# ĐỒ ÁN CUỐI KỲ

# Phân tích bình luận

GVPT:
Phạm Nguyễn Trường An
Lê Đình Duy

Lớp: CS114.K21

# --> Danh sách nhóm <--

- Trịnh Hủng Long 18521060
- Hà Văn Luân 18521062
- Lỗ Đình Phưởng 18521274

# Giới thiệu

- Hiện nay nhu cầu mua hàng qua mạng của người dùng ngày càng trở nên phát triển mạnh hơn do những lợi ích mà nó mang lại, như tiện lợi, chi phí rẻ, có nhiều chương trình khuyến mãi hấp dẫn, có thể ngồi ở nhà để xem sản phẩm mà không cần phải đến tận nơi để xem, … Tuy nhiên, việc mua hàng qua mạng cũng có những nhược điểm, trong đó có việc người dùng không thể tận mắt đánh giá sản phẩm của mình như mua trực tiếp tại các cửa hàng được (Các bình luận đóng vai trò quan trọng)
- Để minh chứng hơn, chúng em sẽ tiến hành dự đoán bình luận của một sản phẩm điện thoại trên trang thương mai điện tử : *Thế Giới Di Động* để hỗ trợ việc mua hàng của người tiêu dùng.

# Mô tả

## *1. Ý tưởng*

- Đưa đoạn bình luận của 1 sản phẩm bất kỳ và đưa ra kết quả dự đoán bình luận (là tích cực, tiêu cực hoặc trung tính)

## *2. Input & Output*

- Input: Một bình luận về sản phẩm điện thoại của khách hàng bằng tiếng Việt.
- Output: Bình luận đó là tiêu cực, trung tính hay tích cực.

# Prepare Dataset

- Crawl data (các bình luận sản phẩm trên trang) từ trang thương mại điện tử: '*https://www.thegioididong.com/dtdd*' *(Thế Giới Di Động)* bằng thư viện BeautifulSoup – 1 package Python dùng để phân tích cú pháp các tài liệu HTML và XML
- Ta thu về được 4679 bình luận các nhãn được gán tự động với:

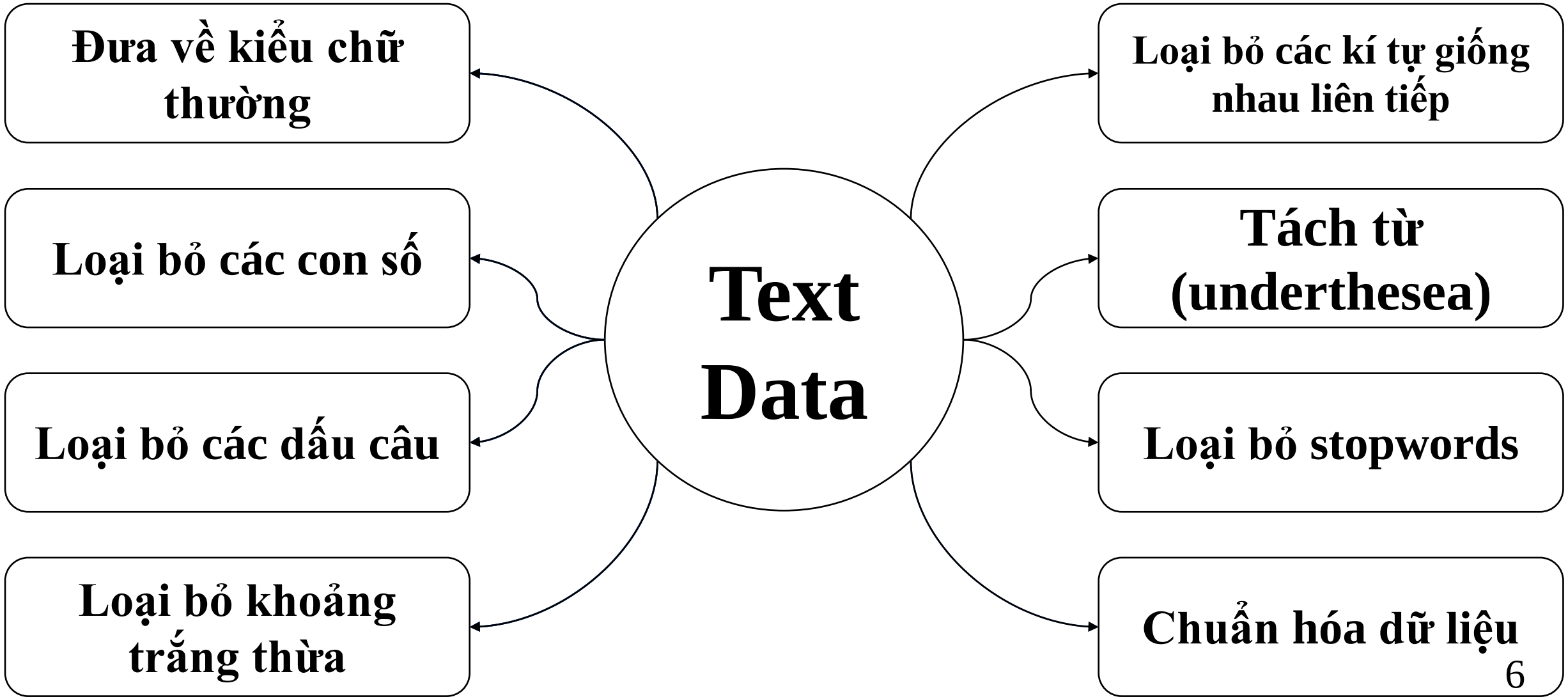
✓Các bình luận có số lượng đạt 5 sao là 5 sẽ cho nhãn bằng 1

