



การจำแนกข้อความ ที่เข้าข่ายการกลั่นแกล้ง บนโลกไซเบอร์

cyberbullying



NEXT

ที่มาและความสำคัญ



การกลั่นแกล้งทางอินเทอร์เน็ตคือ การข่มเหงรังแกกันผ่านทางเทคโนโลยีการสื่อสารสังคมออนไลน์ การกลั่นแกล้ง ในรูปแบบนี้จะมีลักษณะเป็นการส่งข้อความ ภาพถ่าย หรือวิดีโอ ซึ่งเป็นพฤติกรรมที่ไม่ทราบแน่ชัดว่าใครเป็นผู้กระทำจะกระทำกับใครที่ไหนเมื่อไรก็ได้ เพราะคิดว่าไม่ได้เผชิญหน้ากันโดยตรง เช่น การเขียนข้อความ โฟสตีภาพอนาจารของบุคคลอื่น แล้วเปิดให้กลุ่มเพื่อนได้แสดงความคิดเห็นโดยอาจใช้ถ้อยคำหยาบคายกับผู้ที่ตกเป็นเหยื่อ ซึ่งข้อความหรือภาพอาจเป็นเรื่องจริงหรือการตัดต่อคัดลอกคัดแปลงก็ได้และมักมีการส่งต่อ เพื่อขยายวงกว้างของความเสียหาย ทำให้ผู้ที่ตกเป็นเหยื่อ ได้รับความอับอาย หรือถูกล้อเลียน ซึ่งผู้กระทำมักอ้างว่าเป็นการแสดงความคิดเห็นเพื่อความสนุกสนานเท่านั้นไม่น่าจะมีความผิดหรือ ผลเสียอะไรโดยไม่คำนึงถึงผลกระทบที่จะตามมาทั้งทางร่างกายและจิตใจ รวมถึงบุคคล หรือกลุ่มคนที่แสดงพฤติกรรมที่เป็นอันตรายโดยมีความตั้งใจที่จะทำลายชื่อเสียงหรือภาพลักษณ์ ของบุคคลอื่นด้วยวิธีการต่าง ๆ เช่น การรบกวน ข่มขู่ จนนำไปสู่ความตึงเครียดทางอารมณ์ได้ เช่น รู้สึกหดหู่ วิตกกังวล ภาวะซึมเศร้า และอาจถึงขั้นคิดฆ่าตัวตาย

จากปัญหาดังกล่าว อาจทำให้สื่อสังคมออนไลน์ถูกใช้ในการกลั่นแกล้งผู้อื่น ผู้วิจัยจึงเห็นความสำคัญในการตรวจสอบข้อความที่ผู้ใช้แสดงความคิดเห็นบนสื่อสังคมออนไลน์ โดยใช้หลักการการประมวลผลภาษาธรรมชาติ (natural language processing - NLP) และการเรียนรู้ของเครื่อง (machine learning) ซึ่งเป็นสาขาหนึ่งของปัญญาประดิษฐ์ (Artificial intelligence) เพื่อสร้างขั้นตอนจำแนกประเภทข้อความ มีลักษณะที่เข้าข่ายการกลั่นแกล้งบนโลกไซเบอร์ ออกจากข้อความทั่วไป



NEXT

วัตถุประสงค์

- เพื่อพัฒนาวิธีการสำหรับการจำแนกประเภทข้อความที่เข้าข่ายการกลั่นแกล้งบนสื่อสังคมออนไลน์
- เพื่อทดสอบประสิทธิภาพและเปรียบเทียบวิธีการสำหรับการจำแนกประเภทข้อความเข้าข่ายการกลั่นแกล้งบนสื่อสังคมออนไลน์

ขอบเขตการดำเนินงาน

ข้อมูลจากเฟซบุ๊ก เป็นความคิดเห็นเกี่ยวข้องกับบุคคลในวงการบินไทย ตัวอย่างเช่น คารา เน็ตไคคอล นักร้อง ศิลปิน เป็นต้น
จำนวน 1,110 ข้อความ



