

A Model for Age and Gender Profiling of Social Media Accounts Based on Post Contents Documentation

Release

Cheng, Fernandez, Quindoza, Tan

August 17, 2017

1	thesis	1
1.1	Driver module	1
1.2	addEngPOS module	1
1.3	batchprocessing module	1
1.4	combinepos module	1
1.5	docs package	2
1.6	features package	2
1.7	model package.	10
1.8	pipelinewraps package	12
1.9	prepareedstheis module.	16
1.10	utility package	16
	 Python Module Index	 19
	 Index	 21

1.1 Driver module

1.2 addEngPOS module

```
class addEngPOS.ConnectionFactory
    Bases: object
    getConnectionThesis ( )

addEngPOS.add_english_pos ( )
    adds the english pos to the d :return:
```

1.3 batchprocessing module

```
batchprocessing.getPosts ( )

batchprocessing.getPostsFromFile ( filepath )

batchprocessing.updateEngPOS ( ids, texts )

batchprocessing.updatePosts ( ids, posts )

batchprocessing.writePostsToFile ( posts, filepath )
```

1.4 combinepos module

```
combinepos.combinePOS ( )
    populate the texts' combined POS
```

1.5 docs package

1.5.1 Module contents

1.6 features package

1.6.1 Submodules

1.6.2 features.CharacterFeatures module

class `features.CharacterFeatures.CharacterFeatures`

Bases: `object`

Returns the character features of a text

getNumberOfRepeatedPunctuationMarks (*text*)

Parameters **text** -- text to be processed

Returns total number of instances of consecutive punctuation marks

getNumberOfRepetitiveAlphaCharacters (*text*)

Parameters **text** -- text to be processed

Returns total number of instances that alpha characters are repeated more than twice consecutively

getNumberOfSpecialChars (*text*)

Parameters **text** -- text to be processed

Returns total number of special characters besides punctuation marks

getNumberOfWhiteSpaces (*text*)

Parameters **text** -- text to be processed

Returns total number of white spaces

getTotalNumberOfCharacters (*text*)

Parameters **text** -- text to be processed

Returns total number of characters

getTotalNumberOfDigitalNumbers (*text*)

Parameters **text** -- text to be processed

Returns total number of digital numbers

getTotalNumberOfLetters (*text*)

Parameters **text** -- text to be processed

Returns total number of letters

getTotalNumberOfUppercase (*text*)
Parameters **text** -- text to be processed
Returns total number of uppercase letters

1.6.3 features.Context module

class features.Context.**Context**
Bases: object
Returns the contextual features (words after 'my') in a text
process (*s*)
Parameters **s** -- text to be processed
Returns text containing the contextual features

1.6.4 features.EmojisEmoticons module

class features.EmojisEmoticons.**EmojisEmoticons**
Bases: object
getEmojiTFIDF (*data*)
Parameters **data** -- text to be processed
Returns TFIDF of the extracted emojis
getLabels ()

1.6.5 features.Feature module

class features.Feature.**Feature** (*X, y, source, data=None*)
Bases: object
Applies dimension reduction to data
applyExtraction (*selection*)
applies feature selection
Parameters

- **selection** -- feature extraction technique
- **type** -- Gender, Age, or Both

Returns feature extracted data
applySelection (*selection, type*)
applies feature selection
Parameters

- **selection** -- feature selection technique
- **type** -- Gender, Age, or Both

Returns feature selected data
getFeatures (*selection, mode*)
applies feature selection or extraction
Parameters

- **selection** -- feature selection or extraction technique

- **mode** -- Gender, Age, or Both

Returns feature selected or extracted data

useLasso (*mode*)

applies LASSO feature selection

Parameters **selection** -- feature selection or extraction technique

Returns feature selected data

1.6.6 features.FeatureExtract module

class features.FeatureExtract.**FeatureExtract** (*source, mindf, maxdf*)

Bases: object

Extracts features from the text and post time

clean (*x*)

cleans the data

Parameters **x** -- text data

Returns cleaned text

fit_transform (*X*)