✓Các bình luận có số lượng đạt 5 sao là 4 và 3 sẽ cho nhãn bằng 0

✓Các bình luận có số lượng đạt 5 sao là 1 và 2 sẽ cho nhãn bằng -1

- Sau đó, tụi em sẽ thực hiện kiểm tra các nhãn của bộ data trên, thu gọn và làm cân bằng bộ data trên -> thu về được 3000 bình luận (1000 bình luận tích cực, 1000 bình luận trung tính và 1000 bình luận tiêu cực)

# Data preprocessing



# Data preprocessing

## * *Loại bỏ stopwords:*

Link: "https://github.com/stopwords/vietnamese-stopwords/blob/master/vietnamese-stopwords-dash.txt"


a_lô
a_ha
ai
ai_ai
ai_nấy
ai_đó
alô
amen
anh
anh_ấy
ba
ba_ba
ba_bản
ba_cùng
ba_họ
ba_ngày
ba_ngôi
ba_tăng
bao_giờ
bao_lâu
bao_nhiêu
bao_nả
bay_biến
biết
biết_bao
biết_bao_nhiêu
biết_chắc
biết_chừng_nào
biết_mình
biết_mấy
biết_thế
biết_trước
biết_được
buổi
buổi_làm
buổi_mới
buổi_ngày
buổi_sớm
bà
bà_ấy
bài
bài_bác
bài_bỏ
bài_cái
bác
bán
bán_cấp
bán_dạ
bán_thế
bây_bẩy
bây_chừ
bây_giờ
bây_nhiêu
bèn
béng
bên
bên_bị
bên_có
bên_cạnh
bông
bước
bước_khỏi
bước_tới
bước_đi
cho_tới_khi
cho_về
cho_ăn
cho_đang
cho_được
cho_đến
cho_đến_khi
cho_đến_nỗi
choa
chui_cha
chung
chung_cho
chung_chung
chung_cuộc
chung_cục
chung_nhau
chung_qui
chung_quy
chung_quy_lại
chung_ái
chuyển
chuyển_tự
chuyển_đạt
chuyện
chuẩn_bị
chành_chạnh
chí_chết
chùn_chùn
chùn_chũn
chú
chú_dẫn
chú_khách
dào
dì
dưới
dưới_nước
dạ
dạ_bán
dạ_con
dạ_dài
dạ_dạ
dạ_khách
dần_dà
dần_dần
dầu_sao
dẫn
dẫu
dẫu_mà
dẫu_rằng
dẫu_sao
em
em_em
giá_trị
giá_trị_thực_tế
giờ
giờ_lâu
giờ_này
giờ_đi
giờ_đây
giờ_đến
giữ
giữ_lấy
giữ_ý
giữa
ngồi_trệt
ngộ_nhỡ
nhau
nhiên_hậu
nhiệt_liệt
nhung_nhăng
nhà
nhà_chung
nhà_khó
nhà_làm
nhà_ngoài
nhà_ngươi
nhà_tôi
nhà_việc
nhóm
nhón_nhén
nhất_loạt
nhất_luật
nhất_là
nhất_mực
nhất_nhất
nhất_quyết
nhất_sinh
nhất_thiết
nhất_thì
nhất_tâm
nhất_tề
nhất_đán
nhất_định
nhận_biết
nhận_họ
nhận_làm

