Escuela Profesional de
Ciencia de la Computacion

UNIVERSIDAD NACIONAL DE SAN AGUSTÍN DE AREQUIPA

CLASIFICACION DE REQUISITOS DE SOFTWARE

Sergio Mogollon [1]
[1] Escuela Profesional de Ciencia de la Computacion
Universidad Nacional de San Agustin de Arequipa
Arequipa, Perú

Proyecto Final de Carrera 1

01

DEFINICION DEL PROBLEMA

Problema

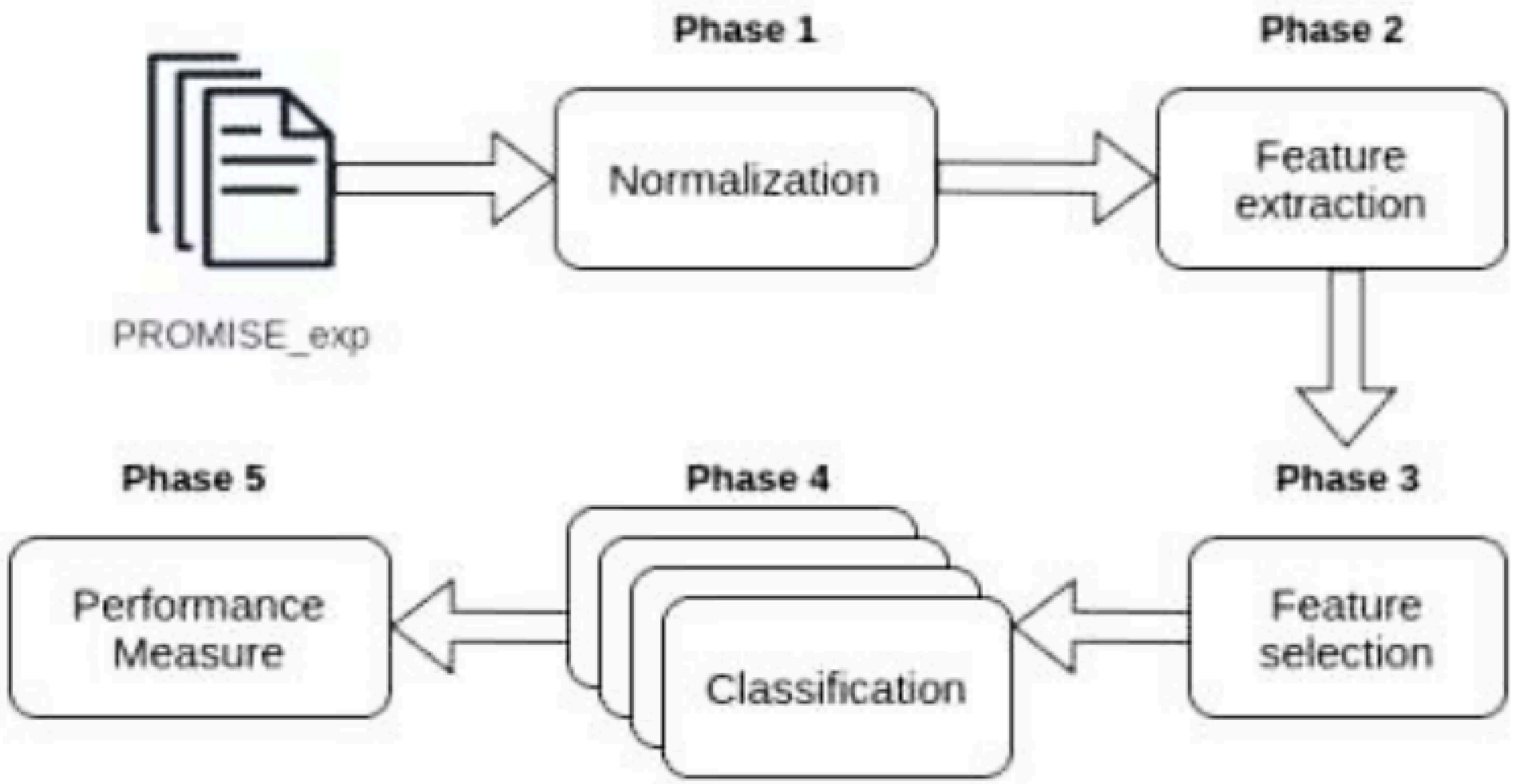
La **naturaleza ambigua e inconsistente del lenguaje natural dificulta la tarea de clasificacion de requisitos de software,** generando interpretaciones variadas y, en consecuencia, problemas de precisión y coherencia en la categorización.

