STUDY PROJECT

DOPPELGÄNGER DETECTION

Alvin Fauzi Murod
Mina Lee

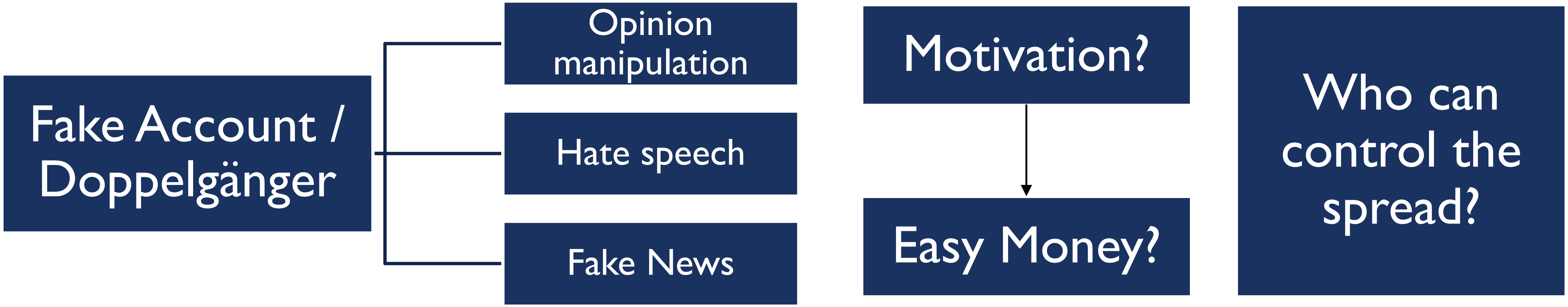
Cyber Security
Brandenburg University of Technology Cottbus-Senftenberg

b-tu Brandenburg
University of Technology
Cottbus - Senftenberg

CONTENTS

- Introduction
- Research question
- Practical tasks
- Conclusion

INTRODUCTION



Fake Account / Doppelgänger

Opinion manipulation

Hate speech

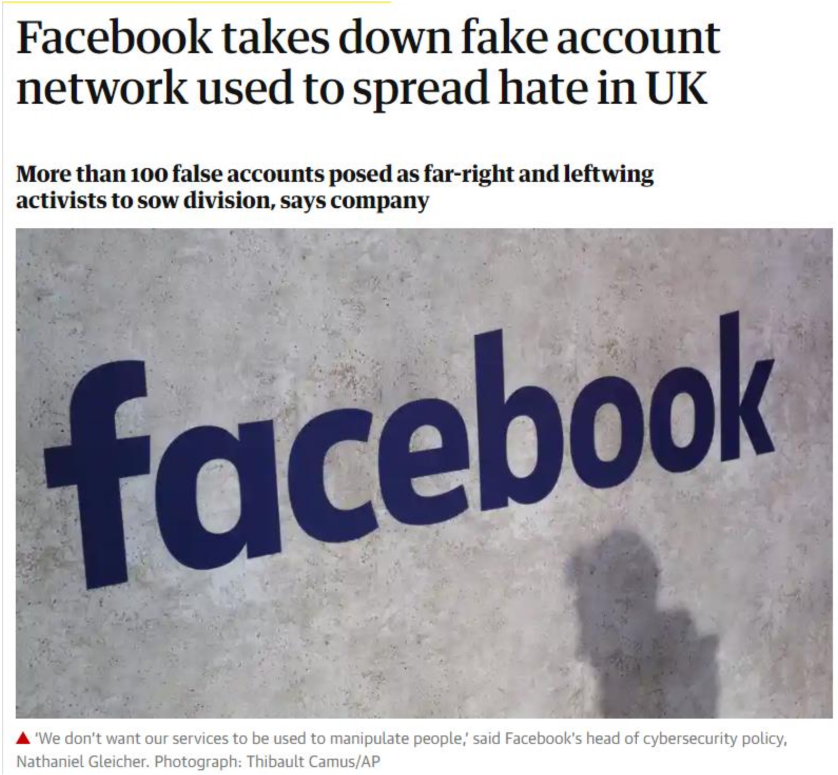
Fake News

Motivation?

Easy Money?

Who can control the spread?

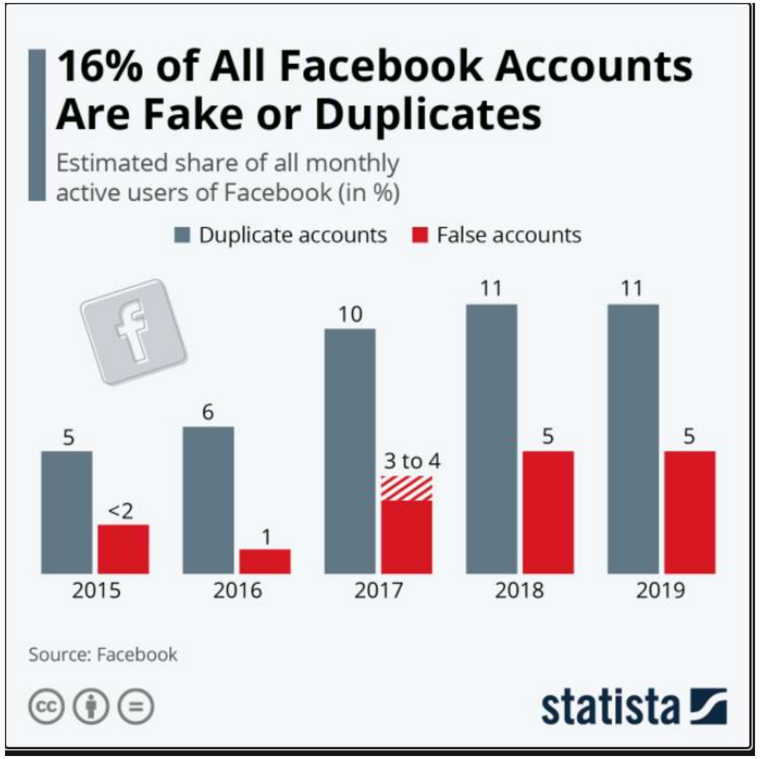
INTRODUCTION



Facebook takes down fake account network used to spread hate in UK

More than 100 false accounts posed as far-right and leftwing activists to sow division, says company