NEXT

DATA EXAMPLE



1	Tweet	Text_Label
2	ถ้าไม่กินข้าวจะผอมเป็นไม่เหมือนพี่นะ	Bullying
3	คุณนั่นสิ ไม่ชอบกินผักเลยอ้วนหน้าเกลียดเหมือนหมูเลย	Bullying
4	กูที่เห็นคอมเมนต์? โง่ๆแบบนี้	Bullying
5	สิ่งที่เป็นภาระคือพวกมึง	Bullying
6	ถ้าสมองมีแค่นี้ก็ออกไปทำอย่างอื่นเถอะ	Bullying
7	เว็บแม่งอย่างกาก	Bullying
8	ตอบคำถามได้ควายจัง อนาคตเด็กต้องมาเจอระบบแบบพวกมึงคิดเนี่ยนะ	Bullying
9	เกิดมาทำไม อีควาย	Bullying
10	ตกใจหมด ไอหน้าผี	Bullying
11	อีข้างน้ำ ก้าวที่แผ่นดินสะเทือนหมด	Bullying
12	อ้วนๆอย่างนี้สงสัยกินผัวเป็นอาหารเช้า	Bullying
13	น้องๆ ขอฟักทองหน่อยเพื่อนพี่เค้าชอบ	Bullying
14	แต่งตัวเหมือนเขมรเลย	Bullying
15	ทุเรศวะ ไอขอทาน	Bullying
16	จะอ้วน หน้าเหมือนหมา	Bullying
17	ฟิวเตอร์ก็ลอยแถมยังแลบลิ้น	Bullying
18	หุ่นเหมือนกรรมกรแบกปูนดีตรงไหนวะ	Bullying
19	สวยสะพรั่งมาก	Bullying
20	ส่งสารหุ่นที่ต้องมาอยู่บนใบหน้าแบบคนอย่างหล่อน	Bullying
21	ดำขึ้นหรือเปล่า	Bullying
22	หน้าดูหมอง ๆ นะ	Bullying
23	ไปทำอะไรมาโทรมจัง	Bullying
24	แต่งหน้าแบบนี้จะไปเล่นจ๊วที่ไหน	Bullying
25	ลด ๆ ได้แล้วมั้ง อ้วนใหญ่แล้วนะ	Bullying
26	อ้วนขึ้นปะเนี่ย	Bullying
27	ตัวบวม ๆ นะช่วงนี้	Bullying
28	ตัวขนาดนี้ ส่งสารกระดุมเสือบ้าง	Bullying
29	ผอมเป็นไม้เสียบผีแล้วเนี่ย กินข้าวบ้างนะ	Bullying
30	บารุงหน้าบ้างก็ดีนะคะ	Bullying

NEXT

เกณฑ์การแบ่งข้อความ Bully



1. การกลั่นแกล้งทางวาจา (Verbal Bullying) คือ การสื่อสารเพื่อสื่อความหมายกลั่นแกล้ง เช่น ล้อเล่น, เรียกชื่อ, แสดงความคิดเห็นทางเพศที่ไม่เหมาะสม, เหน็บแนม และขู่ว่าจะทำอันตราย
2. การกลั่นแกล้งทางสังคม (Social Bullying) คือ วิธีการทำให้เสียหน้า หรือแกล้งให้สูญเสียความสัมพันธ์กับผู้อื่น อย่างตั้งใจ เช่น กระจายข่าวลือให้เสียหาย, ทำให้เกิดความอับอายในที่สาธารณะ
3. การกลั่นแกล้งทางกายภาพ (Physical Bullying) คือ การกลั่นแกล้งที่เกี่ยวข้องกับร่างกายและสวัสดิภาพของผู้ถูกกลั่นแกล้ง เช่น การทุบตี ทำร้าย ทำให้สะดุด แหย่สิ่งของ แสดงออกทำท่าทางหยาบคายใส่

