Loading balance-scale 11 .. Accuracy before CAAFE 0.9375 In paper:0.8444 Accuracy after CAAFE 1.0 In paper: 0.882

-----------------------------------------------------------------------------------------------------------------------

Loading breast-w 15 .. Accuracy before CAAFE 0.9444444444444444 In paper: 0.9783 Accuracy after CAAFE 1.0 In paper:0.9809

-----------------------------------------------------------------------------------------------------------------------

Loading cmc 23 .. Accuracy before CAAFE 0.5962059620596206 In paper: 0.7375 Accuracy after CAAFE 0.5907859078590786 In paper:0.7393

Accuracy after CAAFE using RandomForestClassifier 0.5880758807588076

Selected features: ['Wifes\_age', 'Wifes\_education', 'Number\_of\_children\_ever\_born']

Accuracy after CAAFE using Lasso 0.6043360433604336

Selected features: ['Wifes\_education', 'Husbands\_education', 'Wifes\_religion', 'Wifes\_now\_working%3F', 'Husbands\_occupation', 'Standard-of-living\_index', 'Media\_exposure', 'Age\_Children\_Interaction', 'Age\_squared']

Accuracy after CAAFE using Elastic Net 0.6043360433604336

Selected features: ['Wifes\_education', 'Husbands\_education', 'Wifes\_religion', 'Wifes\_now\_working%3F', 'Husbands\_occupation', 'Standard-of-living\_index', 'Media\_exposure', 'Age\_Children\_Interaction', 'Age\_squared']

Accuracy after CAAFE using Mutual Information 0.4173441734417344

Selected features: ['Wifes\_education', 'Husbands\_education', 'Wifes\_religion', 'Wifes\_now\_working%3F']

Accuracy after CAAFE using Boruta 0.43902439024390244

Selected features: ['Wifes\_education', 'Husbands\_education', 'Wifes\_religion', 'Wifes\_now\_working%3F', 'Media\_exposure']

Accuracy after CAAFE using Embedded Methods with Different Classifiers 0.43902439024390244

Selected features: ['Wifes\_education', 'Husbands\_education', 'Wifes\_now\_working%3F']

-----------------------------------------------------------------------------------------------------------------------

Loading credit-g 31 .. Accuracy before CAAFE 0.756 In paper:0.7824

-----------------------------------------------------------------------------------------------------------------------

Loading diabetes 37 .. Accuracy before CAAFE 0.7916666666666666 In paper: 0.8427 Accuracy after CAAFE 0.796875 In paper:0.8434

Accuracy after CAAFE using RandomForestClassifier 0.8072916666666666

Selected features: ['plas', 'pres', 'mass', 'pedi', 'BMI\_age\_interaction', 'Glucose\_Age\_Interaction', 'Age\_Glucose\_Interaction', 'BMI\_squared', 'Preg\_Age\_Product', 'BMI\_AGE\_interaction']

Accuracy after CAAFE using Lasso 0.796875

Selected features: ['preg', 'plas', 'pres', 'skin', 'mass', 'pedi', 'age']

Accuracy after CAAFE using Elastic Net 0.796875

Selected features: ['preg', 'plas', 'pres', 'skin', 'mass', 'pedi', 'age']

Accuracy after CAAFE using Mutual Information 0.8177083333333334

Selected features: ['preg', 'plas', 'mass', 'age']

Accuracy after CAAFE using Embedded Methods with Different Classifiers 0.8125

Selected features: ['preg', 'plas', 'mass', 'pedi', 'age']

Accuracy after CAAFE using Boruta 0.8072916666666666

Selected features: ['plas', 'mass', 'age']

-----------------------------------------------------------------------------------------------------------------------

Loading tic-tac-toe 50 .. Accuracy before CAAFE 0.7916666666666666 in paper: 0.6989 Accuracy after CAAFE 1.0 In paper: 0.9536

--------------------------------------------------------------------------------------------- --------------------------

Loading eucalyptus 188 .. Accuracy before CAAFE 0.7228260869565217 In paper: 0.9319 Accuracy after CAAFE 0.7336956521739131 In paper:0.9319

Accuracy after CAAFE using RandomForestClassifier 0.7336956521739131

Selected features: ['Rep', 'Locality', 'Map\_Ref', 'Latitude', 'Altitude', 'Rainfall', 'Frosts', 'Year', 'Sp', 'PMCno', 'DBH', 'Ht', 'Surv', 'Vig', 'Ins\_res', 'Stem\_Fm', 'Crown\_Fm', 'Brnch\_Fm', 'Rainfall\_Frosts\_Interaction', 'Rainfall\_Altitude\_Ratio', 'Rainfall\_Frosts\_Ratio', 'DBH\_Ht\_Product', 'Surv\_Vig\_Product', 'Year\_Altitude\_Interaction', 'Altitude\_Rainfall\_Ratio']

-----------------------------------------------------------------------------------------------------------------------

Loading pc1 1068 .. In paper:0.9035

-----------------------------------------------------------------------------------------------------------------------

Loading airlines 1169 .. In paper: 0.6211 after CAAFE:0.6203

-----------------------------------------------------------------------------------------------------------------------

Loading jungle\_chess\_2pcs\_raw\_endgame\_complete 41027 . In paper:0.9334

Loading cmc 23 ..

Accuracy before CAAFE 0.5962059620596206

In paper: 0.7375

Accuracy after CAAFE 0.5907859078590786

In paper:0.7393

\*Dataset description:\*

4. Relevant Information:

This dataset is a subset of the 1987 National Indonesia Contraceptive

Prevalence Survey. The samples are married women who were either not

pregnant or do not know if they were at the time of interview. The

problem is to predict the current contraceptive method choice

(no use, long-term methods, or short-term methods) of a woman based

on her demographic and socio-economic characteristics.