▲ 'We don't want our services to be used to manipulate people,' said Facebook's head of cybersecurity policy, Nathaniel Gleicher. Photograph: Thibault Camus/AP

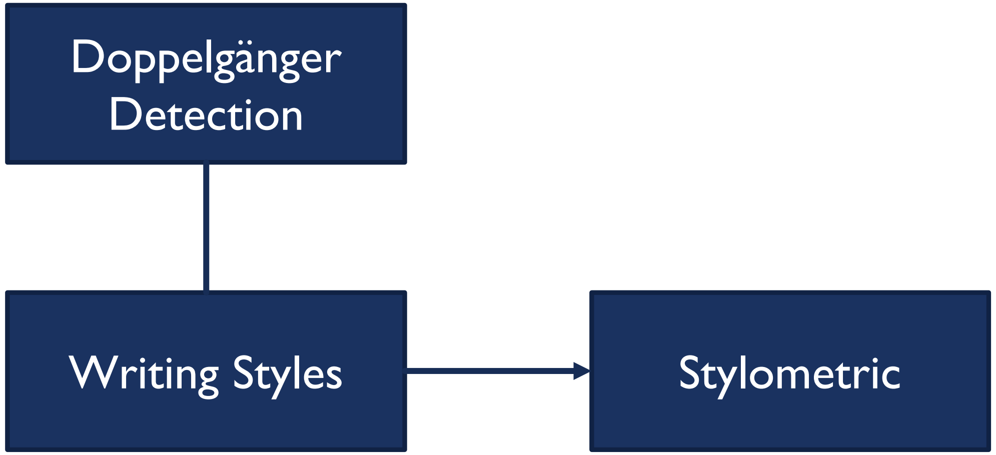


16% of All Facebook Accounts Are Fake or Duplicates

Estimated share of all monthly active users of Facebook (in %)

Source: Facebook

INTRODUCTION



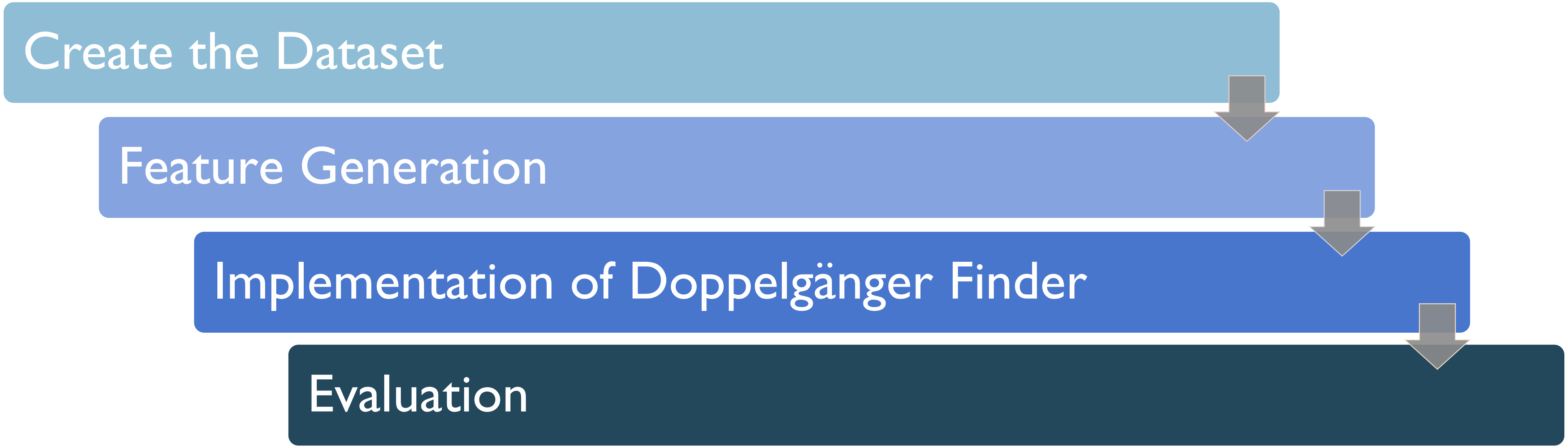
Stylometric :
Techniques of analyzing texts for evidence of authenticity, author identity, and other questions

It is based on the observation that authors tend to write in relatively consistent, recognizable and unique ways

RESEARCH QUESTION

- What is the result regarding the implementation of Doppelgängers detection algorithm in machine learning in terms of detecting doppelgänger based on stylometric features in zeit.de website?

PRACTICAL TASKS



Create the Dataset

Feature Generation

Implementation of Doppelgänger Finder

Evaluation

DATASET

Web Scraping

- Collect user comments (content of comments, username, article's title, and the published date of the comments) from main page of the news website *Zeit Online*
- Steps
  - Download the web page 'https://zeit.de' using the ***request*** library
  - Parse HTML documents in the web page using ***BeautifulSoup*** library
  - Inspect the page and define the classes/tags of data which are to be parsed to extract data
  - Store collected data into the local database(MySQL)
    - Table 'articles' contains data of article title name, article URL, author name, number of comments, published date of articles
    - Table 'comments' contains data of article title name, username, content of comment, published date of comments

DATASET

Collect comments from “Unique Users”

- Collect comments from 50 unique users with at least 100 different comments posted by that user
- Each comment should contain at least 50 words

FEATURE GENERATION

Pre-processing

- Remove stop words
  - Ex) English: "the", "a", "an", "in", "I", "we", etc
  - Ex) German: "der", "die", "das", "mit", "und", "oder", etc
- Lemmatize all words
  - Lemmatization is the process of converting a word to its base form
  - Ex) English: "playing", "plays", "played" → "play"

    "am", "are", "is" → "be"
  - Ex) German: "spielt", "spielte", "gespielt" → "spielen"

    "bin", "ist", "sind" → "sein"

FEATURE GENERATION

Extract features from comments

- Generate the features of comments to discover literary styles and it is used for predicting doppelgänger
- **Word-level** features
  - the average number of characters, the number of lowercase/uppercase letters, the number of digits per word
- **Vocabulary richness** features
  - the number of syllables per word, the ratio of word types
- **Sentence-level** feature
  - the number of short and long sentences, the average sentence length in characters, the average number of words per sentence
- **Content-based** features
  - the average positivity/sensitivity per word and sentence