NEXT



เทคนิคที่นำมาใช้

จำแนกประเภทข้อความที่เข้าข่ายการกลั่นแกล้งบนสื่อสังคมออนไลน์ด้วยวิธีการ Text classification โดยการสร้างแบบจำลองการวิเคราะห์ความรู้สึก (Sentiment Analysis) วิธีที่ใช้ประเมินประสิทธิภาพแบบจำลองเพื่อจำแนกข้อความการกลั่นแกล้งบนสื่อสังคมออนไลน์ ได้แก่ การถดถอยโลจิสติก (Logistic regression) ซัพพอร์ตเวกเตอร์แมชชีน (Support Vector machine – SVM) โครงข่ายประสาทเทียม (Artificial neural networks) และ Decision tree (ต้นไม้ตัดสินใจ)

NEXT

วิธีการสร้างแบบจำลอง

นำเข้าข้อมูลที่จะใช้ในการสร้างแบบจำลอง

```
import string
import re

import numpy as np
import matplotlib.pyplot as plt

from sklearn.linear_model import LogisticRegression
from sklearn.preprocessing import StandardScaler

import pandas as pd

from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.feature_extraction.text import HashingVectorizer

!pip install https://github.com/PyThaiNLP/pythainlp/archive/dev.zip
```

อ่านข้อมูลจากไฟล์



```
[ ] import pandas as pd

df = pd.read_csv('BullyDataset.csv', sep='\t', names=['Tweet', 'Text'], header=None)
df
```

	Tweet	Text
0	ถ้าไม่กินข้าวจะผอมเป็นไม่เหมือนพี่นะ	Bullying
1	ดูคนนั้นสิ ไม่ชอบกินผักเลยอ้วนหน้าเกลียดเหมือน...	Bullying
2	กูที่เห็นคอมเมนต์? โง่ๆแบบนี้	Bullying
3	สิ่งที่เป็นการคือพวกมึง	Bullying
4	ถ้าสมองมีแค่นี้ก็ออกไปทำอย่างอื่นเถอะ	Bullying
...
1105	สวยแบบสตรีโหลลึ่คแม่	Non-Bullying
1106	แข็งแรงมากแม่ 555	Non-Bullying
1107	คิม โนะ โต๊ะ	Non-Bullying
1108	แบัวมากคะยาย5555	Non-Bullying
1109	น่ารักก็พอแล้วปะ555	Non-Bullying

1110 rows x 2 columns

NEXT

ตัดคำ ลบคำ stopwords และ punctuation (เครื่องหมายวรรคตอน)
ออกจากข้อความ ถัดมาเราจะดึง array ของ stopwords หรือคำที่ไม่
ค่อยสื่อความหมาย จาก PyThaiNLP มาเก็บไว้ที่ตัวแปร
thai_stopwords



#ตัดคำ ลบคำ

```
from pythainlp.corpus.common import thai_stopwords
thai_stopwords = list(thai_stopwords())
thai_stopwords
```

```
['เท่าไร',
 'เช่นนี้',
 'พร้อมทั้ง',
 'นี้แหละ',
 'นานๆ',
 'จะได้',
 'นะ',
 'ตั้ง',
 'ที่ไร',
 'นั้นๆ',
```

```
!pip install sklearn-pycrfsuite
from pythainlp import word_tokenize
def text_process(Tweet):
    final = "".join(u for u in Tweet if u not in ("?", ".", ";", ":",
    final = word_tokenize(final)
    final = " ".join(word for word in final)
    final = " ".join(word for word in final.split()
                        if word.lower not in thai_stopwords)
    return final
df['Tweet_tokens'] = df['Tweet'].apply(text_process)
df
```