# Data preprocessing

** Chuẩn hóa dữ liệu:*

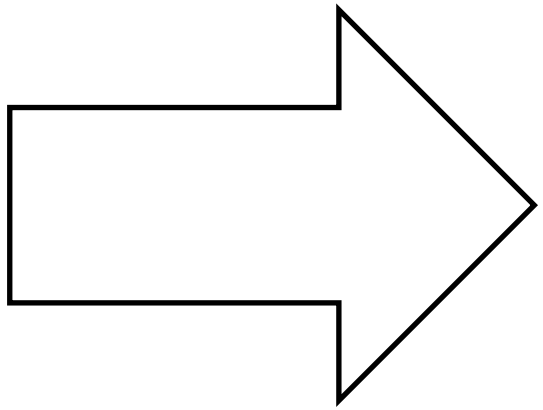
Trong lúc thu thập dữ liệu, em thu thập được một dict chứa các teencode, viết tắt,… Sau đó tìm trong các bình luận nếu chứa các từ giống với key của phần tử trong replace_list ,ta gán giá trị từ đó bằng value của key tương ứng

```
replace_list = {
        'ship': 'vận chuyển', 'shop': 'cửa hàng', 'sho': 'cửa hàng', 'm': 'mình', 'mik': 'mình', 'ko': 'không' , 'k': 'không', 'kh': 'không'
        , 'khong': 'không', 'kg': 'không', 'khg': 'không', 'tl': 'trả lời','rep': 'trả lời', 'r': 'rồi', 'fb': 'facebook', 'face': 'faceook'
        , 'thanks': 'cảm ơn', 'thank': 'cảm ơn', 'tks': 'cảm ơn', 'tk': 'cảm ơn', 'ok': 'tốt', 'oki': 'tốt', 'okie': 'tốt', 'sp': 'sản phẩm'
        ,'dc': 'được', 'vs': 'với', 'đt': 'điện thoại', 'thjk': 'thích', 'thik': 'thích', 'qá': 'quá', 'trể': 'trễ', 'bgjo': 'bao giờ'
        , 'h': 'giờ', 'qa': 'quá', 'dep': 'đẹp', 'xau': 'xấu', 'ib': 'nhắn tin','cute': 'dễ thương', 'sz': 'size', 'good': 'tốt', 'god': 'tốt'
`       , 'bt': 'bình thường','sz': 'cỡ', 'size': 'cỡ', 'đx': 'được', 'dk': 'được', 'dc': 'được', 'đk': 'được','đc': 'được'
        , 'authentic': 'chuẩn chính hãng', 'aut': 'chuẩn chính hãng', 'auth ': 'chuẩn chính hãng', 'thick': 'thích','gud': 'tốt', 'god': 'tốt'
        , 'wel done': 'tốt', 'good': 'tốt', 'gút': 'tốt','sấu': 'xấu', 'gut': 'tốt', 'tot': 'tốt', 'nice': 'tốt', 'perfect': 'rất tốt'
        , 'bt': 'bình thường','time': 'thời gian', 'qá': 'quá', 'ship': 'giao hàng','product': 'sản phẩm', 'quality': 'chất lượng', 'chat': 'chất'
        , 'excelent': 'hoàn hảo', 'bad': 'tệ', 'sad': 'tệ','beautiful': 'đẹp', 'tl': 'trả lời', 'r': 'rồi', 'order': 'đặt hàng'
        ,'chất lg': 'chất lượng', 'sd': 'sử dụng', 'dt': 'điện thoại', 'nt': 'nhắn tin', 'tl': 'trả lời', 'sài': 'xài', 'bjo': 'bao giờ',
        'thik': 'thích', 'sop': 'cửa hàng', 'fb': 'facebook', 'face': 'facebook', 'very': 'rất','dep': 'đẹp', 'xau': 'xấu', 'iu': 'yêu'
        , 'fake': 'giả mạo', 'trl': 'trả lời', '><': 'tiêu cực','por': 'tệ', 'poor': 'tệ', 'ib': 'nhắn tin', 'rep': 'trả lời', 'fback': 'feedback'
        , 'fedback': 'feedback','bin': 'pin', 'cx': 'cũng', 'nch': 'nói chuyện', 'ntn': 'như thế nào', 'vde': 'vấn đề'
    }
```