FEATURE GENERATION

Extract features from comments

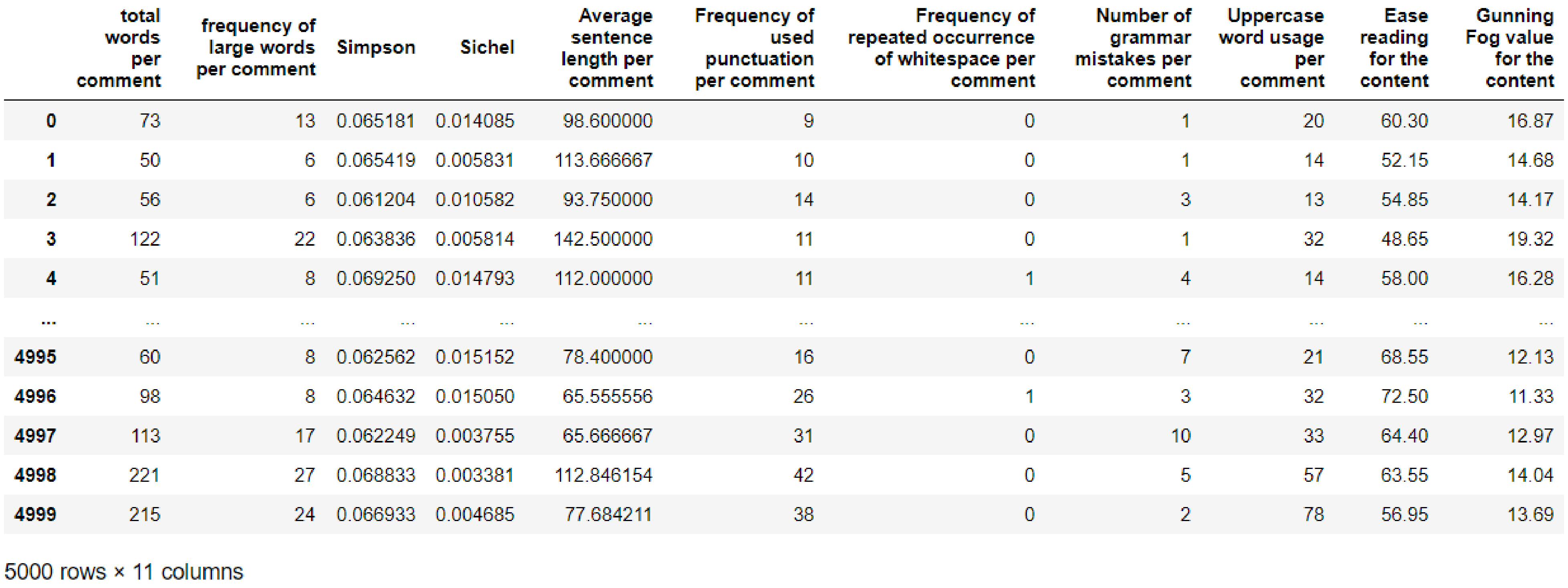
- **Idiosyncratic** features
  - the number of grammar mistakes, uppercase word usage
- Additional features
  - Noun phrase: extract noun phrases from each comment and it is useful for understanding context
  - Named Entity Recognition: recognize various types of named entities in comment
  - Language detection: detect which language is used in comment
  - Ease-reading: score how easily readable the comment is
  - Gunning Fog: readability test that aims to determine the level of text difficulty

IMPLEMENTATION OF DOPPELGÄNGER FINDER

Reduction of the number of features

- Calculate the **covariance matrix** of a feature matrix
  - The covariance matrix measures how much the features vary from the mean with respect to each other
- Calculate the **Eigenvectors** and **Eigenvalues**
  - The eigenvector with the highest eigenvalue is the most dominant principal component of the dataset
- Calculate the **PCA** (*Principal Component Analysis*)
  - PCA is a statistical procedure that extracts the most important features of a dataset

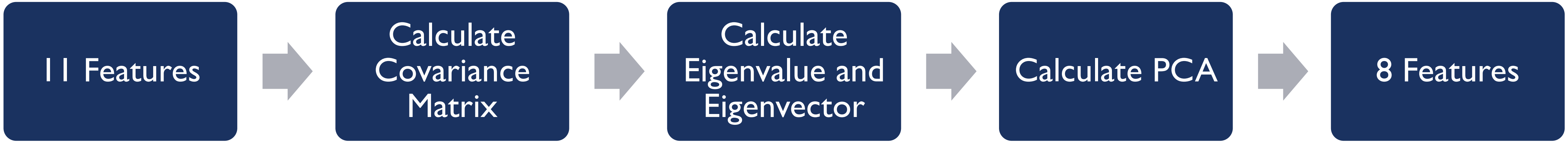
FEATURES VECTOR



	total words per comment	frequency of large words per comment	Simpson	Sichel	Average sentence length per comment	Frequency of used punctuation per comment	Frequency of repeated occurrence of whitespace per comment	Number of grammar mistakes per comment	Uppercase word usage per comment	Ease reading for the content	Gunning Fog value for the content
0	73	13	0.065181	0.014085	98.600000	9	0	1	20	60.30	16.87
1	50	6	0.065419	0.005831	113.666667	10	0	1	14	52.15	14.68
2	56	6	0.061204	0.010582	93.750000	14	0	3	13	54.85	14.17
3	122	22	0.063836	0.005814	142.500000	11	0	1	32	48.65	19.32
4	51	8	0.069250	0.014793	112.000000	11	1	4	14	58.00	16.28
...	...	...	...	...	...	...	...	...	...	...	...
4995	60	8	0.062562	0.015152	78.400000	16	0	7	21	68.55	12.13
4996	98	8	0.064632	0.015050	65.555556	26	1	3	32	72.50	11.33
4997	113	17	0.062249	0.003755	65.666667	31	0	10	33	64.40	12.97
4998	221	27	0.068833	0.003381	112.846154	42	0	5	57	63.55	14.04
4999	215	24	0.066933	0.004685	77.684211	38	0	2	78	56.95	13.69

5000 rows × 11 columns

PCA



11 Features → Calculate Covariance Matrix → Calculate Eigenvalue and Eigenvector → Calculate PCA → 8 Features

IMPLEMENTATION OF DOPPELGÄNGER FINDER

Doppelgänger Finder

- Split dataset into training set and testing set
- Train the classifier model (SVM model using a linear kernel)
  - SVM is a supervised machine learning model that uses classification algorithms