	Tweet	Text	Tweet_tokens
0	ถ้าไม่กินข้าวจะผอมเป็นไม่เหมือนพี่นะ	Bullying	ถ้า ไม่ กินข้าว จะ ผอม เป็น ไม่ เหมือน พี่ นะ
1	ดูคนนั้นสิ ไม่ชอบกินผักเลยอ้วนหน้าเกลียดเหมือน...	Bullying	ดู คน นั้น สิ ไม่ ชอบ กิน ผัก เลย อ้วน หน้า เกล...
2	กูที่เห็นคอมเมนต์? โง่ๆแบบนี้	Bullying	กู ที่ เห็น คอมเมนต์ โง่ แบบนี้
3	สิ่งที่เป็นภาระคือพวกมึง	Bullying	สิ่ง ที่ เป็น ภาระ คือ พวก มึง
4	ถ้าสมองมีแค่นี้ก็ออกไปทำอย่างอื่นเถอะ	Bullying	ถ้า สมอง มี แค่นี้ ก็ ก็ ออก ไป ทำ อย่าง อื่น ...
...
1105	สวยแบบสตรีเหล็กค่ะแม่	Non-Bullying	สวย แบบ สตรี เหล็ก ค่ะ แม่
1106	แข็งแรงมากแม่ 555	Non-Bullying	แข็งแรง มาก แม่ 555
1107	คิมโน้โตะ	Non-Bullying	คิ ม โน้ โตะ
1108	แบ้วมากค่ะยาย5555	Non-Bullying	แบ้ว มาก ค่ะ ยาย 5555
1109	น่ารักก็พอแล้วปะ555	Non-Bullying	น่ารัก ก็ พอ แล้ว ปะ 555

1110 rows x 3 columns

NEXT

ข้อความไม่บุลลึ

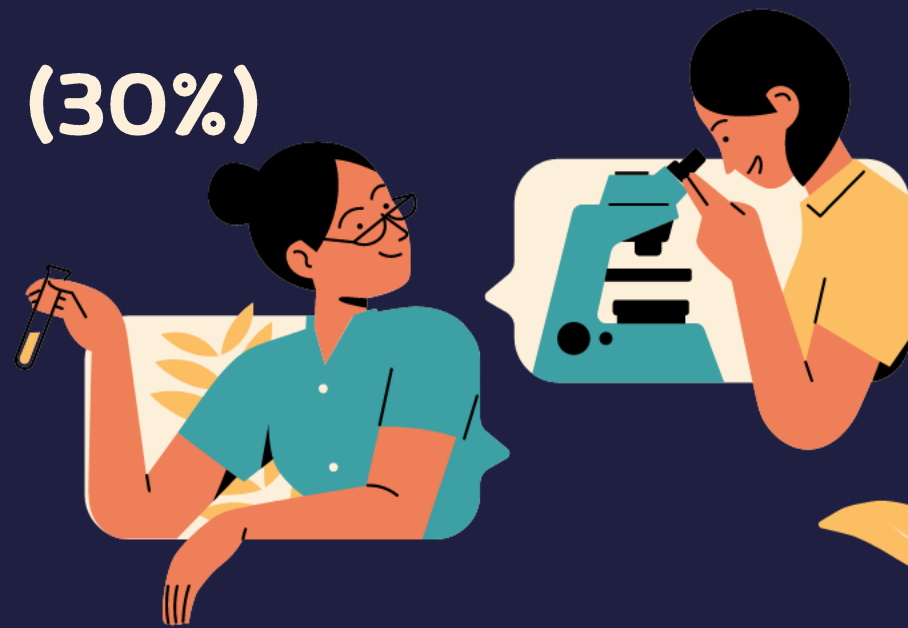


ข้อความบุลลั

NEXT

Split ข้อมูลเป็น Train (70%) Test (30%)

แบ่งข้อมูลออกเป็น 2 ส่วน คือส่วนของการฝึกฝน (train) แบบจำลอง 70% และส่วนของการทดสอบ (test) แบบจำลอง 30% โดย X คือตัวแปรต้นที่เป็นข้อความ และ y คือตัวแปรตามที่เป็น sentiment (pos หรือ neg)



```
#Split ข้อมูลเป็น Train (70%) Test (30%)  
from sklearn.model_selection import train_test_split  
  
X = df[['Tweet_tokens']]  
y = df['Text']  
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=101)
```