7. Attribute Information:

1. Wife's age (numerical)

2. Wife's education (categorical) 1=low, 2, 3, 4=high

3. Husband's education (categorical) 1=low, 2, 3, 4=high

4. Number of children ever born (numerical)

5. Wife's religion (binary) 0=Non-Islam, 1=Islam

6. Wife's now working? (binary) 0=Yes, 1=No

7. Husband's occupation (categorical) 1, 2, 3, 4

8. Standard-of-living index (categorical) 1=low, 2, 3, 4=high

9. Media exposure (binary) 0=Good, 1=Not good

10. Contraceptive method used (class attribute) 1=No-use

2=Long-term

3=Short-term

all\_features\_dictionary

{'Age\_Education': 'Wifes\_age \* Wifes\_education',

'Education\_Difference': 'Husbands\_education - Wifes\_education',

'Children\_Age': 'Number\_of\_children\_ever\_born \* Wifes\_age',

'Working\_Occupation': 'Wifes\_now\_working%3F \* Husbands\_occupation',

'Religion\_Education': 'Wifes\_religion \* Wifes\_education',

'Age\_Children\_Interaction': 'Wifes\_age \* Number\_of\_children\_ever\_born', 'Wifes\_age\_squared': 'Wifes\_age\*\*2',

'Combined\_education': 'Wifes\_education + Husbands\_education', 'Occupation\_living\_interaction': 'Husbands\_occupation \* Standard-of-living\_index', 'Working\_media\_interaction': 'Wifes\_now\_working%3F \* Media\_exposure'}

only\_selected\_features\_dictionary

{'Age\_Children\_Interaction': 'Wifes\_age \* Number\_of\_children\_ever\_born'}

"Baseline Features: ['Wifes\_age', 'Wifes\_education', 'Husbands\_education', 'Number\_of\_children\_ever\_born', 'Wifes\_religion', 'Wifes\_now\_working%3F', 'Husbands\_occupation', 'Standard-of-living\_index', 'Media\_exposure']

Generated Features (All): ['Wifes\_age', 'Wifes\_education', 'Husbands\_education', 'Number\_of\_children\_ever\_born', 'Wifes\_religion', 'Wifes\_now\_working%3F', 'Husbands\_occupation', 'Standard-of-living\_index', 'Media\_exposure', 'Age\_Education', 'Education\_Difference', 'Children\_Age', 'Working\_Occupation', 'Religion\_Education', 'Age\_Children\_Interaction', 'Wifes\_age\_squared', 'Combined\_education', 'Occupation\_living\_interaction', 'Working\_media\_interaction']

Feature selection:

Applying feature selection using RFE...

Selected features (RFE): ['Wifes\_age', 'Number\_of\_children\_ever\_born', 'Standard-of-living\_index', 'Age\_Education', 'Children\_Age', 'Working\_Occupation', 'Age\_Children\_Interaction', 'Wifes\_age\_squared', 'Combined\_education', 'Occupation\_living\_interaction']

TabPFNClassifier Accuracy (RFE): 0.5880758807588076

Applying feature selection using LassoCV...

Selected features (LassoCV): ['Wifes\_age', 'Number\_of\_children\_ever\_born', 'Wifes\_religion', 'Wifes\_now\_working%3F', 'Standard-of-living\_index', 'Media\_exposure', 'Age\_Education', 'Education\_Difference', 'Children\_Age', 'Working\_Occupation', 'Religion\_Education', 'Age\_Children\_Interaction', 'Wifes\_age\_squared', 'Occupation\_living\_interaction', 'Working\_media\_interaction']

TabPFNClassifier Accuracy (LassoCV): 0.6043360433604336

Applying feature selection using ElasticNetCV...

Selected features (ElasticNetCV): ['Wifes\_age', 'Wifes\_education', 'Number\_of\_children\_ever\_born', 'Wifes\_religion', 'Wifes\_now\_working%3F', 'Standard-of-living\_index', 'Media\_exposure', 'Age\_Education', 'Education\_Difference', 'Children\_Age', 'Working\_Occupation', 'Religion\_Education', 'Age\_Children\_Interaction', 'Wifes\_age\_squared', 'Occupation\_living\_interaction', 'Working\_media\_interaction']

TabPFNClassifier Accuracy (ElasticNetCV): 0.5989159891598916

Applying feature selection using Mutual Information...

Selected features (Mutual Information): ['Wifes\_education', 'Number\_of\_children\_ever\_born', 'Husbands\_occupation', 'Media\_exposure', 'Age\_Education', 'Education\_Difference', 'Children\_Age', 'Age\_Children\_Interaction', 'Combined\_education']

TabPFNClassifier Accuracy (Mutual Information): 0.5907859078590786

Applying feature selection using Boruta...

Selected features (Boruta): ['Wifes\_age', 'Wifes\_education', 'Number\_of\_children\_ever\_born', 'Husbands\_occupation', 'Age\_Education', 'Children\_Age', 'Age\_Children\_Interaction', 'Wifes\_age\_squared', 'Combined\_education']

TabPFNClassifier Accuracy (Boruta): 0.5962059620596206

Applying feature selection using Gradient Boosting...

Selected features (Gradient Boosting): ['Wifes\_age', 'Number\_of\_children\_ever\_born', 'Age\_Education', 'Children\_Age', 'Age\_Children\_Interaction', 'Wifes\_age\_squared', 'Combined\_education']

TabPFNClassifier Accuracy (Gradient Boosting): 0.6070460704607046

Applying feature selection using Decision Tree...

Selected features (Decision Tree): ['Standard-of-living\_index', 'Age\_Education', 'Children\_Age', 'Age\_Children\_Interaction', 'Wifes\_age\_squared', 'Occupation\_living\_interaction']

TabPFNClassifier Accuracy (Decision Tree): 0.6016260162601627

Causality + feature selection

Causality Selected Features: ['Wifes\_education', 'Husbands\_education', 'Media\_exposure', 'Working\_media\_interaction']

TabPFNClassifier Accuracy (causality): 0.4146341463414634

Applying feature selection using RFE...

Selected features (RFE): ['Wifes\_education', 'Husbands\_education', 'Media\_exposure', 'Working\_media\_interaction']

TabPFNClassifier Accuracy (RFE): 0.4146341463414634

Applying feature selection using LassoCV...

Selected features (LassoCV): ['Wifes\_education', 'Husbands\_education', 'Media\_exposure']

TabPFNClassifier Accuracy (LassoCV): 0.43902439024390244

Applying feature selection using ElasticNetCV...

Selected features (ElasticNetCV): ['Wifes\_education', 'Husbands\_education', 'Media\_exposure']

TabPFNClassifier Accuracy (ElasticNetCV): 0.43902439024390244

Applying feature selection using Mutual Information...

Selected features (Mutual Information): ['Husbands\_education', 'Working\_media\_interaction']

TabPFNClassifier Accuracy (Mutual Information): 0.4146341463414634

Applying feature selection using Boruta...