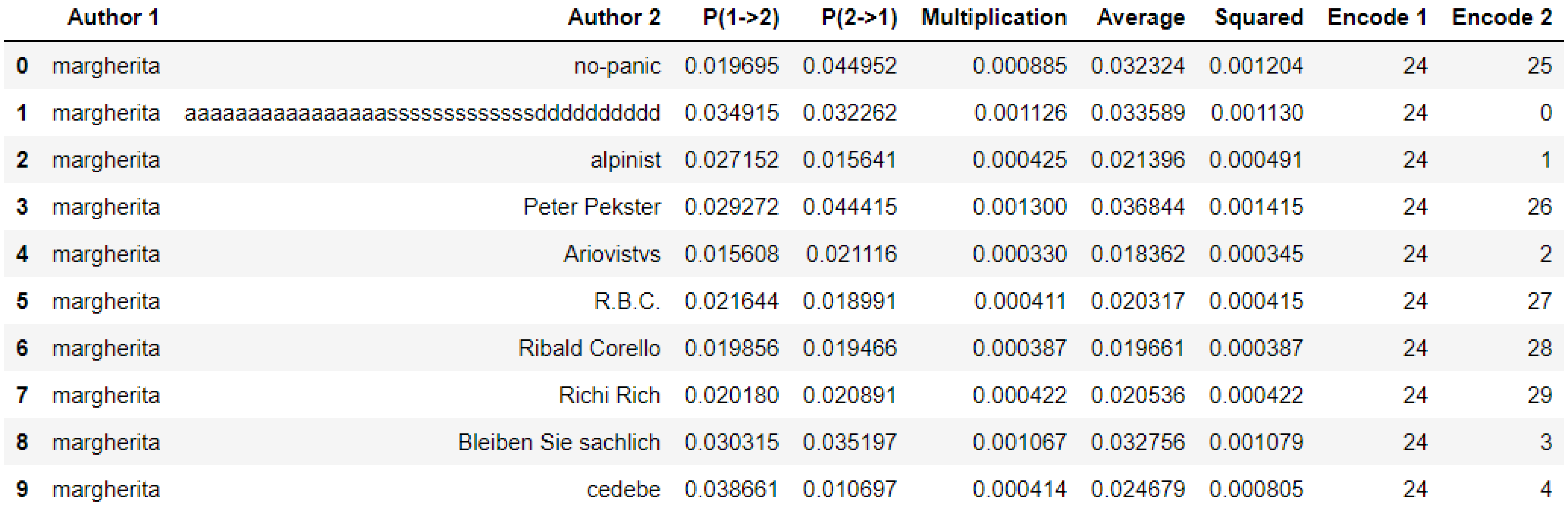
    (Supervised learning is when you train a machine learning model using labelled data)
  - SVM is effective on datasets with multiple features
  - Linear kernel works well when there are a lot of features

IMPLEMENTATION OF DOPPELGÄNGER FINDER

Doppelgänger Finder

- Calculate the pairwise probabilities with testing set
  - For author A and B, compute Pr(A→B), Pr(B→A)
- Combine the two probabilities (Pr(A→B), Pr(B→A)) into a single probability per pair
  - Average: the average of both probabilities
  - Multiplication: the multiplication of both probabilities
  - Squared average: the average of the squared probabilities
- By using a predefined threshold, decide that two authors A and B are doppelgänger or not
  - Doppelgänger if combined probability > threshold

PROBABILITIES



	Author 1	Author 2	P(1->2)	P(2->1)	Multiplication	Average	Squared	Encode 1	Encode 2
0	margherita	no-panic	0.019695	0.044952	0.000885	0.032324	0.001204	24	25
1	margherita	aaaaaaaaaaaaaaaassssssssssssdddddddddd	0.034915	0.032262	0.001126	0.033589	0.001130	24	0
2	margherita	alpinist	0.027152	0.015641	0.000425	0.021396	0.000491	24	1
3	margherita	Peter Pekster	0.029272	0.044415	0.001300	0.036844	0.001415	24	26
4	margherita	Ariovistvs	0.015608	0.021116	0.000330	0.018362	0.000345	24	2
5	margherita	R.B.C.	0.021644	0.018991	0.000411	0.020317	0.000415	24	27
6	margherita	Ribald Corello	0.019856	0.019466	0.000387	0.019661	0.000387	24	28
7	margherita	Richi Rich	0.020180	0.020891	0.000422	0.020536	0.000422	24	29
8	margherita	Bleiben Sie sachlich	0.030315	0.035197	0.001067	0.032756	0.001079	24	3
9	margherita	cedebe	0.038661	0.010697	0.000414	0.024679	0.000805	24	4

EVALUATION

Automated Threshold and Statistical Measures

- Calculate Confusion Matrix (True Negative, False positive, False Negative, True Positive)
- Calculate the Accuracy, Precision, Recall and F-score using confusion matrix
  - Accuracy = (TP + TN) / (TP + TN + FP + FN)
  - Precision = TP / (TP + FP)
  - Recall = TP / (TP + FN)
  - F-score = 2 * Precision * Recall / (Precision + Recall)

    (Weighted average of Precision and Recall)

EVALUATION

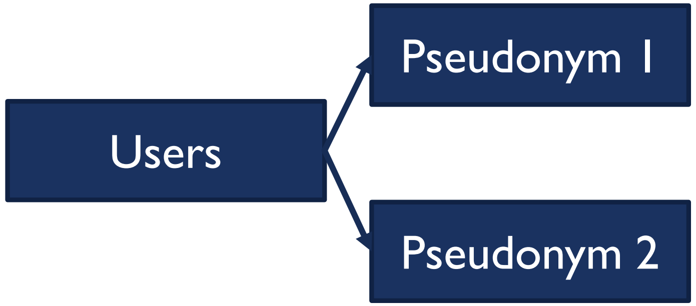
Automated Threshold and Statistical Measures

- Test all threshold between 0.0 and 1.0 with a step size of 0.001
- Apply threshold to probabilities and map all values equal to or greater than the threshold to 1 and all values less than the threshold to 0
- Evaluate each threshold by calculating F1-score
- Locate the array index that has the largest F1-score and will have the optimal threshold

EVALUATION

Experiments: Known Number of Doppelgängers

- Splitting the Users into Pseudonyms



- Number of pseudonyms (20, 40, 60) with 20 comments

Experiment 1:
20 Pseudonyms (20 comments)