NEXT

Word Vectorizer และ Bag-of-Words (BoW)



```
#Word Vectorizer และ Bag-of-Words (BoW)
#การจะสร้างแบบจำลองวิเคราะห์ความรู้สึกจากข้อความ จำเป็นจะต้องแปลงข้อมูลให้อยู่ในรูปแบบที่มีจำนวนของตัวแปรที่เท่ากัน
#โดยเริ่มต้นเราจะใช้ Count Vectorizer ของ sklearn มาช่วยในการจัดการดึงค่าทั้งหมดออกมาจากข้อความ และจัดเก็บในรูปแบบ Vector

from sklearn.feature_extraction.text import CountVectorizer
cvec = CountVectorizer(analyzer=lambda x:x.split(' '))
cvec.fit_transform(X_train['Tweet_tokens'])
cvec.vocabulary_

{'ก็': 93,
 'จี': 247,
 'แหละ': 1426,
 'สก็อย': 872,
 'อปเเกรด': 1060,
 'ทำตัว': 456,
 'ไร้ค่า': 1511,
 'คราวหน้า': 179,
 'ไม่': 1503,
 'อยาก': 1032,
 'ตอย': 410,
 'จอ': 261,
 'แล้ว': 1418,
```

การจะสร้างแบบจำลองวิเคราะห์ความรู้สึกจากข้อความ จำเป็นจะต้องแปลงข้อมูลให้อยู่ในรูปแบบที่มีจำนวนของตัวแปรที่เท่ากัน โดยเริ่มต้นเราจะใช้ Count Vectorizer ของ sklearn มาช่วยในการจัดการดึงค่าทั้งหมดออกมาจากข้อความ และจัดเก็บในรูปแบบ Vector (จะคล้าย ๆ กับลักษณะของพจนานุกรมที่มีการระบุตัวเลข index ของแต่ละคำด้วย)

NEXT

	#)	+	1	100	17	2	2022	25	27	3	32	4	5	53535362	55	555	5555	55555	555555	5555555	55555555	555555555
Tweet_tokens																							
ก็ จี แหละ สก๊อย อ๊ปเกรด	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ทำตัว ไร้ค่า	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
คราวหน้า ไม่ อยาก ต่อย จอ แล้ว อะ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
มัน ไม่ มันส์ ซ้ำม ปี หลอก หัว ร้อน ซ้ำม ปี อะ ดี	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ผม ทรง นี้ ที่ กิต เท่มาก	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
...
พรุน ไป ทั้งตัว ละ เพิ่ง มา ขอ โทด	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0



จากนั้นที่บรรทัดแรกทำการสร้าง Bag-of-words (Bow) ที่เปรียบเสมือนกับตารางที่มีแถวเป็นข้อความ คอลัมน์เป็นคำทั้งหมด และค่าคือจำนวนคำที่ปรากฏในข้อความดังกล่าว และเราจะใช้ Bow นี้ในการฝึกฝน แบบจำลอง โดยบรรทัดที่ 2 ใช้แสดงข้อมูลให้เห็นภาพเฉย ๆ

NEXT



แบบจำลอง

#สร้างแบบจำลอง Logistic Regression เพื่อจำแนกความ ไม่บูลี่ หรือ บูลี่

```
from sklearn.linear_model import LogisticRegression
lr = LogisticRegression()
lr.fit(train_bow, y_train)
```

```
from sklearn.metrics import confusion_matrix, classification_report
test_bow = cvec.transform(X_test['Tweet_tokens'])
test_predictions = lr.predict(test_bow)
```

