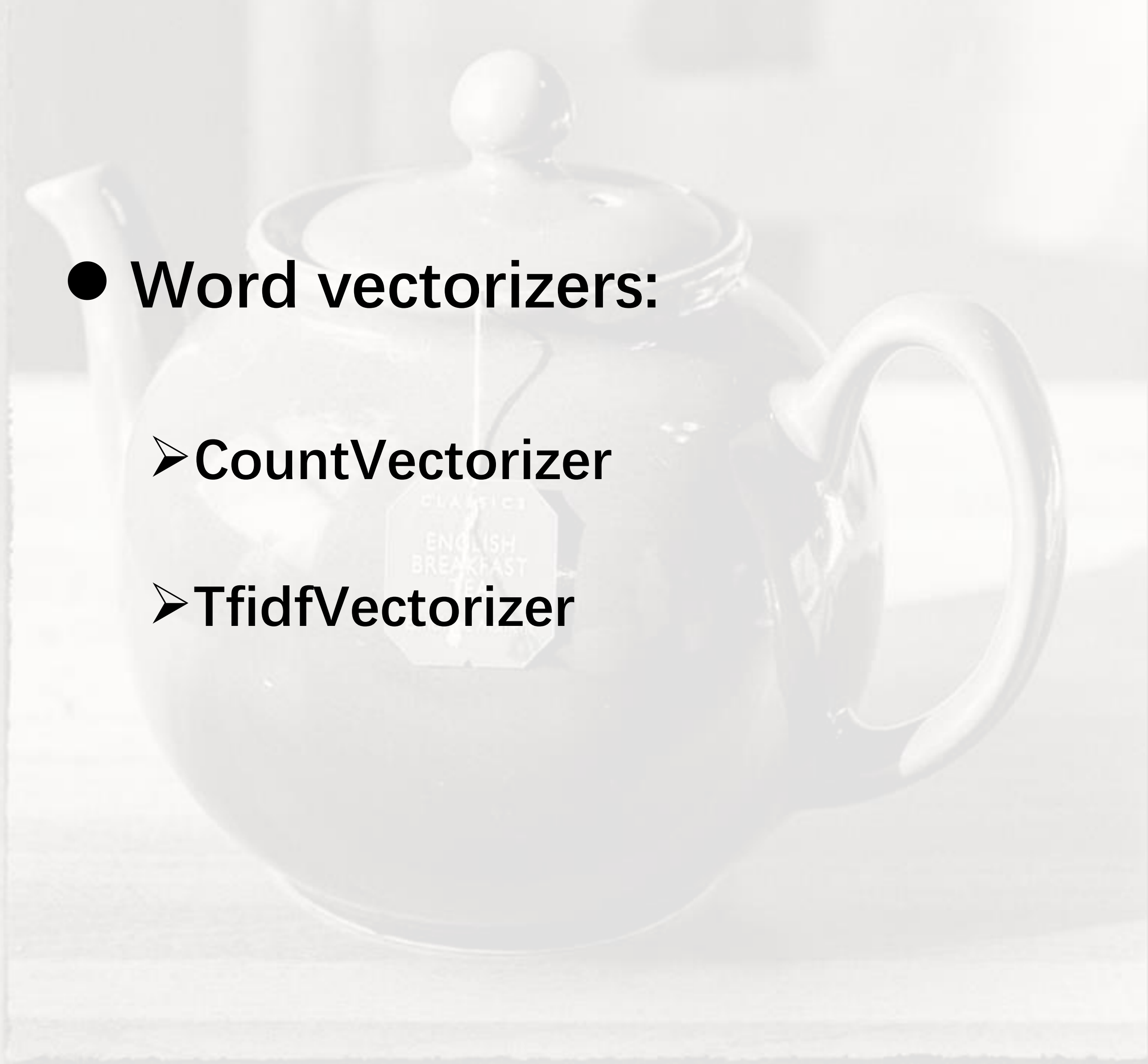
Top 20 TFIDF Words in coffee



Top 20 Words in tea

Top 20 TFIDF Words in tea

Modeling



- Word vectorizers:
  - CountVectorizer
  - TfidfVectorizer

- Classifiers:
  - Logistic Regression
  - Naive Bayes

Modeling Comparison

- **Default hyper-parameters**

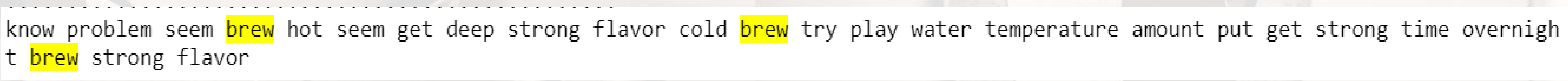
Model Name	Vectorizer	Train Score	Test Score	Overfit diff
Logistic Regression	CountVectorizer	99.5%	91.8%	7.7%
Logistic Regression	TfidfVectorizer	97.1%	91.6%	5.5%
Naive Bayes	CountVectorizer	96.2%	90.8%	5.4%
Naive Bayes	TfidfVectorizer	96.7%	90.6%	6.1%

- **Use GridSeach to tune hyper-parameters**.

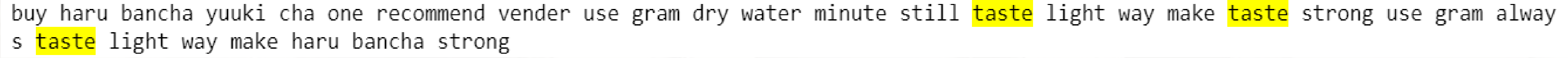
Model Name	Vectorizer	Train Score	Test Score	Overfit diff
Logistic Regression	CountVectorizer	99.1%	91.2%	7.9%
Logistic Regression	TfidfVectorizer	97.2%	91.4%	5.8%
Naive Bayes	CountVectorizer	94.2%	90.4%	3.8%