Experiment 2:
40 Pseudonyms (20 comments)

Experiment 3:
60 Pseudonyms (20 comments)

- Number of comments per pseudonym (60 Pseudonyms)

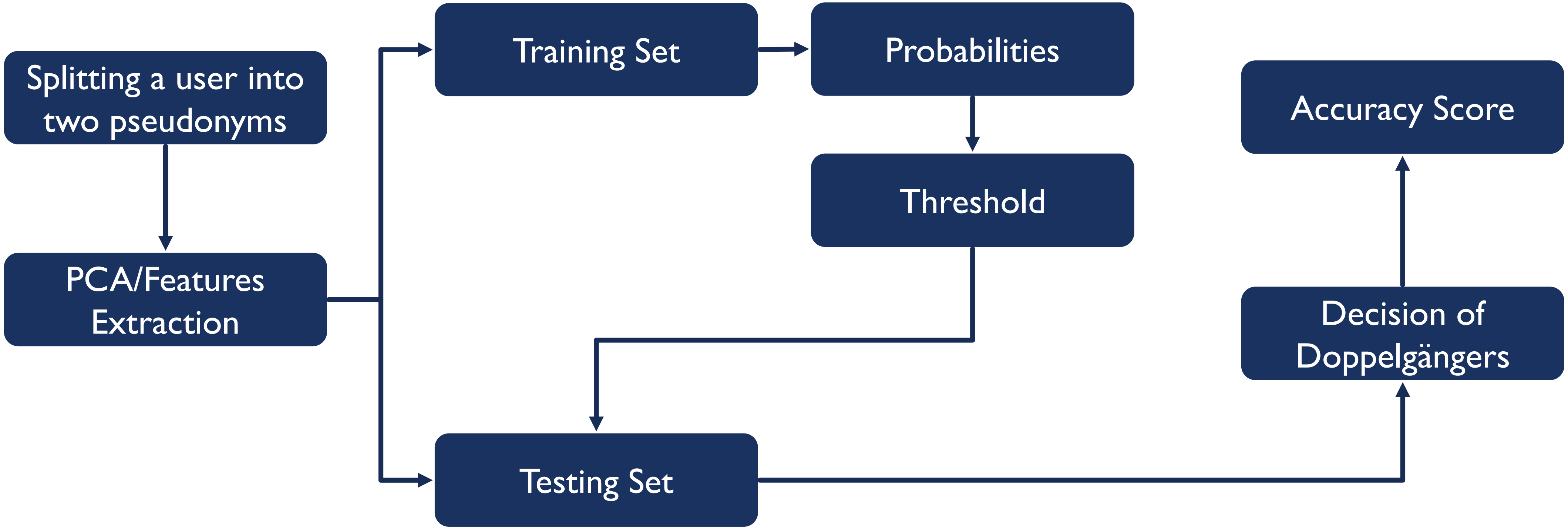
Experiment 1:
60 Pseudonyms (10 comments)

Experiment 2:
60 Pseudonyms (20 comments)

Experiment 3:
60 Pseudonyms (30 comments)

EVALUATION

Experiments: Known Number of Doppelgängers



Brandenburg
University of Technology
Cottbus - Senftenberg

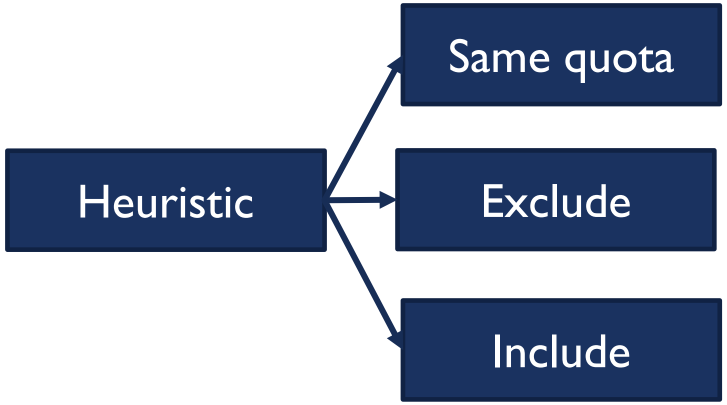
EVALUATION

Experiments: Unknown Number of Doppelgängers

- Split the Users into Pseudonyms (with minimum 750 characters written in the comment)
- Use 4 scenarios which are None, Single, Random, and All
- Use Heuristic to put Doppelgänger in the dataset (Same Quota, Exclude and Include)
- Use another classifier which are KNN and Random Forest