Selected features (Boruta): ['Wifes\_education', 'Husbands\_education']

TabPFNClassifier Accuracy (Boruta): 0.43902439024390244

Applying feature selection using Gradient Boosting...

Selected features (Gradient Boosting): ['Wifes\_education']

TabPFNClassifier Accuracy (Gradient Boosting): 0.44173441734417346

Applying feature selection using Decision Tree...

Selected features (Decision Tree): ['Wifes\_education', 'Husbands\_education']

TabPFNClassifier Accuracy (Decision Tree): 0.43902439024390244

feature selection + Causality

Applying feature selection using RFE...

No results

Applying feature selection using LassoCV...

Selected features (LassoCV): ['Media\_exposure', 'Working\_media\_interaction']

TabPFNClassifier Accuracy (LassoCV): 0.4146341463414634

Applying feature selection using ElasticNetCV...

Selected features (ElasticNetCV): ['Wifes\_education', 'Media\_exposure', 'Working\_media\_interaction']

TabPFNClassifier Accuracy (ElasticNetCV): 0.44173441734417346

Applying feature selection using Mutual Information...

Selected features (Mutual Information): ['Wifes\_education']

TabPFNClassifier Accuracy (Mutual Information): 0.44173441734417346

Applying feature selection using Boruta...

Selected features (Boruta): ['Wifes\_education']

TabPFNClassifier Accuracy (Boruta): 0.44173441734417346

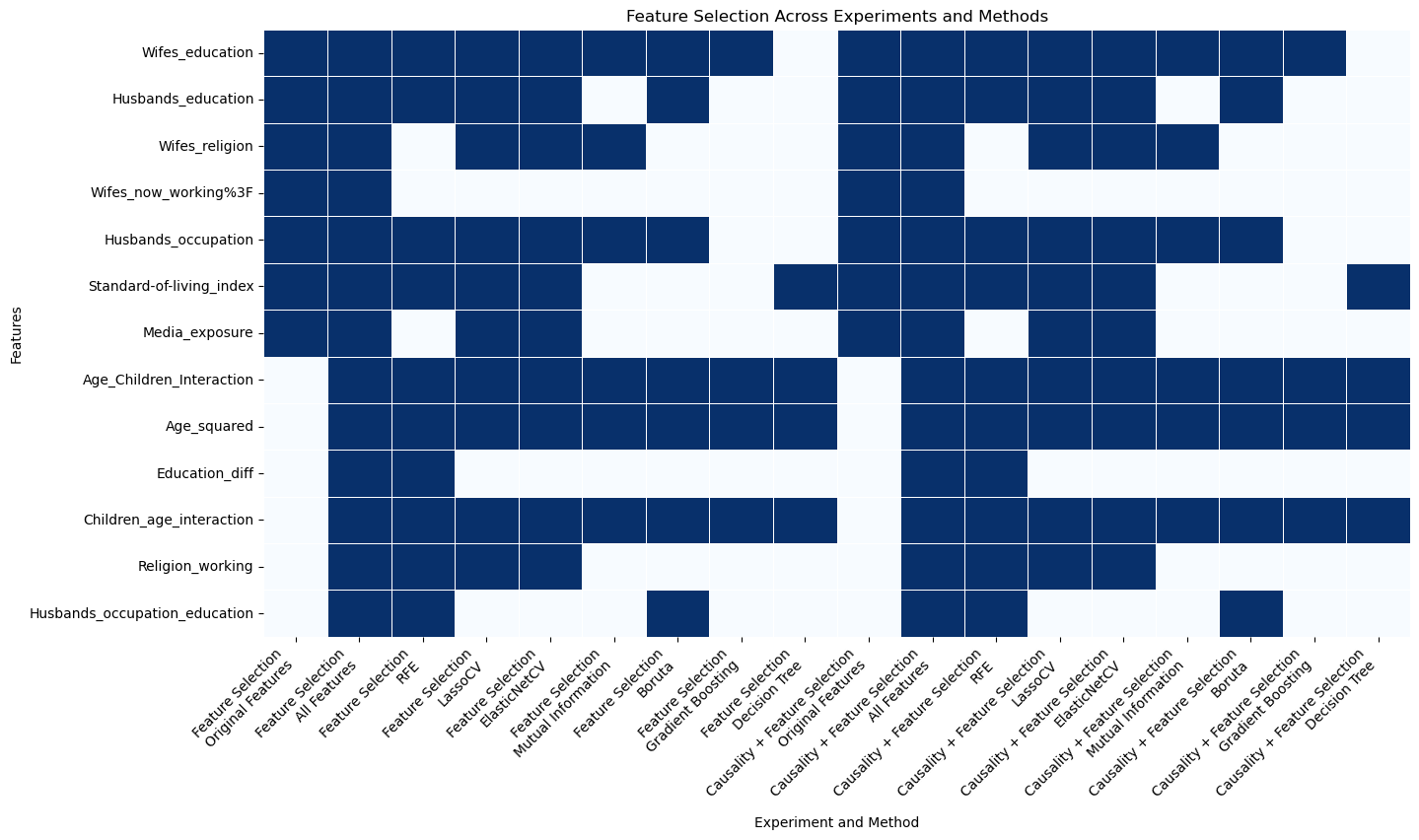
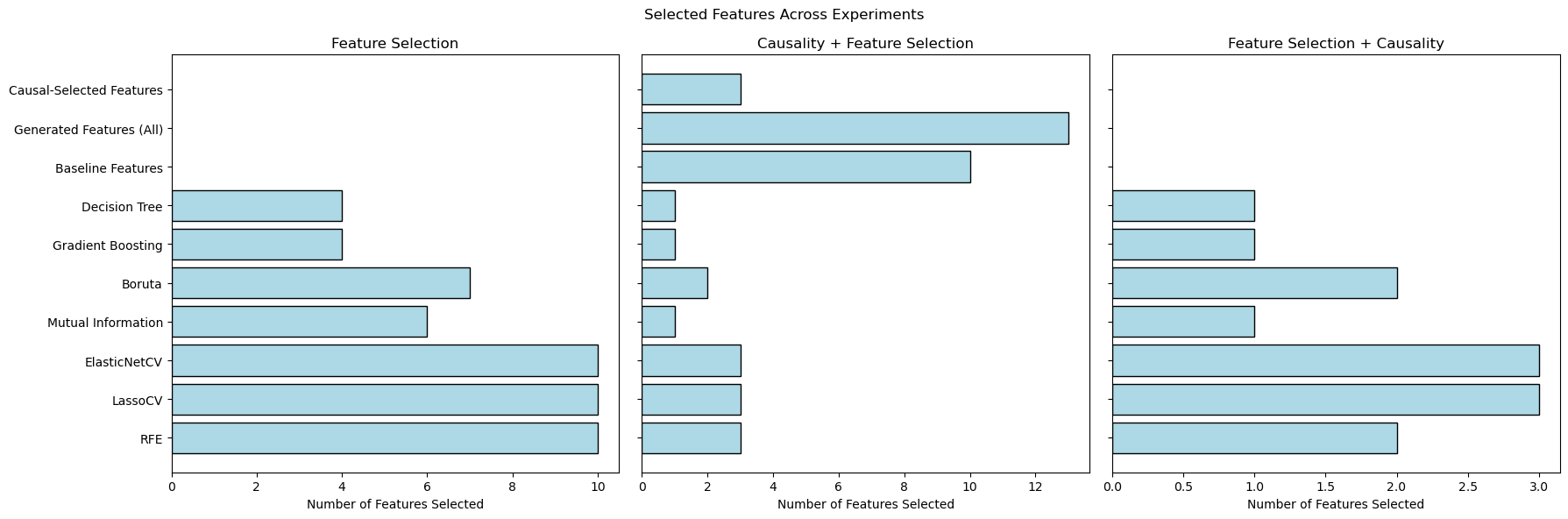
Applying feature selection using Gradient Boosting...