Parameters **X** -- text data

Returns dataframe containing features extracted

get_liwc ()

reads the LIWC csv files

Returns dataframe containing the LIWC results

transform (*X*)

The transform is only done after fitting the data, useful for TFIDF features

Parameters **X** -- text data

Returns dataframe containing features extracted

1.6.7 features.FunctionWordCount module

class features.FunctionWordCount.**FunctionWordCount**

Bases: object

FUNCTIONWORDS_FILENAME = 'features/functionwords.txt'

getAdpositionCount (*text*)

Parameters **text** -- string to be counted for adposition words

Returns total number of adposition words

getAllFunctionWordCount (*text*)

Parameters **text** -- string to be counted for all function words

Returns total number of all function words

getArticleCount (*text*)

Parameters **text** -- string to be counted for articles

Returns total number of articles

getAuxillaryCount (*text*)

Parameters **text** -- string to be counted for auxillary words

Returns total number of auxillary words

getConjunctionCount (*text*)

Parameters **text** -- string to be counted for conjunctions

Returns total number of conjunctions

getInterjectionCount (*text*)

Parameters **text** -- string to be counted for interjections

Returns total number of interjections

getProSentenceCount (*text*)

Parameters **text** -- string to be counted for pro-sentence words

Returns total number of pro-sentence words

getPronounCount (*text*)

Parameters **text** -- string to be counted for pronouns

Returns total number of pronouns

1.6.8 features.Links module

class features.Links.**Links**

Bases: object

get_keywords (*link*)

get_links (*text*)

get_list_keywords (*text*)

get_title (*link*)

1.6.9 features.POSFeature module

class features.POSFeature.**POSFeature**

Bases: object

getCombinedPOSTag (*post*)

This method combines the resulting English and Filipino POS tags from the two separate POS Tagger. :param post: the document to be processed :return: returns a string of combined POS tags joined by "-"

getEnglishPOS (*text*)

Parameters **text** -- text to be processed

Returns returns a string of the resulting POS tags joined by "-"

populateMappingDictionary ()

This method populates the dictionary to include the list of mapped Filipino POS to its equivalent English POS by reading the contents of the mapping.txt file

1.6.10 features.POSSequencePattern module

class features.POSSequencePattern.POSSequencePattern (documentList)

Bases: object

MAX_LENGTH = 7

candidateGen (fList)

Param fList: list of POS sequence generated previously

Returns dictionary of POS sequences newly generated, all initialized with a value of 0

computeFairSCP (key, count)

Parameters

- **key** -- pos sequence
- **count** -- document frequency of pos sequence

Returns symmetrical conditional probability of given pos sequence

minePOSPatterns (minsup, minadherence)

Parameters

- **minsup** -- user-supplied minimum support
- **minadherence** -- user-supplied minimum adherence

Returns list of pos patterns that satisfy the thresholds

retrievePOSTags_docFrequency ()

creates a dictionary with the POS tags document frequency

1.6.11 features.Structure module

class features.Structure.Structure

Bases: object

ABBREVIATIONS_FILENAME = 'features/abbreviations.txt'

getAvgNCharacterPerParagraph (text)

Parameters **text** -- text to be processed

Returns returns a float of the average number of characters per paragraphs detected in the text

getAvgNSentencePerParagraph (text)

Parameters **text** -- text to be processed

Returns returns a float of the average number of sentences per paragraphs detected in the text

getAvgNWordPerParagraph (*text*)

Parameters **text** -- text to be processed

Returns returns a float on the average number of words per paragraphs detected in the text

getAvgNWordPerSentence (*text*)

Parameters **text** -- text to be processed

Returns returns a float of the average number of words per sentences detected in the text

getNParagraphs (*text*)

Parameters **text** -- text to be processed

Returns returns an integer of the number of detected paragraphs in the text.

getNSentenceBegLower (*text*)