EXPERIMENTS

- Heuristic



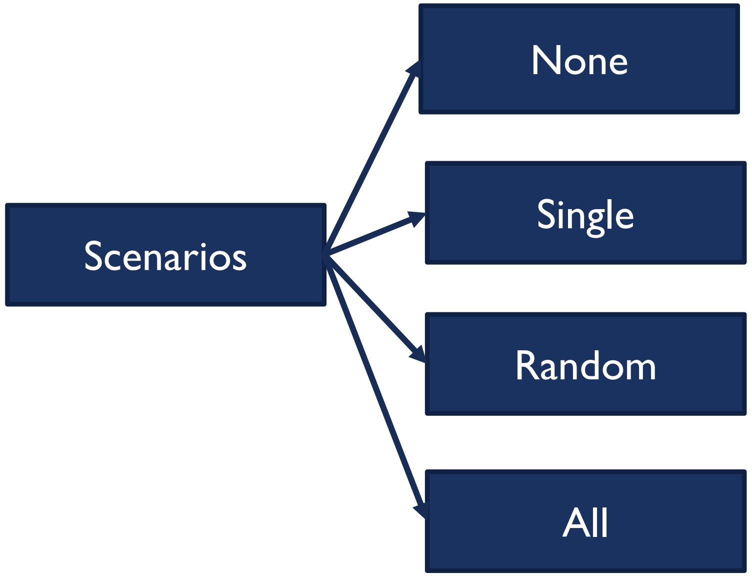
Heuristic

Same quota

Exclude

Include

- Scenarios



Scenarios

None

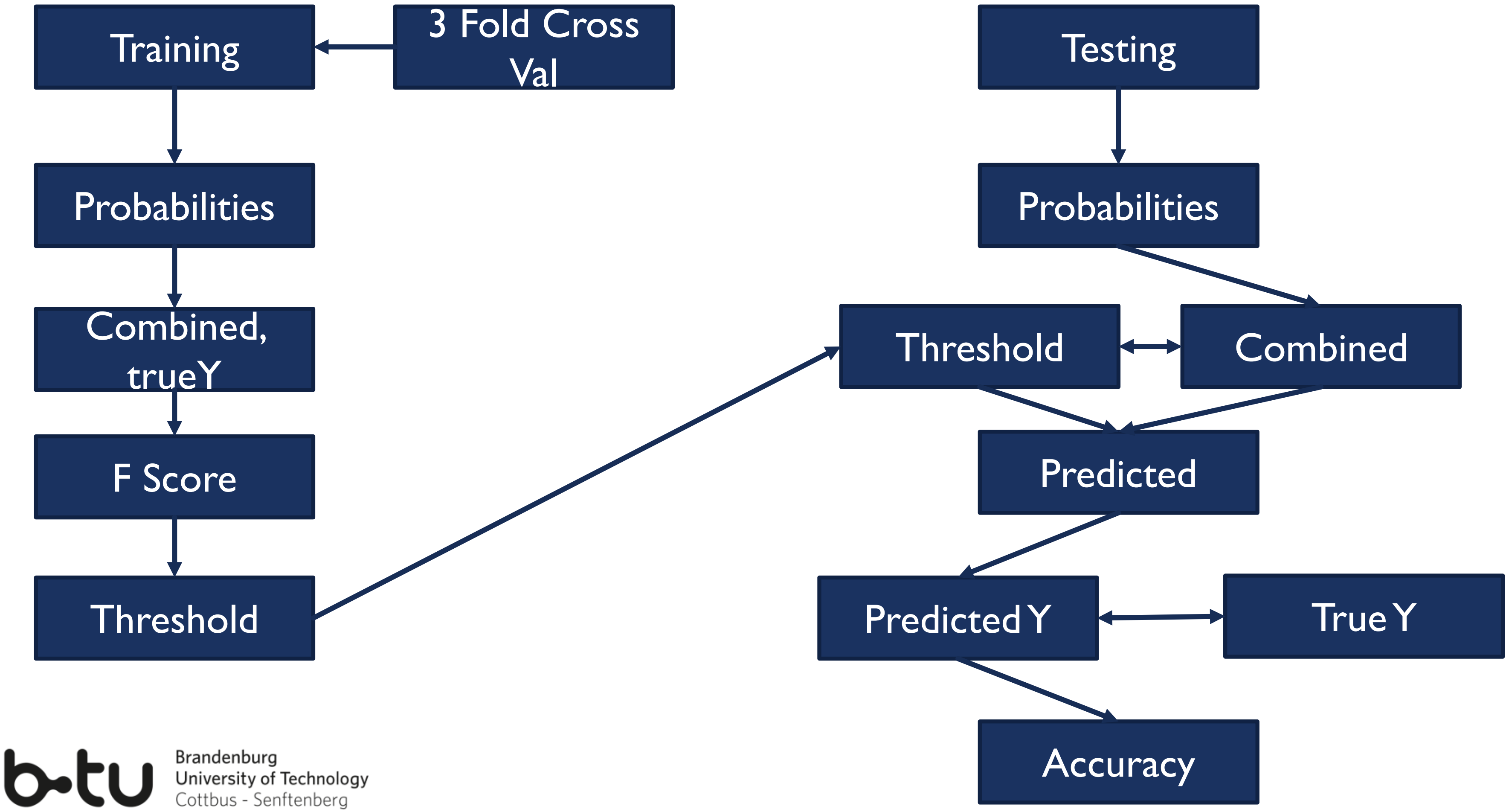
Single

Random

All

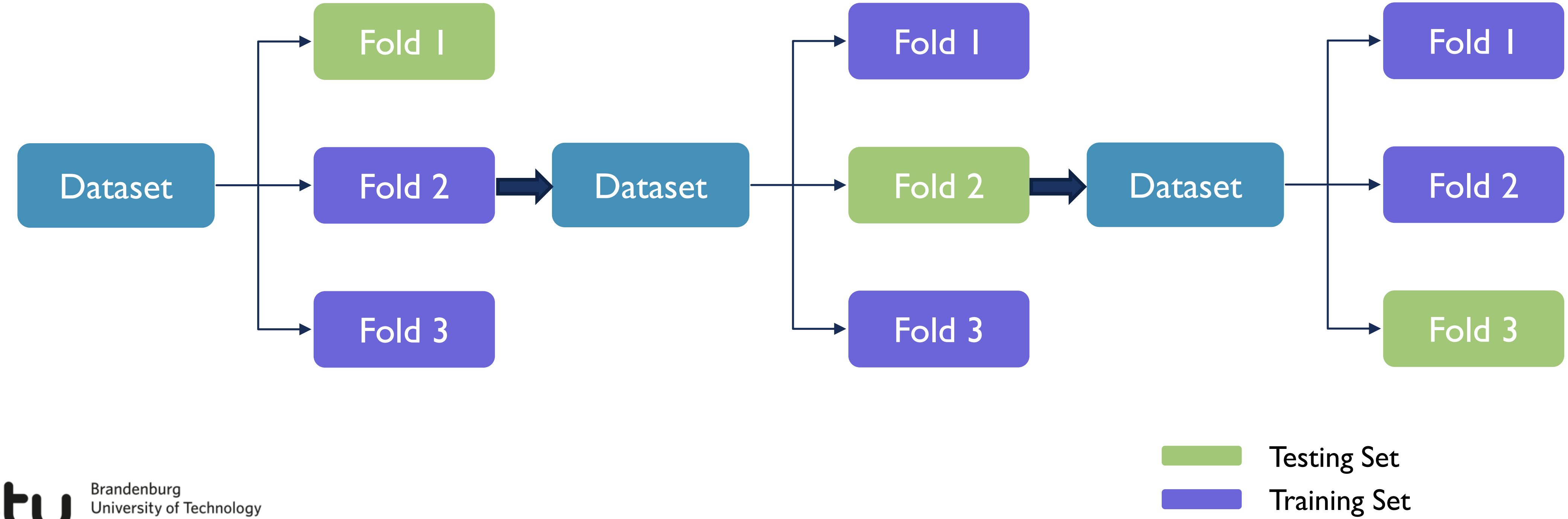
Brandenburg
University of Technology
Cottbus - Senftenberg