No results

Applying feature selection using Decision Tree…

No results

No results means after causality there are no selected features



Loading diabetes 37 ..

Accuracy before CAAFE 0.7916666666666666

In paper: 0.8427

Accuracy after CAAFE 0.796875

In paper:0.8434

Feature selection:

Applying feature selection using RFE...

Selected features (RFE): ['plas', 'mass', 'pedi', 'age', 'BMI\_age\_interaction', 'Glucose\_Age\_Interaction', 'Age\_Glucose\_Interaction', 'BMI\_squared', 'Insulin\_Skin\_Ratio', 'Preg\_Age\_Product']

TabPFNClassifier Accuracy (RFE): 0.8072916666666666

Applying feature selection using LassoCV...

Selected features (LassoCV): ['preg', 'plas', 'pres', 'skin', 'pedi', 'age', 'BMI\_age\_interaction', 'plas\_insu', 'Age\_squared', 'BMI\_squared', 'Insulin\_Skin\_Ratio', 'Preg\_Age\_Product']

TabPFNClassifier Accuracy (LassoCV): 0.796875

Applying feature selection using ElasticNetCV...

Selected features (ElasticNetCV): ['preg', 'plas', 'pres', 'skin', 'pedi', 'age', 'BMI\_age\_interaction', 'plas\_insu', 'Age\_squared', 'BMI\_squared', 'Insulin\_Skin\_Ratio', 'Preg\_Age\_Product']

TabPFNClassifier Accuracy (ElasticNetCV): 0.796875

Applying feature selection using Mutual Information...

Selected features (Mutual Information): ['plas', 'age', 'BMI\_age\_interaction', 'Glucose\_Age\_Interaction', 'Age\_squared', 'Age\_Glucose\_Interaction', 'Preg\_Age\_Product']

TabPFNClassifier Accuracy (Mutual Information): 0.8177083333333334

Applying feature selection using Boruta...

Selected features (Boruta): ['plas', 'mass', 'pedi', 'age', 'BMI\_age\_interaction', 'Glucose\_Age\_Interaction', 'Age\_squared', 'Age\_Glucose\_Interaction', 'BMI\_squared', 'Preg\_Age\_Product']

TabPFNClassifier Accuracy (Boruta): 0.8072916666666666

Applying feature selection using Gradient Boosting...

Selected features (Gradient Boosting): ['plas', 'mass', 'pedi', 'BMI\_age\_interaction', 'Glucose\_Age\_Interaction', 'Age\_Glucose\_Interaction']

TabPFNClassifier Accuracy (Gradient Boosting): 0.8072916666666666

Applying feature selection using Decision Tree...

Selected features (Decision Tree): ['plas', 'mass', 'pedi', 'BMI\_age\_interaction', 'Glucose\_Age\_Interaction']

TabPFNClassifier Accuracy (Decision Tree): 0.8125

Best selected features: ['plas', 'mass', 'pedi', 'age', 'BMI\_age\_interaction', 'Glucose\_Age\_Interaction', 'Age\_squared', 'Age\_Glucose\_Interaction', 'BMI\_squared', 'Preg\_Age\_Product']

Accuracy after CAAFE 0.8072916666666666

Causality + feature selection:

All Features: ['preg', 'plas', 'pres', 'skin', 'mass', 'pedi', 'age', 'BMI\_age\_interaction', 'Glucose\_Age\_Interaction', 'plas\_insu', 'Age\_squared', 'Age\_Glucose\_Interaction', 'BMI\_squared', 'Insulin\_Skin\_Ratio', 'Preg\_Age\_Product']

Causality Selected Features: ['pedi']

TabPFNClassifier Accuracy (causality): 0.6302083333333334

feature selection + Causality:

Applying feature selection using RFE...

Selected features (RFE): ['pedi']

TabPFNClassifier Accuracy (RFE): 0.6302083333333334

Applying feature selection using LassoCV...

Selected features (LassoCV): ['pedi']

TabPFNClassifier Accuracy (LassoCV): 0.6302083333333334

Applying feature selection using LassoCV...

Selected features (LassoCV): ['pedi']

TabPFNClassifier Accuracy (LassoCV): 0.6302083333333334

Selected features (LassoCV): ['pedi']

TabPFNClassifier Accuracy (LassoCV): 0.6302083333333334

Applying feature selection using ElasticNetCV...

Applying feature selection using ElasticNetCV...

Selected features (ElasticNetCV): ['pedi']

TabPFNClassifier Accuracy (ElasticNetCV): 0.6302083333333334

Applying feature selection using Mutual Information...

Applying feature selection using Boruta...

Selected features (Boruta): ['pedi']

TabPFNClassifier Accuracy (Boruta): 0.6302083333333334

Applying feature selection using Gradient Boosting...