Naive Bayes	TfidfVectorizer	96.6%	92.1%	**4.5%**

Evaluation and Conceptual Understanding

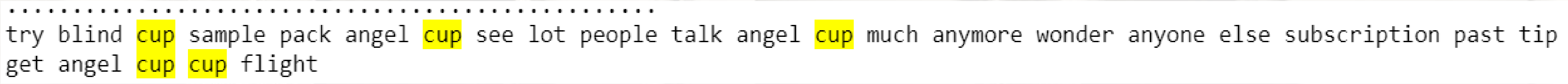
- Misclassification analysis:
  - Common words for both coffee and tea posts
    - Such as ‘brew’, ‘cup’, ‘taste’



```
..................................................
know problem seem brew hot seem get deep strong flavor cold brew try play water temperature amount put get strong time overnigh
t brew strong flavor
```

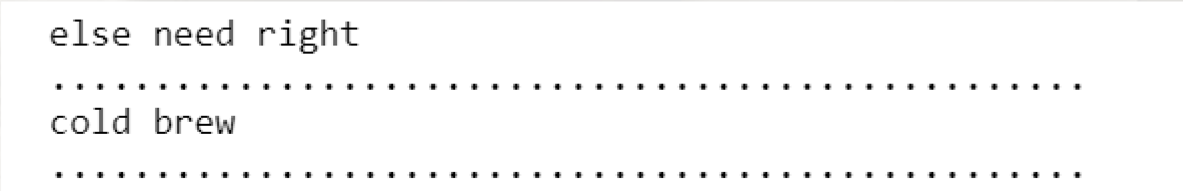


```
buy haru bancha yuuki cha one recommend vender use gram dry water minute still taste light way make taste strong use gram alway
s taste light way make haru bancha strong
```



```
.....................................................
try blind cup sample pack angel cup see lot people talk angel cup much anymore wonder anyone else subscription past tip
get angel cup cup flight
```