APPLY THRESHOLD

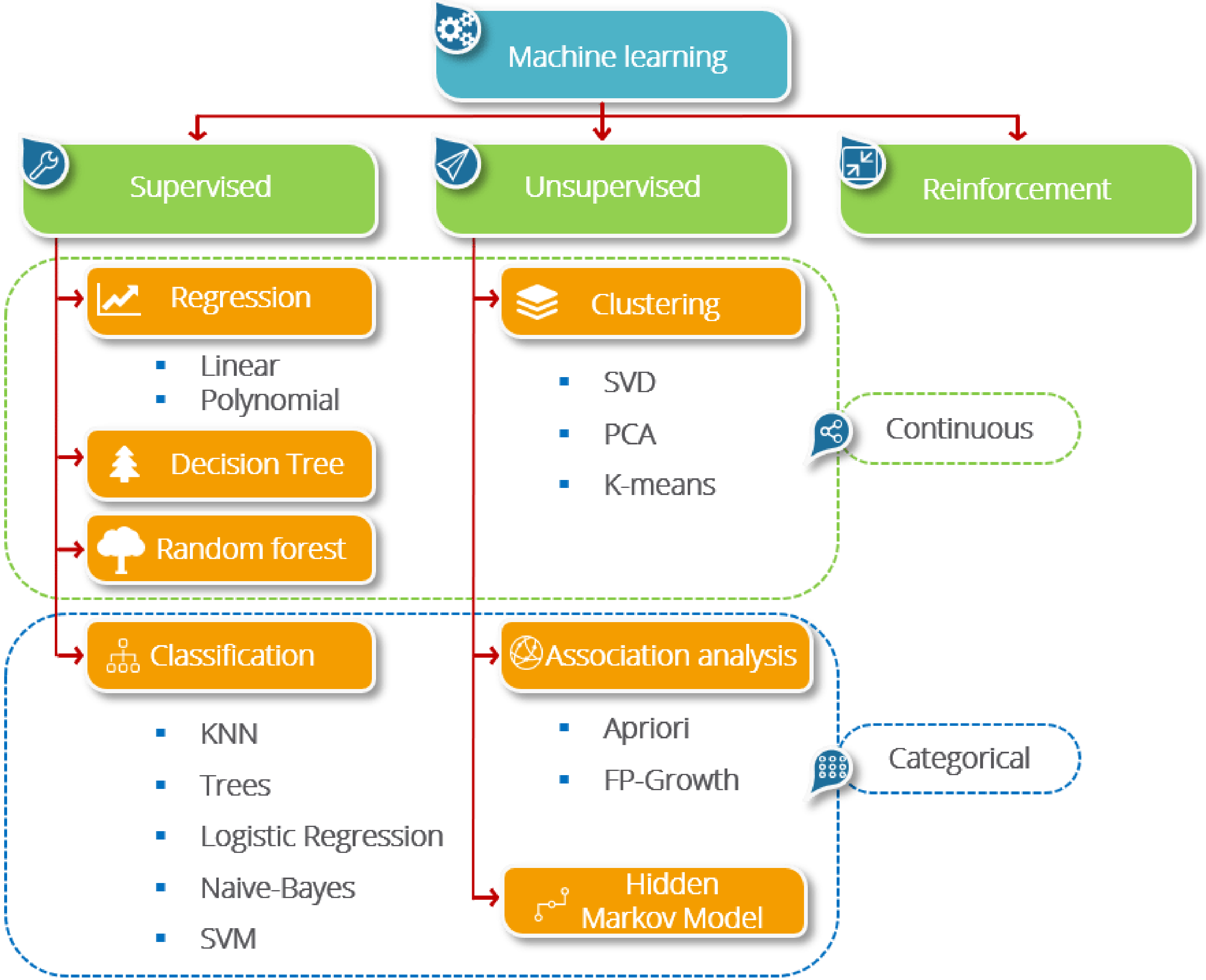


K-FOLD CROSS VALIDATION

- K-Fold Cross Validation is where a given dataset is split into a ***K*** number of folds where each fold is used as a testing set at some point



CLASSIFIERS



Machine learning

Supervised

Unsupervised

Reinforcement

Regression

- Linear
- Polynomial

Decision Tree

Random forest

Clustering

- SVD
- PCA
- K-means

Continuous

Classification

- KNN
- Trees
- Logistic Regression
- Naive-Bayes
- SVM

Association analysis

- Apriori
- FP-Growth

Hidden Markov Model

Categorical

CLASSIFIERS

SVM

- Based on vector representations
- SVM is more effective in high dimensional spaces.
- SVM is relatively memory efficient
- SVM algorithm is not suitable for large data sets.