Selected features (Gradient Boosting): ['pedi']

TabPFNClassifier Accuracy (Gradient Boosting): 0.6302083333333334

Applying feature selection using Decision Tree...

Selected features (Decision Tree): ['pedi']

TabPFNClassifier Accuracy (Decision Tree): 0.6302083333333334

Loading eucalyptus 188 ..

Accuracy before CAAFE 0.7228260869565217

In paper: 0.9319

Accuracy after CAAFE 0.7336956521739131

In paper:0.9319

Baseline Features:['Abbrev', 'Rep', 'Locality', 'Map\_Ref', 'Latitude', 'Altitude',

'Rainfall', 'Frosts', 'Year', 'Sp', 'PMCno', 'DBH', 'Ht', 'Surv', 'Vig',

'Ins\_res', 'Stem\_Fm', 'Crown\_Fm', 'Brnch\_Fm', 'Utility']

Generated Features (All): ['Rep', 'Locality', 'Map\_Ref', 'Latitude', 'Altitude', 'Rainfall', 'Frosts', 'Year', 'Sp', 'PMCno', 'DBH', 'Ht', 'Surv', 'Vig', 'Ins\_res', 'Stem\_Fm', 'Crown\_Fm', 'Brnch\_Fm', 'Rainfall\_Frosts\_Interaction', 'Rainfall\_Altitude\_Ratio', 'Rainfall\_Frosts\_Ratio', 'DBH\_Ht\_Product', 'Surv\_Vig\_Product', 'Year\_Altitude\_Interaction', 'Altitude\_Rainfall\_Ratio']

Feature selection:

Applying feature selection using RFE...

Selected features (RFE): ['Map\_Ref', 'Latitude', 'PMCno', 'DBH', 'Ht', 'Surv', 'Vig', 'Crown\_Fm', 'DBH\_Ht\_Product', 'Surv\_Vig\_Product']

TabPFNClassifier Accuracy (RFE): 0.717391304347826

Applying feature selection using LassoCV...

Selected features (LassoCV): ['Rep', 'Locality', 'Map\_Ref', 'Latitude', 'Altitude', 'Rainfall', 'Frosts', 'Year', 'Sp', 'PMCno', 'DBH', 'Ht', 'Surv', 'Vig', 'Ins\_res', 'Stem\_Fm', 'Crown\_Fm', 'Brnch\_Fm', 'Rainfall\_Frosts\_Interaction', 'Rainfall\_Altitude\_Ratio', 'Rainfall\_Frosts\_Ratio', 'DBH\_Ht\_Product', 'Surv\_Vig\_Product', 'Altitude\_Rainfall\_Ratio']

TabPFNClassifier Accuracy 0.717391304347826

Applying feature selection using ElasticNetCV...

Selected features (ElasticNetCV): ['Rep', 'Locality', 'Map\_Ref', 'Latitude', 'Altitude', 'Frosts', 'Year', 'Sp', 'PMCno', 'DBH', 'Ht', 'Surv', 'Vig', 'Ins\_res', 'Stem\_Fm', 'Crown\_Fm', 'Brnch\_Fm', 'Rainfall\_Frosts\_Interaction', 'Rainfall\_Altitude\_Ratio', 'Rainfall\_Frosts\_Ratio', 'Surv\_Vig\_Product', 'Year\_Altitude\_Interaction', 'Altitude\_Rainfall\_Ratio']

TabPFNClassifier Accuracy (ElasticNetCV): 0.717391304347826

Applying feature selection using Mutual Information...

Selected features (Mutual Information): ['Map\_Ref', 'Latitude', 'Rainfall', 'PMCno', 'Ht', 'Vig', 'Ins\_res', 'Stem\_Fm', 'Crown\_Fm', 'Rainfall\_Frosts\_Interaction', 'Surv\_Vig\_Product', 'Year\_Altitude\_Interaction']

TabPFNClassifier Accuracy (Mutual Information): 0.717391304347826

Applying feature selection using Boruta...

Selected features (Boruta): ['Locality', 'Map\_Ref', 'Latitude', 'Rainfall', 'Year', 'PMCno', 'DBH', 'Ht', 'Surv', 'Vig', 'Ins\_res', 'Stem\_Fm', 'Crown\_Fm', 'Brnch\_Fm', 'Rainfall\_Frosts\_Interaction', 'Rainfall\_Frosts\_Ratio', 'DBH\_Ht\_Product', 'Surv\_Vig\_Product', 'Year\_Altitude\_Interaction', 'Altitude\_Rainfall\_Ratio']

TabPFNClassifier Accuracy (Boruta): 0.7282608695652174

Applying feature selection using Gradient Boosting...

Selected features (Gradient Boosting): ['Latitude', 'Year', 'Ht', 'Vig', 'Stem\_Fm', 'Crown\_Fm', 'DBH\_Ht\_Product', 'Surv\_Vig\_Product']

TabPFNClassifier Accuracy (Gradient Boosting): 0.7282608695652174

Applying feature selection using Decision Tree...

Selected features (Decision Tree): ['Latitude', 'PMCno', 'DBH', 'Ht', 'Surv', 'Vig', 'Ins\_res', 'Stem\_Fm', 'Crown\_Fm', 'Brnch\_Fm', 'Surv\_Vig\_Product']

TabPFNClassifier Accuracy (Decision Tree): 0.7336956521739131

Causality + feature selection:

Causal-Selected Features: ['Locality', 'Latitude', 'Year', 'Ht', 'Vig', 'Ins\_res', 'Stem\_Fm', 'Crown\_Fm', 'Brnch\_Fm', 'Altitude\_Rainfall\_Ratio']

TabPFNClassifier Accuracy (causality): 0.6902173913043478

Applying feature selection using RFE...

Selected features (RFE): ['Locality', 'Latitude', 'Year', 'Ht', 'Vig', 'Ins\_res', 'Stem\_Fm', 'Crown\_Fm', 'Brnch\_Fm', 'Altitude\_Rainfall\_Ratio']

TabPFNClassifier Accuracy (RFE): 0.6902173913043478

Applying feature selection using LassoCV...

Selected features (LassoCV): ['Locality', 'Latitude', 'Year', 'Ht', 'Vig', 'Ins\_res', 'Stem\_Fm', 'Crown\_Fm', 'Brnch\_Fm', 'Altitude\_Rainfall\_Ratio']

TabPFNClassifier Accuracy (LassoCV): 0.6902173913043478

Applying feature selection using ElasticNetCV...