Desafios

- Limitación de conjuntos de datos existentes
- Falta de métricas estándar
- Falta de transparencia en la metodología y ausencia de código

02

PIPELINE


Phase 1
Phase 2
PROMISE_exp
Normalization
Feature extraction
Phase 5
Phase 4
Phase 3
Performance Measure
Classification
Feature selection

03

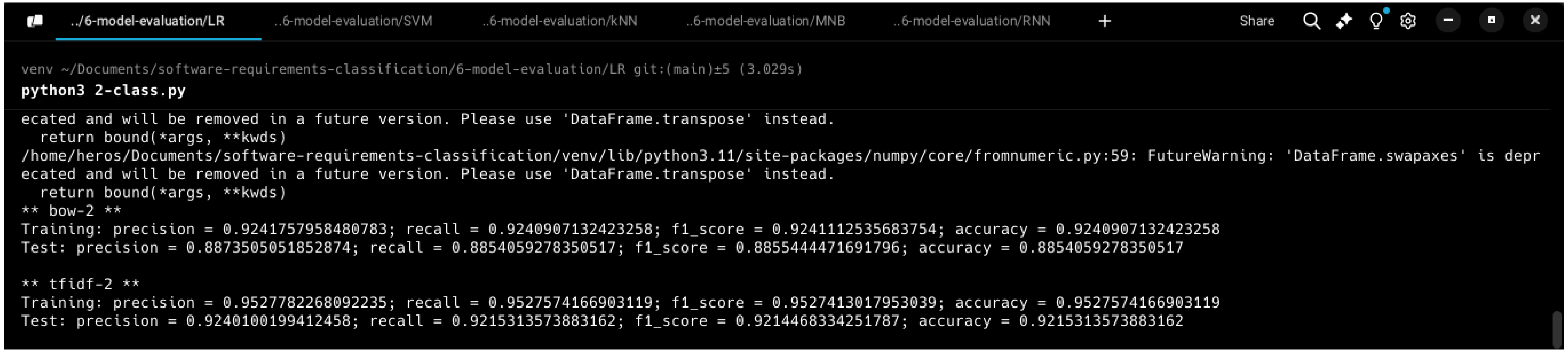
BASES DE DATOS

BASE DE DATOS	NUMERO DE REQUISITOS	NUMERO DE CLASES	CARACTERISTICAS RELEVANTES
PROMISE NFR	625	12 CLASES (255 FR Y 370 NFR)	Es el conjunto de datos mas utilizado en los estudios de RE.
PROMISE_EXP	969	12 CLASES (443 FR Y 526 NFR)	Expansión del dataset PROMISE NFR
SPANISH PROMISE	625	12 CLASES (255 FR Y 370 NFR)	Basado en PROMISE NFR, construida para estudios sobre requisitos en español