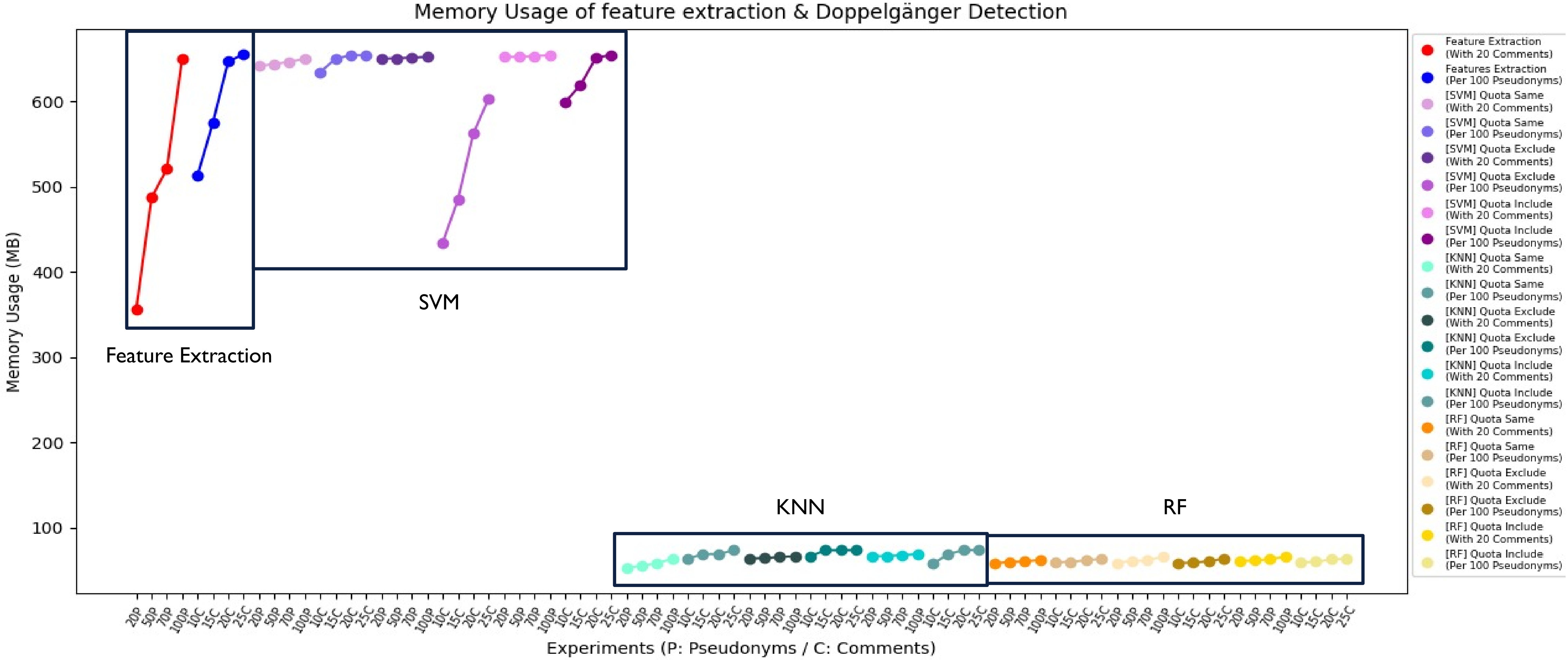
KNN

- Lazy Learning
- Based on Distance Method
- Robust with regard to the search space; for instance, classes don't have to be linearly separable
- Expensive testing of each instance, as we need to compute its distance to all known instances.

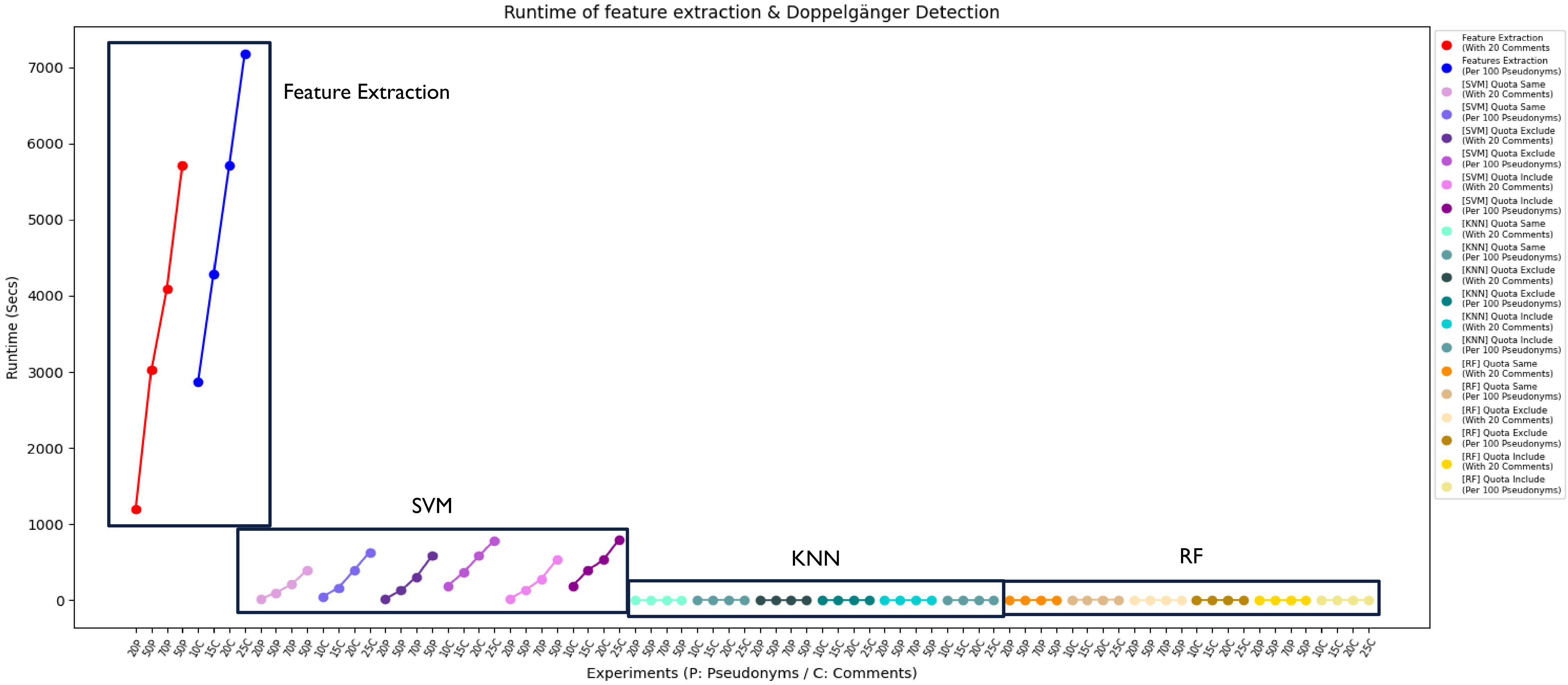
Random Forest

- Based on Decision Tree
- Can handle large data sets due to its capability to work with many variables running to thousands
- Random forests provide the highest accuracy
- It requires much computational power as well as resources as it builds numerous trees to combine their outputs

COMPUTATIONAL COMPLEXITY (MEMORY USAGE)



COMPUTATIONAL COMPLEXITY (RUNNING TIME)



CONCLUSION

- Stylometric feature helps to detect doppelgängers in comments by observing and analyzing text related to a specific author
- The number of comments influence the accuracy score
- The number of pseudonyms and the heuristics influence the accuracy score of doppelgänger detection algorithm
- The accuracy scores were increasing for all scenarios when putting more pseudonyms
- In terms of obtaining the best result to detect the Doppelgänger using stylometric features, it is needed to do more experiments in order to get the best threshold and increase the accuracy score

THANK YOU FOR YOUR ATTENTION