Selected features (ElasticNetCV): ['Locality', 'Latitude', 'Year', 'Ht', 'Vig', 'Ins\_res', 'Stem\_Fm', 'Crown\_Fm', 'Brnch\_Fm', 'Altitude\_Rainfall\_Ratio']

TabPFNClassifier Accuracy (ElasticNetCV): 0.6902173913043478

Applying feature selection using Mutual Information...

Selected features (Mutual Information): ['Latitude', 'Vig', 'Ins\_res', 'Stem\_Fm', 'Crown\_Fm']

TabPFNClassifier Accuracy (Mutual Information): 0.6358695652173914

Applying feature selection using Boruta...

Selected features (Boruta): ['Locality', 'Latitude', 'Year', 'Ht', 'Vig', 'Ins\_res', 'Stem\_Fm', 'Crown\_Fm', 'Brnch\_Fm', 'Altitude\_Rainfall\_Ratio']

TabPFNClassifier Accuracy (Boruta): 0.6902173913043478

Applying feature selection using Gradient Boosting...

Selected features (Gradient Boosting): ['Ht', 'Vig']

TabPFNClassifier Accuracy (Gradient Boosting): 0.6032608695652174

Applying feature selection using Decision Tree...

Selected features (Decision Tree): ['Latitude', 'Ht', 'Vig', 'Stem\_Fm']

TabPFNClassifier Accuracy (Decision Tree): 0.6630434782608695

feature selection + Causality:

Selected features (RFE): ['Latitude', 'Ht', 'Vig', 'Crown\_Fm']

TabPFNClassifier Accuracy (RFE): 0.6684782608695652

Applying feature selection using LassoCV...

Selected features (LassoCV): ['Locality', 'Latitude', 'Year', 'Ht', 'Vig', 'Ins\_res', 'Stem\_Fm', 'Crown\_Fm', 'Brnch\_Fm', 'Altitude\_Rainfall\_Ratio']

TabPFNClassifier Accuracy (LassoCV): 0.6902173913043478

Applying feature selection using ElasticNetCV...

Selected features (RFE): ['Latitude', 'Ht', 'Vig', 'Crown\_Fm']

TabPFNClassifier Accuracy (RFE): 0.6684782608695652

Applying feature selection using LassoCV...

Selected features (LassoCV): ['Locality', 'Latitude', 'Year', 'Ht', 'Vig', 'Ins\_res', 'Stem\_Fm', 'Crown\_Fm', 'Brnch\_Fm', 'Altitude\_Rainfall\_Ratio']

TabPFNClassifier Accuracy (LassoCV): 0.6902173913043478

Applying feature selection using ElasticNetCV...

Selected features (ElasticNetCV): ['Locality', 'Latitude', 'Year', 'Ht', 'Vig', 'Ins\_res', 'Stem\_Fm', 'Crown\_Fm', 'Brnch\_Fm', 'Altitude\_Rainfall\_Ratio']

TabPFNClassifier Accuracy (ElasticNetCV): 0.6902173913043478

Applying feature selection using Mutual Information...

Selected features (Mutual Information): ['Latitude', 'Ht', 'Vig', 'Ins\_res', 'Stem\_Fm', 'Crown\_Fm']

TabPFNClassifier Accuracy (Mutual Information): 0.6521739130434783

Applying feature selection using Boruta...

Selected features (Boruta): ['Locality', 'Latitude', 'Year', 'Ht', 'Vig', 'Ins\_res', 'Stem\_Fm', 'Crown\_Fm', 'Brnch\_Fm', 'Altitude\_Rainfall\_Ratio']

TabPFNClassifier Accuracy (Boruta): 0.6902173913043478

Applying feature selection using Gradient Boosting...

Selected features (Gradient Boosting): ['Latitude', 'Year', 'Ht', 'Vig', 'Stem\_Fm', 'Crown\_Fm']

TabPFNClassifier Accuracy (Gradient Boosting): 0.6902173913043478

Applying feature selection using Decision Tree...

Selected features (Decision Tree): ['Latitude', 'Ht', 'Vig', 'Ins\_res', 'Stem\_Fm', 'Crown\_Fm', 'Brnch\_Fm']

TabPFNClassifier Accuracy (Decision Tree): 0.657608695652174

Loading airlines 1169 ..

Accuracy before CAAFE :0.642

In paper: 0.6211

Accuracy after CAAFE:0.6203

In paper: 6203

all\_features\_dictionary:

{'TotalTimeOfWeek': 'DayOfWeek \* 24 \* 60 + Time',

'FlightDurationRatio': 'Length / Time',

'AirportTraffic': 'AirportFrom + AirportTo',

'combined\_airport': "AirportFrom.astype(str) + '\_' + AirportTo.astype(str)",

'week': 'DayOfWeek.isin([5, 6]).astype(int)',

'Time\_of\_Day': "pd.cut(Time, bins=[0, 600, 1200, 1800, 2400], labels=['Night', 'Morning', 'Afternoon', 'Evening'])"}

only\_selected\_features\_dictionary

{'combined\_airport': "AirportFrom.astype(str) + '\_' + AirportTo.astype(str)"}

Generated Features (All): ['Airline', 'Flight', 'AirportFrom', 'AirportTo', 'DayOfWeek', 'Time', 'Length', 'Week', 'FlightTime\_hr']

['Airline', 'Flight', 'AirportFrom', 'AirportTo', 'DayOfWeek', 'Time', 'Length', 'combined\_airport']

Baseline Features: ['Airline', 'Flight', 'AirportFrom', 'AirportTo', 'DayOfWeek', 'Time', 'Length']

Feature selection:

Applying feature selection using RFE...

Selected features (RFE): ['Airline', 'Flight', 'AirportFrom', 'AirportTo', 'Time', 'Length', 'TotalTimeOfWeek', 'FlightDurationRatio', 'AirportTraffic', 'combined\_airport']

TabPFNClassifier Accuracy (RFE): 0.642

Applying feature selection using LassoCV...

Selected features (LassoCV): ['Airline', 'Flight', 'AirportFrom', 'AirportTo', 'DayOfWeek', 'Time', 'Length', 'combined\_airport', 'week', 'Time\_of\_Day']

TabPFNClassifier Accuracy (LassoCV): 0.648

Applying feature selection using ElasticNetCV...