#สร้างแบบจำลอง Decition tree

```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
```

```
from sklearn.model_selection import train_test_split
X = df[['Tweet_tokens']]
y = df['Text']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=101)
```

```
from sklearn.tree import DecisionTreeClassifier
from sklearn import tree
```

```
mtree = tree.DecisionTreeClassifier(max_depth=4)
mtree.fit(train_bow, y_train)
```

```
from sklearn.metrics import confusion_matrix, classification_report
test_bow = cvec.transform(X_test['Tweet_tokens'])
test_predictions2 = mtree.predict(test_bow)
```

NEXT



แบบจำลอง

#สร้างแบบจำลอง artificial neural network

```
from sklearn import datasets
from sklearn.neural_network import MLPClassifier
from sklearn.model_selection import cross_val_score
import numpy as np
import matplotlib.pyplot as plt

from sklearn.model_selection import train_test_split
X = df[['Tweet_tokens']]
y = df['Text']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=101)
```

```
model = MLPClassifier(hidden_layer_sizes=(10, 10, 10), max_iter=1000)
model.fit(train_bow, y_train)
```

```
test_bow = cvec.transform(X_test['Tweet_tokens'])
test_predictions3 = model.predict(test_bow)
```

#สร้างแบบจำลอง SVM

```
from sklearn.svm import SVC
svm = SVC(kernel="linear")
svm.fit(train_bow, y_train)
```

```
from sklearn.metrics import confusion_matrix, classification_report
test_bow = cvec.transform(X_test['Tweet_tokens'])
test_predictions4 = svm.predict(test_bow)
```

```
print ('SVM Model')
print(classification_report(test_predictions4, y_test))
```

NEXT

ผลการทดลอง

```
[ ] #ทดสอบกับข้อความที่เราสร้างขึ้นเอง
```

```
my_text = 'ลดได้แล้วมั้ง อ้วนขนาดนี้'  
my_tokens = text_process(my_text)  
my_bow = cvec.transform(pd.Series([my_tokens]))  
my_predictions = lr.predict(my_bow)  
my_predictions
```

```
array(['Bullying'], dtype=object)
```

```
[ ] my_text = 'น่ารักอะไรเบอร์นี้'
```

```
my_tokens = text_process(my_text)  
my_bow = cvec.transform(pd.Series([my_tokens]))  
my_predictions = lr.predict(my_bow)  
my_predictions
```

```
array(['Non-Bullying'], dtype=object)
```



จำแนกประเภทข้อความที่เข้าข่ายการกลั่นแกล้งบนสื่อสังคมออนไลน์ด้วยวิธีการ Text classification โดยการสร้างแบบจำลองการวิเคราะห์ความรู้สึก (Sentiment Analysis) วิธีที่ใช้ประเมินประสิทธิภาพแบบจำลองเพื่อจำแนกข้อความการกลั่นแกล้งบนสื่อสังคมออนไลน์ ได้แก่ การถดถอยโลจิสติก (Logistic regression) ซัพพอร์ทเวกเตอร์แมชชีน (Support Vector Machine – SVM) เครือข่ายประสาทเทียม (Artificial Neural Networks) และ Decision tree (ต้นไม้ตัดสินใจ)

NEXT

ผลการทดลอง

SVM Model				
	precision	recall	f1-score	support
Bullying	0.78	0.84	0.81	129
Non-Bullying	0.89	0.85	0.87	204
accuracy			0.84	333
macro avg	0.83	0.84	0.84	333
weighted avg	0.85	0.84	0.84	333



Support Vector Machine – SVM มีค่า accuracy อยู่ที่ร้อยละ 84 ค่า precision ของข้อความที่เข้าข่ายการกลั่นแกล้งบนโลกไซเบอร์ อยู่ที่ร้อยละ 78 ค่า recall อยู่ที่ร้อยละ 84 ค่า f1-score อยู่ที่ร้อยละ 81 ส่วนข้อความที่ไม่เข้าข่ายการกลั่นแกล้งบนสื่อสังคมออนไลน์ค่า precision อยู่ที่ร้อยละ 89 ค่า recall อยู่ที่ร้อยละ 85 ค่า f1-score อยู่ที่ร้อยละ 87