Parameters **text** -- text to be processed

Returns returns an integer on the number of sentences beginning with an lowercase.

getNSentenceBegUpper (*text*)

Parameters **text** -- text to be processed

Returns returns an integer on the number of sentences beginning with an uppercase.

getNSentences (*text*)

Parameters **text** -- text to be processed

Returns returns an integer of the number of sentences detected in the text

getParagraphs (*text*)

Parameters **text** -- text to be processed

Returns returns a list containing the detected paragraphs

1.6.12 features.TFIDF module

class features.TFIDF.**TFIDF** (*mindf, maxdf*)

Bases: object

Processes the TFIDF of text

getFeatureNames ()

Returns labels of the features

get_testing_TFIDF (*test*)

Parameters **documentList** -- testing text data

Returns tfidf of the text

get_training_TFIDF (*documentList*)

Parameters **documentList** -- training text data

Returns tfidf of the text

1.6.13 features.WordCount module

```
class features.WordCount.WordCount
    Bases: object

    ABBREVIATIONS_FILENAME = 'features/abbreviations.txt'

    getAveLengthWords ( text )
        Parameters text -- string to be used
        Returns     average length of words

    getDictOfWordsMappedToOccurrence ( text )
        Parameters text -- string to be used
        Returns     dictionary of words mapped to occurrence

    getEntropy ( text )
        Parameters text -- string to be used
        Returns     entropy

    getHapaxDislegomena ( text )
        Parameters text -- string to be used
        Returns     hapax dislegomena

    getHapaxLegomena ( text )
        Parameters text -- string to be used
        Returns     hapax legomena

    getHonoresR ( text )
        Parameters text -- string to be used
        Returns     honores r

    getLolHmCount ( text )
        Parameters text -- string to be used
        Returns     number of lol's and hmm's with the use of regex

    getNDifferentWords ( text )
        Parameters text -- string to be used
        Returns     number of unique words

    getNWordsBegCapital ( text )
        Parameters text -- string to be used
        Returns     number of words beginning with a capital letter

    getNWordsWithRepLetters ( text )
        Parameters text -- string to be used
        Returns     number of words with repeating letters
```

getOccurrenceArray (*text*)

Parameters **text** -- string to be used

Returns array of word occurrences

getRatioOfHapaxDislegomena (*text*)

Parameters **text** -- string to be used

Returns ratio of hapax dislegomena to total number of words

getRatioOfHapaxLegomena (*text*)

Parameters **text** -- string to be used

Returns ratio of hapax legomena to total number of words

getRatioOfNetAbbrev (*text*)

Parameters **text** -- string to be used

Returns ratio of net abbreviations to total number of words

getRatioOfShortWords (*text*)

Parameters **text** -- string to be used

Returns ratio of words with less than 3 characters to total number of words

getRatioOfUniqueWords (*text*)

Parameters **text** -- string to be used

Returns ratio of unique words to total number of words

getSichelsS (*text*)

Parameters **text** -- string to be used

Returns sichels s

getSimpsonsD (*text*)

Parameters **text** -- string to be used

Returns simpsons d

getTotalNumberOfWords (*text*)

Parameters **text** -- string to be used

Returns total number of words

getWordLengthFreqDist (*text*)

Parameters **text** -- string to be used

Returns an array with the word length frequency distribution from length 1 to 20

getYulesK (*text*)

Parameters **text** -- string to be used

Returns yules k measure

1.6.14 Module contents

1.7 model package

1.7.1 Submodules

1.7.2 model.Document module

```
class model.Document.Document ( content, posSequence )  
    Bases: object
```

1.7.3 model.Post module

```
class model.Post.Post ( id, content, epos, fpos )  
    Bases: object
```

1.7.4 model.RootModel module

```
class model.RootModel.RootModel ( data, type, modelType, k=10 )  
    Bases: object  
    This class represents the parallel and combined structure. Its results can be fed to the StackModel for the stacked model structure.
```

```
evaluateKfold ( train_predictions=None, test_predictions=None )  
    Parameters    • train_predictions -- predictions of the model for the training data  
                  • test_predictions -- predictions of the model for the testing data
```