Selected features (ElasticNetCV): ['Airline', 'Flight', 'AirportFrom', 'AirportTo', 'DayOfWeek', 'Time', 'Length', 'combined\_airport', 'week', 'Time\_of\_Day']

TabPFNClassifier Accuracy (ElasticNetCV): 0.648

Applying feature selection using Mutual Information...

Selected features (Mutual Information): ['Airline', 'AirportFrom', 'TotalTimeOfWeek', 'FlightDurationRatio', 'combined\_airport', 'Time\_of\_Day']

TabPFNClassifier Accuracy (Mutual Information): 0.642

Applying feature selection using Mutual Information...

Selected features (Mutual Information): ['Airline', 'AirportFrom', 'TotalTimeOfWeek', 'FlightDurationRatio', 'combined\_airport', 'Time\_of\_Day']

TabPFNClassifier Accuracy (Mutual Information): 0.642

Applying feature selection using Boruta...

Selected features (Boruta): ['Airline', 'Time', 'FlightDurationRatio']

f\_Day']

TabPFNClassifier Accuracy (Boruta): 0.652

Applying feature selection using Gradient Boosting...

Selected features (Gradient Boosting): ['Airline', 'Flight', 'Time', 'TotalTimeOfWeek', 'FlightDurationRatio', 'combined\_airport']

TabPFNClassifier Accuracy (Gradient Boosting): 0.64

Applying feature selection using Decision Tree...

Selected features (Decision Tree): ['Airline', 'Flight', 'AirportTo', 'Time', 'Length', 'TotalTimeOfWeek', 'FlightDurationRatio', 'combined\_airport']

TabPFNClassifier Accuracy (Decision Tree): 0.648

Causality + feature selection:

Causality Selected Features: ['Time\_of\_Day']

TabPFNClassifier Accuracy (causality): 0.55

Cant do feature selection from one feature

Causal-selected Features:

feature selection + Causality:

Applying feature selection using RFE...

Applying feature selection using LassoCV...

Selected features (LassoCV): ['Time\_of\_Day']

TabPFNClassifier Accuracy (LassoCV): 0.55

Applying feature selection using ElasticNetCV...

Selected features (ElasticNetCV): ['Time\_of\_Day']

TabPFNClassifier Accuracy (ElasticNetCV): 0.55

Applying feature selection using Mutual Information...

Selected features (Mutual Information): ['Time\_of\_Day']

TabPFNClassifier Accuracy (Mutual Information): 0.55

Applying feature selection using Boruta...

Applying feature selection using Gradient Boosting...

Applying feature selection using Decision Tree...

**Enhancing Feature Engineering with Large Language Models through Causal Feature Engineering & a Self-Improving Mechanism**

**By:** Yuval Saadaty

## **Abstract**

Feature engineering is a crucial but challenging step in the data science pipeline, often requiring domain knowledge and extensive manual effort. This research aims to improve the feature engineering process by leveraging **LLM knowledge** instead of human domain expertise. By incorporating **causal feature engineering** and a **self-improving mechanism**, we enhance the automated generation of meaningful features.

Our approach automates feature generation using the prompt shown in the paper *"Large Language Models for Automated Data Science: Introducing CAAFE for Context-Aware Automated Feature Engineering."* However, in our method, we generate new features **over 10 epochs**, where in each epoch, a new feature is generated. We then **apply causality analysis** followed by **feature selection** to refine the feature set.

Showing that in these experiments, **adding causality either before or after feature selection resulted in a decrease in accuracy**. However, this suggests that while **causal filtering may remove some predictive features, it can potentially improve model generalization** by focusing on features with true causal impact rather than spurious correlations.

Need to add: the classifier version they used is not the same like i used so i evaluate the their model and than my model

## **Problem Description**

### **What Element in the DS Pipeline Are We Improving?**

Feature engineering, the process of creating meaningful features from raw data, remains a bottleneck due to its reliance on domain expertise. While existing automated solutions, such as those utilizing LLMs, can generate new features, they lack mechanisms to ensure these features are causally relevant and robust to different datasets.

### **Challenges**

* Feature engineering is often domain-dependent and requires expert intuition.
* Traditional feature selection methods can capture correlations but do not guarantee causality.

### **Solution Overview**

This research enhances the **Context-Aware Automated Feature Engineering (CAAFE) model** by integrating **causal inference** and an **iterative feature selection process**, refining the feature engineering pipeline. Our approach aims to improve feature relevance, reduce reliance on domain knowledge, and enhance model generalization.

### **Key Enhancements to the CAAFE Model**

#### **Causal Feature Engineering**

* Identifies and retains features that have a true **causal impact** on the target variable rather than relying on correlations.
* Uses **causal inference techniques**, specifically the **DoWhy** library, to measure the causal importance of features.
* Investigates the impact of applying **causality filtering before and after feature selection** to assess its effect on model performance.

**Finding:**

* **Applying causality filtering alone led to a decrease in accuracy**, indicating that removing features purely based on causality may discard predictive attributes.
* However, **causal inference can enhance model generalization** by reducing the reliance on spurious correlations.
* The experiments demonstrate that **applying causality before or after feature selection reduced accuracy**, suggesting that **causal filtering should be carefully integrated** rather than used in isolation.

#### **Self-Improving Feature Selection**

* Implements an **iterative selection mechanism**, where different feature selection methods are applied sequentially to refine the feature set.
* Ensures that only **the most relevant and non-redundant features** are retained.

**Methods Used:**

1. **Recursive Feature Elimination (RFE)** – Iteratively removes the least important features.
2. **LassoCV** – Uses L1 regularization to eliminate unimportant features.
3. **ElasticNetCV** – Combines L1 and L2 regularization to improve feature selection.
4. **Mutual Information** – Measures information gain between features and the target variable (**achieved the best accuracy across datasets**).
5. **Boruta** – A feature selection algorithm based on Random Forest importance scores.
6. **Gradient Boosting** – Identifies the most relevant features based on boosting techniques.
7. **Decision Tree** – Selects features based on their contribution to tree splits.

**Finding:**

* **Mutual Information** consistently provided the **best results across datasets**, suggesting that selecting features based on information gain enhances model performance.
* Iterative feature selection ensures that **redundant or weakly predictive features** are eliminated, improving classification accuracy.