NEXT

ผลการทดลอง

Logistic Regression Model				
	precision	recall	f1-score	support
Bullying	0.76	0.88	0.81	119
Non-Bullying	0.93	0.84	0.88	214
accuracy			0.86	333
macro avg	0.84	0.86	0.85	333
weighted avg	0.87	0.86	0.86	333



Logistic regression มีค่า accuracy อยู่ที่ร้อยละ 86 ค่า precision ของข้อความที่เข้าข่ายการกลั่นแกล้งบนโลกไซเบอร์ อยู่ที่ร้อยละ 76 ค่า recall อยู่ที่ร้อยละ 88 ค่า f1-score อยู่ที่ร้อยละ 81 ส่วนข้อความที่ไม่เข้าข่ายการกลั่นแกล้งบนโลกไซเบอร์ ค่า precision อยู่ที่ร้อยละ 93 ค่า recall อยู่ที่ร้อยละ 84 ค่า f1-score อยู่ที่ร้อยละ 88

NEXT

ผลการทดลอง

Decition tree Model	precision	recall	f1-score	support
Bullying	0.19	0.90	0.32	30
Non-Bullying	0.98	0.63	0.77	303
accuracy			0.65	333
macro avg	0.59	0.77	0.54	333
weighted avg	0.91	0.65	0.73	333



Decision Tree Model มีค่า accuracy อยู่ที่ร้อยละ 65 ค่า precision ของข้อความที่เข้าข่ายการกลั่นแกล้งบนโลกโซเชียล อยู่ที่ร้อยละ 19 ค่า recall อยู่ที่ร้อยละ 90 ค่า f1-score อยู่ที่ร้อยละ 32 ส่วนข้อความที่ไม่เข้าข่ายการกลั่นแกล้งบนโลกโซเชียล ค่า precision อยู่ที่ร้อยละ 98 ค่า recall อยู่ที่ร้อยละ 63 ค่า f1-score อยู่ที่ร้อยละ 77

NEXT

ผลการทดลอง

Artificial neural network model				
	precision	recall	f1-score	support
Bullying	0.81	0.86	0.83	132
Non-Bullying	0.90	0.87	0.89	201
accuracy			0.86	333
macro avg	0.86	0.86	0.86	333
weighted avg	0.87	0.86	0.87	333



Artificial Neural Networks มีค่า accuracy อยู่ที่ ร้อยละ 86 ค่า precision ของข้อความที่เข้าข่ายการกลั่นแกล้งบนโลกโซเชียล อยู่ที่ร้อยละ 81 ค่า recall อยู่ที่ร้อยละ 86 ค่า f1-score อยู่ที่ร้อยละ 83 ส่วนข้อความที่ไม่เข้าข่ายการกลั่นแกล้งบนโลกโซเชียล ค่า precision อยู่ที่ร้อยละ 90 ค่า recall อยู่ที่ร้อยละ 87 ค่า f1-score อยู่ที่ร้อยละ 89

NEXT



ข้อเสนอแนะ

งานวิจัยนี้เป็นการจำแนกข้อความการกลั่นแกล้งบนสื่อสังคมออนไลน์ ไม่ได้จำกัดเพียงแค่การใช้ข้อความเท่านั้น โดยสามารถกลั่นแกล้งบนสื่อสังคมออนไลน์ได้ด้วยรูปภาพ คลิปวิดีโอ และอื่น ๆ นอกจากนี้การทดลองยังเป็นการทดลองเพื่อจำแนก ข้อความ"เข้าข่าย"การกลั่นแกล้งบนสื่อสังคมออนไลน์ในเบื้องต้นโดยพิจารณาจากเพียงความคิดเห็นเท่านั้น การวิเคราะห์ว่าข้อความเป็นข้อความการกลั่นแกล้งบนสื่อสังคมออนไลน์แท้จริงหรือไม่ต้องดูองค์ประกอบหลายอย่าง เช่น เจตนาของผู้กระทำ เป็นต้น

NEXT

Thank You