04

RESULTADOS

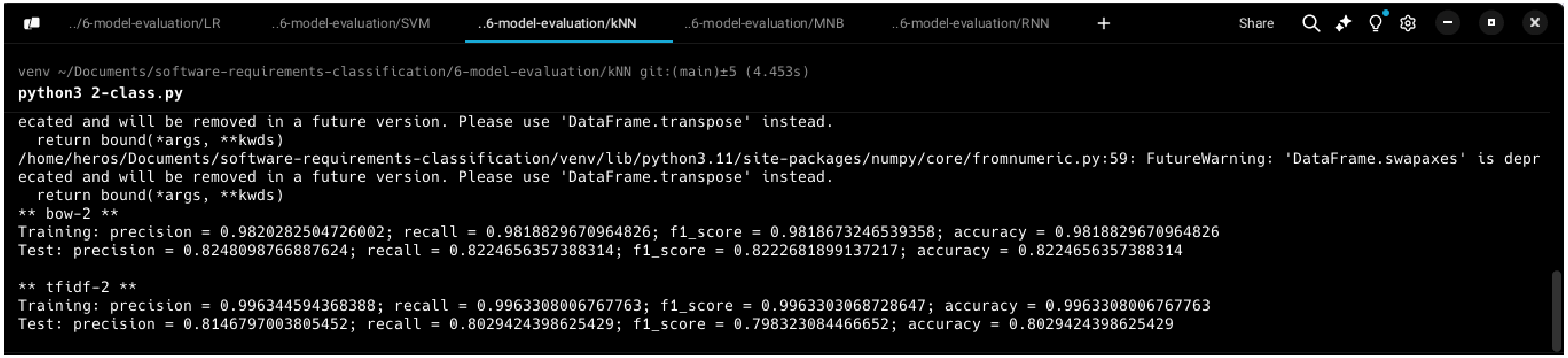
Regresion Logistica (LR)




```
venv ~/Documents/software-requirements-classification/6-model-evaluation/LR git:(main)±5 (3.029s)
python3 2-class.py
ecated and will be removed in a future version. Please use 'DataFrame.transpose' instead.
  return bound(*args, **kwds)
/home/heros/Documents/software-requirements-classification/venv/lib/python3.11/site-packages/numpy/core/fromnumeric.py:59: FutureWarning: 'DataFrame.swapaxes' is depr
ecated and will be removed in a future version. Please use 'DataFrame.transpose' instead.
  return bound(*args, **kwds)
** bow-2 **
Training: precision = 0.9241757958480783; recall = 0.9240907132423258; f1_score = 0.9241112535683754; accuracy = 0.9240907132423258
Test: precision = 0.8873505051852874; recall = 0.8854059278350517; f1_score = 0.8855444471691796; accuracy = 0.8854059278350517

** tfidf-2 **
Training: precision = 0.9527782268092235; recall = 0.9527574166903119; f1_score = 0.9527413017953039; accuracy = 0.9527574166903119
Test: precision = 0.9240100199412458; recall = 0.9215313573883162; f1_score = 0.9214468334251787; accuracy = 0.9215313573883162
```

K-Nearest Neighboard(KNN)




```
venv ~/Documents/software-requirements-classification/6-model-evaluation/kNN git:(main)±5 (4.453s)
python3 2-class.py
ecated and will be removed in a future version. Please use 'DataFrame.transpose' instead.
  return bound(*args, **kwds)
/home/heros/Documents/software-requirements-classification/venv/lib/python3.11/site-packages/numpy/core/fromnumeric.py:59: FutureWarning: 'DataFrame.swapaxes' is depr
ecated and will be removed in a future version. Please use 'DataFrame.transpose' instead.
  return bound(*args, **kwds)
** bow-2 **
Training: precision = 0.9820282504726002; recall = 0.9818829670964826; f1_score = 0.9818673246539358; accuracy = 0.9818829670964826
Test: precision = 0.8248098766887624; recall = 0.8224656357388314; f1_score = 0.8222681899137217; accuracy = 0.8224656357388314

** tfidf-2 **
Training: precision = 0.996344594368388; recall = 0.9963308006767763; f1_score = 0.9963303068728647; accuracy = 0.9963308006767763
Test: precision = 0.8146797003805452; recall = 0.8029424398625429; f1_score = 0.798323084466652; accuracy = 0.8029424398625429
```

04

RESULTADOS

Support Vector Machine (SVM)

```
../6-model-evaluation/LR    ..6-model-evaluation/kNN    ..6-model-evaluation/SVM    ..6-model-evaluation/MNB    ..6-model-evaluation/RNN    +    Share

venv ~/Documents/software-requirements-classification/6-model-evaluation/MNB git:(main)±11 (2.749s)
python3 2-class.py

n matmul
  ret = a @ b
/home/heros/Documents/software-requirements-classification/venv/lib/python3.11/site-packages/sklearn/metrics/_classification.py:1531: UndefinedMetricWarning: Precisio
n is ill-defined and being set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, f"{metric.capitalize()} is", len(result))
** bow-2 **
Training: precision = 0.21002890184031325; recall = 0.4582053868869342; f1_score = 0.28800922894889236; accuracy = 0.4582053868869342
Test: precision = 0.21624643146042205; recall = 0.4582903780068729; f1_score = 0.29203287153409035; accuracy = 0.4582903780068729

** tfidf-2 **
Training: precision = 0.20996358251098773; recall = 0.4582044673539519; f1_score = 0.28796659783068831; accuracy = 0.4582044673539519
Test: precision = 0.21095129372475818; recall = 0.45821520618556705; f1_score = 0.28861401322177127; accuracy = 0.45821520618556705
```