# Data preprocessing

Ví dụ:

"Hôm qua 2/3/2019, mình có mua tại cơ sở Thế Giới Di Động Võ Văn Ngân, sản phẩm đẹp, chất lượng, pin trâu, sạc nhanh. Nhân viên nhiệt tình chu đáo kkkkkkk thanks"



"hôm qua mua cơ sở thế giới di động võ văn ngân sản phẩm đẹp chất lượng pin trâu sạc nhân viên nhiệt tình chu đáo k cảm ơn"

# Feature Engineering

- **TF-IDF** (Term Frequency – Inverse Document Frequency)
  +TF: Tần số xuất hiện của 1 từ trong văn bản
  +IDF: Tần số nghịch của 1 từ trong một tập các văn bản
- Kỹ thuật **TF-IDF** dùng để tính toán tuần suất xuất hiện của một từ trong văn bản, dựa trên đó để đánh giá mức độ quan trọng của từng từ trong văn bản
- TfidfVectorizer dùng để chuyển đổi dữ liệu văn bản sang ma trận các features **TF-IDF**

"Hôm qua 2/3/2019, mình có mua tại cơ sở Thế Giới Di Động Võ Văn Ngân, sản phẩm đẹp, chất lượng, pin trâu, sạc nhanh. Nhân viên nhiệt tình chu đáo kkkkkkk thanks"

**TF-IDF**

```
[[0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]
 .....
 [0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]]
```

# Xây dựng và huấn luyện model

Cách tính độ chính xác của model bằng *score* chỉ cho ta biết phần trăm dữ liệu được phân loại đúng mà không chỉ ra được phân loại như thế nào nên ta sử dụng một ma trận gọi là *confusion matrix*.

Bài toán này có 3 class (tích cực, tiêu cực và trung tính)nên sẽ có True/False Positive, True/False Negative, True/False Neutral.

| True label \ Predicted label | -1 | 0 | 1 |
|---|---|---|---|
| -1 | True Negative | False Neutral | False Positive |
| 0 | False Negative | True Neutral | False Positive |
| 1 | False Negative | False Neutral | True Positive |

Confusion Matrix

$$\text{Precision\_1} = \frac{\text{True Negative}}{\text{True Negative + False Negative +False Negative}}$$

$$\text{Precision\_2} = \frac{\text{True Neutral}}{\text{True Neutral + False Neutral +False Neutral}}$$

$$\text{Precision\_3} = \frac{\text{True Positive}}{\text{True Positive + False Positive +False Positive}}$$

$$\text{Recall\_1} = \frac{\text{True Negative}}{\text{True Negative + False Neutral +False Positive}}$$

$$\text{Recall\_2} = \frac{\text{True Neutral}}{\text{True Neutral + False Negative +False Positive}}$$

$$\text{Recall\_3} = \frac{\text{True Positive}}{\text{True Positive + False Neutral +False Negative}}$$

$$\text{F1-Score\_1} = \frac{2 \text{ x } (\text{Precision\_1} + \text{Recall\_1})}{\text{Precision\_1} + \text{Recall\_1}}$$

$$\text{F1-Score\_2} = \frac{2 \text{ x } (\text{Precision\_2} + \text{Recall\_2})}{\text{Precision\_2} + \text{Recall\_2}}$$

$$\text{F1-Score\_3} = \frac{2 \text{ x } (\text{Precision\_3} + \text{Recall\_3})}{\text{Precision\_3} + \text{Recall\_3}}$$

$$\textbf{F1-Score} = \frac{\text{F1-Score\_1} + \text{F1-Score\_2} + \text{F1-Score\_3}}{3}$$

# Xây dựng và huấn luyện model

## Model SVC

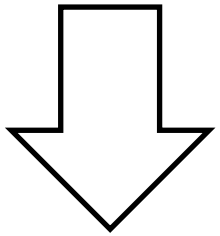
```
Model SVC
Train score:  0.9079166666666667
Test score:  0.745
F1 score:  0.7447209222834282
```

## Multinomial Naïve Bayes

```
Model MultinomialNB
Train score:  0.8495833333333334
Test score:  0.7433333333333333
F1 score:  0.7414419125535217
```

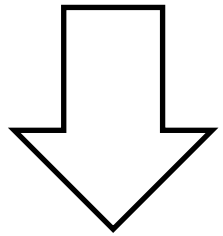
# Xây dựng và huấn luyện model

## Logistic Regression



```
Model LogisticRegression
Train score:  0.8554166666666667
Test score:  0.735
F1 score:  0.7304342878113371
```