#### **Automated Feature Engineering with LLMs**

* **Generates new features** using **Large Language Models (LLMs)** based on dataset descriptions, enhancing feature diversity.
* **Employs a 10-epoch iterative feature generation process**, where each iteration introduces a new feature.
* **Applies causal filtering and feature selection** after generating new features to retain only the most meaningful attributes.
* **Uses CAAFE-generated features**, including:
  + **Aggregation-based features** (e.g., sums, averages, ratios).
  + **Dimensionality reduction-based features** (e.g., transformations that enhance separability).

**Finding:**

* **Using all CAAFE-generated features across 10 iterations improves feature diversity** but requires rigorous selection methods to remove non-informative features.
* Evaluating feature importance using **TabPFNClassifier** ensures that only the most **predictive** features are retained.

## **Experimental Evaluation**

### **Evaluation Metrics**

| **Dataset** | **Best Feature Selection Method** | **Feature Selection Accuracy** | **Accuracy** |
| --- | --- | --- | --- |
| **CMC** | Mutual Information | **0.61** | 0.5908 |
| **Diabetes** | Mutual Information | **0.8177** | 0.7969 |
| **Eucalyptus Dataset** | Decision Tree | **0.734** | 0.734 |

## 

### **Datasets**

To ensure robustness, we test our approach on multiple datasets:

* **Diabetes Dataset** (Medical domain)
* **Credit-G Dataset** (Financial domain)
* **Tic-Tac-Toe Dataset** (Game strategy domain)
* **Kaggle Health-Insurance Dataset** (Economics domain)
* **Student Performance Dataset** (Education domain)

**Baseline Comparison**

We compare our method against several baseline approaches to demonstrate its efficacy:

1. **Standard CAAFE Approach**: This method, which utilized GPT-3.5 and GPT-4 for feature generation, did not incorporate causal filtering or iterative self-improvement processes. Our approach builds upon this by integrating these elements to enhance feature utility and relevance.
2. **Feature Selection without Causality**: Here, features are selected purely based on statistical metrics without considering causal relationships. This contrasts with our method where causality informs the selection process, ensuring more robust and theoretically sound features.
3. **Causality First, Then Feature Selection**: Initially, features are evaluated for causal relationships, and only those passing this scrutiny are subjected to further selection using multiple methods. This ensures that the features retained are both causally relevant and optimally informative.
4. **Integrated Causality and Feature Selection**: Our approach applies causality analysis directly integrated with feature selection. This dual process uses multiple methods to refine the feature set, ensuring that the causality-informed features are optimized for predictive performance.

In each of these comparisons, our use of Gemini-2.0-Flash for feature generation marks a significant technological step forward from the earlier LLMs used in CAAFE, potentially contributing to our method's enhanced ability to discern and utilize complex patterns in data.

**Hypothesis:** Features identified through **causal inference will enhance model generalization**.

🔹 **Findings:**

* Removing features using causal inference **decreased accuracy** (CMC dataset dropped to **0.439**).

### **Experiment 1: Evaluating Causal and then Feature Engineering**

* **Hypothesis:** Features identified through causal inference will enhance model generalization.
* **Process:**
  1. Extract dataset metadata.
  2. Generate features using LLMs.
  3. Filter features using causal inference techniques.
  4. Filter features using features selection methods.
  5. Train TabPFNClassifier on selected features.
  6. Evaluate using Accuracy.

### **Experiment 2: Evaluating Feature Engineering and then Causal**

* **Hypothesis:** Features identified through causal inference will enhance model generalization.
* **Process:**
  1. Extract dataset metadata.
  2. Generate features using LLMs.
  3. Filter features using features selection methods.
  4. Filter features using causal inference techniques.
  5. Train TabPFNClassifier on selected features.
  6. Evaluate using Accuracy.

**Hypothesis:** Iterative refinement using **features selection** improves model performance.

🔹 **Findings:**

* **Mutual Information consistently achieved the best results** (0.61 on CMC, 0.8177 on Diabetes).

### **Experiment 3: Evaluating Self-Improving Feature Selection**

* **Hypothesis:** Iterative refinement using SHAP values and Decision Trees improves model performance.
* **Process:**
  1. Extract dataset metadata.
  2. Generate features using LLMs.
  3. Filter features using features selection methods.
  4. Train TabPFNClassifier on selected features.
  5. Evaluate using Accuracy.

### **Related Work**

* **Large Language Models for Automated Data Science: Introducing CAAFE for Context-Aware Automated Feature Engineering** – This work introduced the use of Large Language Models (LLMs) for automated feature generation, but it did not incorporate causal filtering. Our approach builds upon this by utilizing all features generated by the LLM, as opposed to their method of selectively adding features only if they demonstrate accuracy improvements. We extend their framework by applying both feature selection and causality analysis to all generated features. Our methodology was indeed inspired by their initial use of LLMs for feature creation.
* **Optimized Feature Generation for Tabular Data via LLMs with Decision Tree Reasoning** – This study employs decision trees to evaluate features generated by LLMs but does not incorporate causality in its analysis. In contrast, our method uses a modified version of the CAAFE code for feature generation and introduces an iterative process where feature improvement is continuously sought from the LLM, utilizing more advanced models. This iterative enhancement inspired us to manipulate the generated features more extensively.
* **Automated Feature Engineering Using Causal Inference** – While this research focuses on applying causality in feature engineering, it does not leverage LLMs for the generation of features. Our solution differs by integrating causality with LLM-driven feature generation, thereby enriching the potential for uncovering meaningful patterns. We drew inspiration from their causal approach to apply a similar methodology to the LLM-generated features.

## **Conclusion**

This research enhances the CAAFE model by integrating feature selection. The proposed approach demonstrates improved generalization and robustness across multiple datasets. By ensuring selected features iteratively refined, our method provides a scalable solution for automated feature engineering. Future work includes exploring reinforcement learning for adaptive feature selection and expanding LLM capabilities for domain-specific feature generation.

This research advances traditional feature engineering by incorporating a combination of automated feature generation, causal inference, and iterative feature selection. Initially, we modified the CAAFE code to generate all possible features using LLMs. We then implemented feature selection techniques to refine this extensive set of features. Interestingly, after applying feature selection, our method achieved higher scores than the original CAAFE approach. However, subsequent application of causal analysis unexpectedly reduced model accuracy. These findings indicate that while causality is theoretically appealing for ensuring feature relevance, it may not consistently enhance predictive performance across different contexts. This underscores the importance of context-specific evaluations when integrating sophisticated methods such as causal inference in feature engineering processes.