  - Short text length



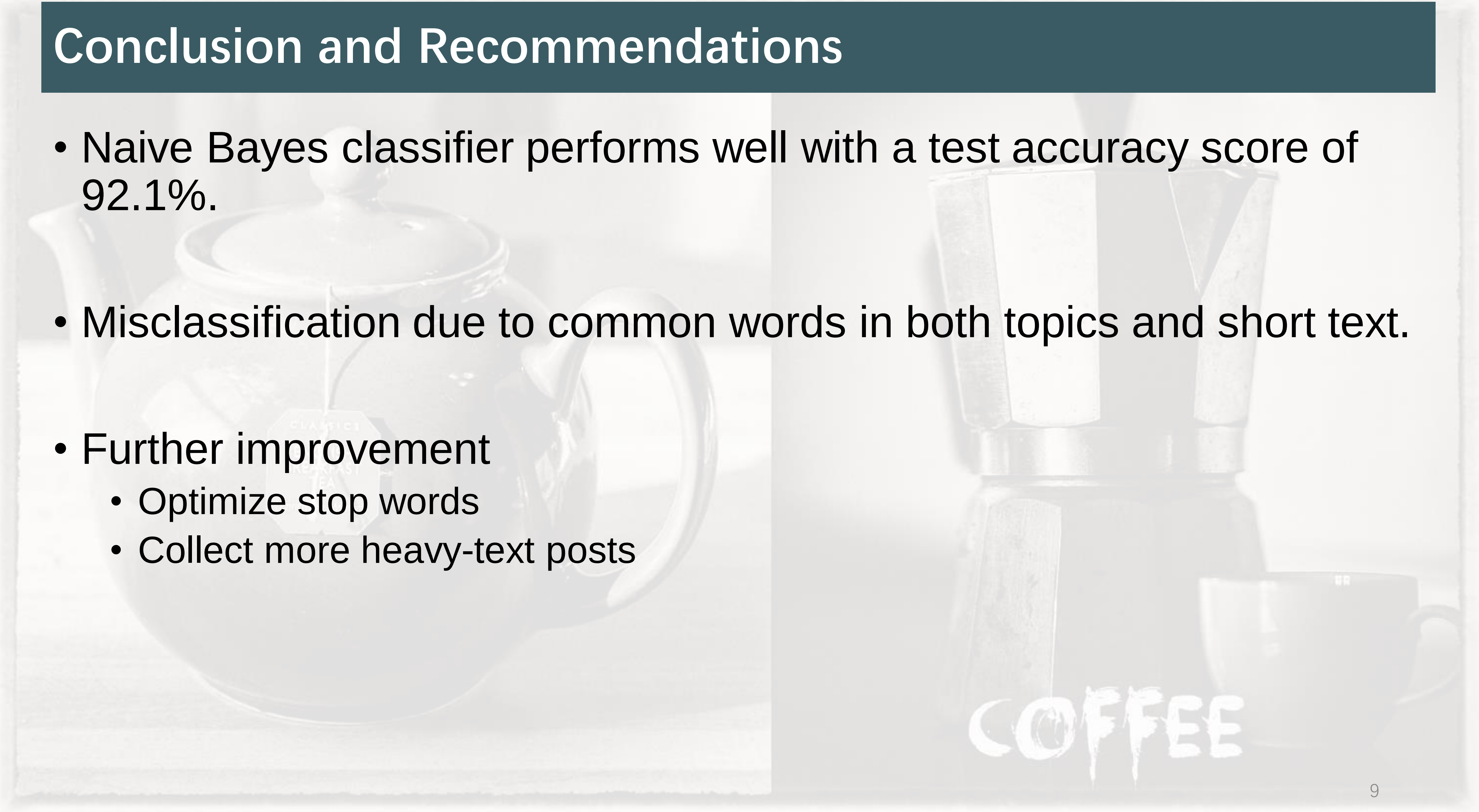
```
else need right
.....................................................
cold brew
.....................................................
```

Conclusion and Recommendations

- Naive Bayes classifier performs well with a test accuracy score of 92.1%.

- Misclassification due to common words in both topics and short text.

- Further improvement
  - Optimize stop words
  - Collect more heavy-text posts



Thank you