Multinomial Naive Bayes (MNB)

```
../6-model-evaluation/LR    ..6-model-evaluation/kNN    ..6-model-evaluation/SVM    ..6-model-evaluation/MNB    ..6-model-evaluation/RNN    +    Share

venv ~/Documents/software-requirements-classification/6-model-evaluation/MNB git:(main)±13 (2.638s)
python3 2-class.py

ecated and will be removed in a future version. Please use 'DataFrame.transpose' instead.
  return bound(*args, **kwds)
/home/heros/Documents/software-requirements-classification/venv/lib/python3.11/site-packages/numpy/core/fromnumeric.py:59: FutureWarning: 'DataFrame.swapaxes' is depr
ecated and will be removed in a future version. Please use 'DataFrame.transpose' instead.
  return bound(*args, **kwds)
** bow-2 **
Training: precision = 0.9129963447212435; recall = 0.9125102462246601; f1_score = 0.9123084232470393; accuracy = 0.9125102462246601
Test: precision = 0.8973825584766615; recall = 0.8937177835051549; f1_score = 0.8936282457047657; accuracy = 0.8937177835051549

** tfidf-2 **
Training: precision = 0.9535965099040071; recall = 0.9535601689838897; f1_score = 0.9535323632304141; accuracy = 0.9535601689838897
Test: precision = 0.9192283758952895; recall = 0.9184922680412372; f1_score = 0.918440912539031; accuracy = 0.9184922680412372
```

RESULTADOS

LR, KNN, SVM y MNB

TABLE II
RESULTADOS DE CLASIFICACIÓN BINARIA CON BOW Y TF-IDF PARA VARIOS MODELOS DE ML

Modelo	Vectorización	Entrenamiento				Prueba			
		Precisión	Recall	F1-Score	Exactitud	Precisión	Recall	F1-Score	Exactitud
LR	BoW	0.9242	0.9241	0.9241	0.9241	0.8874	0.8854	0.8855	0.8854
	TF-IDF	0.9528	0.9528	0.9527	0.9528	0.9240	0.9215	0.9214	0.9215
KNN	BoW	0.9820	0.9819	0.9819	0.9819	0.8248	0.8225	0.8223	0.8225
	TF-IDF	0.9963	0.9963	0.9963	0.9963	0.8147	0.8029	0.7983	0.8029
SVM	BoW	0.9663	0.9659	0.9660	0.9659	0.8692	0.8668	0.8668	0.8668
	TF-IDF	0.9901	0.9899	0.9899	0.9899	0.9009	0.8999	0.8999	0.8999
MNB	BoW	0.9130	0.9125	0.9123	0.9125	0.8974	0.8937	0.8936	0.8937
	TF-IDF	0.9536	0.9536	0.9535	0.9536	0.9192	0.9185	0.9184	0.9185