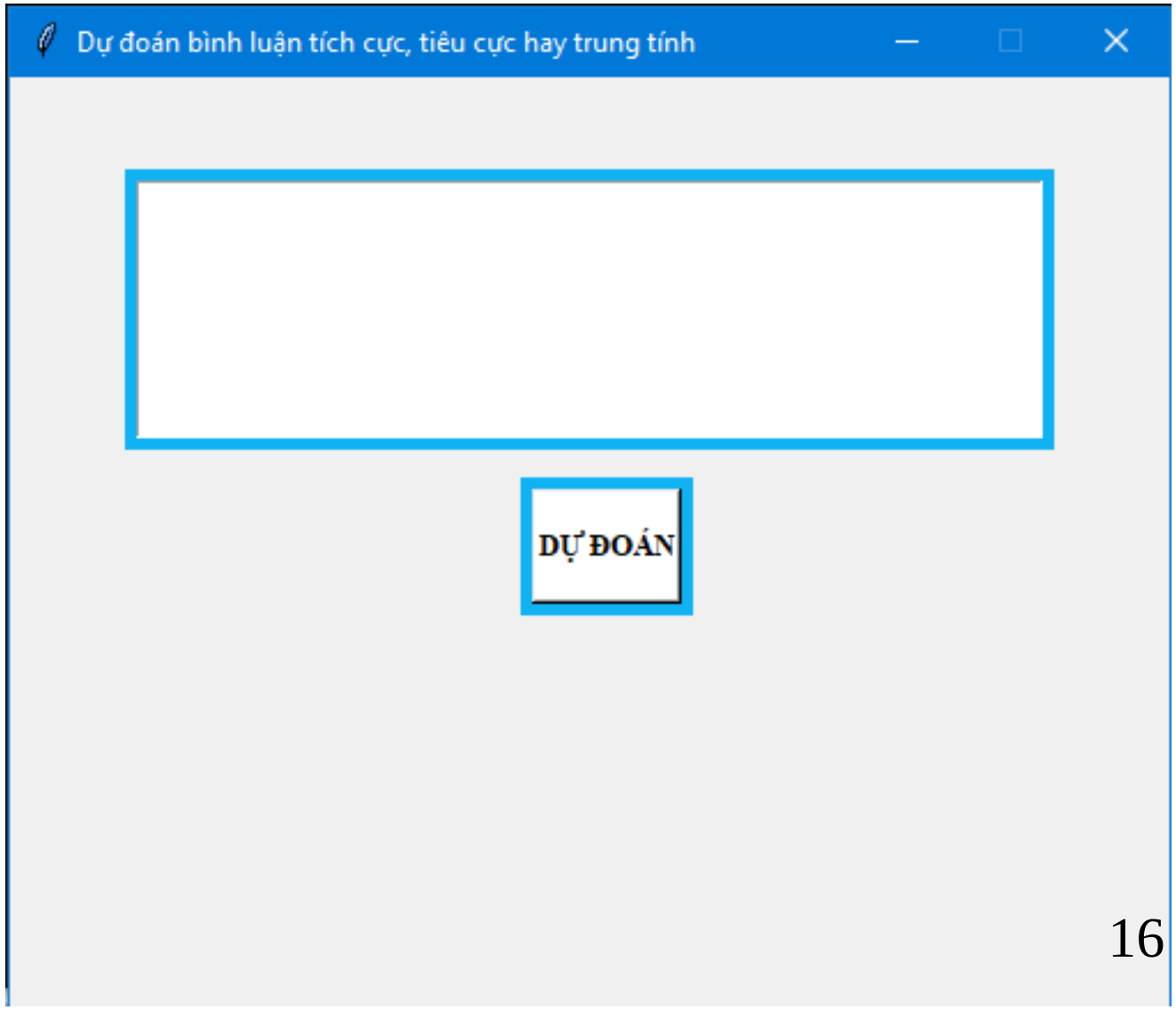
## Random Forest



```
Model RandomForestClassifier
Train score:  0.99125
Test score:  0.65
F1 score:  0.6488799173754572
```

# Thiết kế giao diện

Sử dụng thư viện **Tkinter** – một package trong python có chứa module Tk hỗ trợ cho việc lập trình GUI (Graphical User Interface)



# DEMO


Thanks for listening