RESULTADOS

RNN (W2V + LSTM) y RNN (W2V + GRU)

```
venv ~/Documents/software-requirements-classification/6-model-evaluation/RNN git:(main)±36 (4h 18m 51s)
python3 2-class.py
28/28 ━━━━━━━━━━━━━━━━━━━━ 7s 240ms/step - loss: 0.3749
Epoch 91/100
28/28 ━━━━━━━━━━━━━━━━━━━━ 7s 235ms/step - loss: 0.3360
Epoch 92/100
28/28 ━━━━━━━━━━━━━━━━━━━━ 7s 235ms/step - loss: 0.3264
Epoch 93/100
28/28 ━━━━━━━━━━━━━━━━━━━━ 7s 240ms/step - loss: 0.3383
Epoch 94/100
28/28 ━━━━━━━━━━━━━━━━━━━━ 7s 258ms/step - loss: 0.3381
Epoch 95/100
28/28 ━━━━━━━━━━━━━━━━━━━━ 8s 268ms/step - loss: 0.2970
Epoch 96/100
28/28 ━━━━━━━━━━━━━━━━━━━━ 7s 235ms/step - loss: 0.3366
Epoch 97/100
28/28 ━━━━━━━━━━━━━━━━━━━━ 7s 242ms/step - loss: 0.3881
Epoch 98/100
28/28 ━━━━━━━━━━━━━━━━━━━━ 7s 236ms/step - loss: 0.3635
Epoch 99/100
28/28 ━━━━━━━━━━━━━━━━━━━━ 7s 237ms/step - loss: 0.3204
Epoch 100/100
28/28 ━━━━━━━━━━━━━━━━━━━━ 7s 238ms/step - loss: 0.3560
28/28 ━━━━━━━━━━━━━━━━━━━━ 3s 111ms/step
3/3 ━━━━━━━━━━━━━━━━━━━━ 0s 102ms/step
** 2-w2v-lstm **
Training: precision = 0.9486775567302452 +/- 0.008629155797389649; recall = 0.948288223672457 +/- 0.008933296519537777; f1_score = 0.9482279635103875 +/- 0.009027596714315033; accuracy = 0.948288223672457 +/- 0.008933296519537777
Test: precision = 0.8580150003782571 +/- 0.033055955871817144; recall = 0.8524591924398625 +/- 0.029830784881126345; f1_score = 0.8518945470921631 +/- 0.030201438969365068; accuracy = 0.8524591924398625 +/- 0.029830784881126345

** 2-w2v-gru **
Training: precision = 0.9605378664916253 +/- 0.004551594616551951; recall = 0.9602110196832603 +/- 0.004651645803128308; f1_score = 0.9601789604723583 +/- 0.004683359084519128; accuracy = 0.9602110196832603 +/- 0.004651645803128308
Test: precision = 0.8563507510025798 +/- 0.03342641079145002; recall = 0.8514175257731958 +/- 0.032567107034540084; f1_score = 0.8513426599557075 +/- 0.03220000064051147; accuracy = 0.8514175257731958 +/- 0.032567107034540084
```

04

RESULTADOS

RNN (W2V + LSTM y W2V + GRU)

TABLE III
RESULTADOS DE WORD2VEC + LSTM Y WORD2VEC + GRU

Modelo	Conjunto	Precisión	Recall	F1-Score	Accuracy
w2v + LSTM	Entrenamiento	0.9487 ± 0.0086	0.9483 ± 0.0089	0.9482 ± 0.0090	0.9483 ± 0.0089
	Prueba	0.8580 ± 0.0331	0.8525 ± 0.0298	0.8519 ± 0.0302	0.8525 ± 0.0298
w2v + GRU	Entrenamiento	0.9605 ± 0.0046	0.9602 ± 0.0047	0.9602 ± 0.0047	0.9602 ± 0.0047
	Prueba	0.8564 ± 0.0334	0.8514 ± 0.0326	0.8513 ± 0.0322	0.8514 ± 0.0326

METRICAS

Precision (P)

$$P = \frac{TP}{TP + FP}$$

Recall (R)

$$R = \frac{TP}{TP + FN}$$

F1-Score

$$F1\text{-}Score = 2 \times \frac{P \times R}{P + R}$$

Útil en contextos desbalanceados.

Accuracy (A)

$$A = \frac{TP + TN}{TP + TN + FP + FN}$$

Engañosa en conjuntos de datos desequilibrados

REFERENCIAS

Dias Canedo, E., & Cordeiro Mendes, B. (2020). Software Requirements Classification Using Machine Learning Algorithms. *Entropy*, *22*(9), 1057. https://doi.org/10.3390/e22091057

Kaur, K., Kaur, P. The application of AI techniques in requirements classification: a systematic mapping. Artif Intell Rev 57, 57 (2024). https://doi.org/10.1007/s10462-023-10667-1

MUCHAS GRACIAS

Computer Science