Returns returns the metrics for both training data and testing data

```
getPredictions ( )  
    Returns the predictions of the model for training and testing data
```

```
getTestingUser ( ind )  
    Parameters ind -- k-fold index  
    Returns users for the testing data for the ith k-fold
```

```
getTestingX ( ind )  
    Parameters ind -- k-fold index  
    Returns testing data for the ith k-fold
```

```
getTestingy ( ind )  
    Parameters ind -- k-fold index  
    Returns testing results for the ith k-fold
```

```
getTrainingUser ( ind )  
    Parameters ind -- k-fold index  
    Returns users for the training data for the ith k-fold
```

getTrainingX (*ind*)
Parameters **ind** -- k-fold index
Returns training data for the ith k-fold

getTrainingy (*ind*)
Parameters **ind** -- k-fold index
Returns training results for the ith k-fold

1.7.5 model.StackModel module

class model.StackModel.**StackModel** (*root, modelType, data, type, k=10*)
Bases: object
This class represents the stacked structure.

evaluateKfold (*train_predictions=None, test_predictions=None*)
Parameters • **train_predictions** -- predictions of the model for the training data
• **test_predictions** -- predictions of the model for the testing data
Returns returns the metrics for both training data and testing data

getPredictions ()
Returns the predictions of the model for training and testing data

getTestingUser (*ind*)
Parameters **ind** -- k-fold index
Returns users for the testing data for the ith k-fold

getTestingX (*ind*)
Parameters **ind** -- k-fold index
Returns testing data for the ith k-fold

getTestingy (*ind*)
Parameters **ind** -- k-fold index
Returns testing results for the ith k-fold

getTrainingUser (*ind*)
Parameters **ind** -- k-fold index
Returns users for the training data for the ith k-fold

getTrainingX (*ind*)
Parameters **ind** -- k-fold index
Returns training data for the ith k-fold

getTrainingy (*ind*)
Parameters **ind** -- k-fold index
Returns training results for the ith k-fold

1.7.6 Module contents

1.8 pipelinewraps package

1.8.1 Submodules

1.8.2 pipelinewraps.AgeRangeWrap module

class pipelinewraps.AgeRangeWrap.**AgeRangeWrap**

Bases: sklearn.base.TransformerMixin

Transforms the age to numerical labels TransformerMixin gives it the standard fit and transform functions to transform the data

fit (*X*, *y=None*, ****fit_params**)

transform (*X*, ****transform_params**)

pipelinewraps.AgeRangeWrap.**enrange** (*x*)

Parameters *x* -- age of the user

Returns age range group

pipelinewraps.AgeRangeWrap.**getClasses** ()

Returns array of the age ranges

1.8.3 pipelinewraps.CharacterWrap module

class pipelinewraps.CharacterWrap.**CharacterWrap**

Bases: sklearn.base.TransformerMixin

Processes all character features of the data. TransformerMixin gives it the standard fit and transform functions to transform the data

fit (*X*, *y=None*, ****fit_params**)

transform (*X*, *y=None*, ****transform_params**)

1.8.4 pipelinewraps.ContextualWrap module

class pipelinewraps.ContextualWrap.**ContextualWrap** (*target=None*)

Bases: sklearn.base.TransformerMixin

Processes all contextual features of the data. TransformerMixin gives it the standard fit and transform functions to transform the data

fit (*X*, **args*, ****kwargs**)

transform (*X*, *y=None*, ****transform_params**)

1.8.5 pipelinewraps.EmojiWrap module

```
class pipelinewraps.EmojiWrap.EmojiWrap ( target=None )
    Bases: sklearn.base.TransformerMixin
    Processes all emoji features of the data. TransformerMixin gives it the standard fit and transform
    functions to transform the data

    fit ( X, *args, **kwargs )

    transform ( X, y=None, **transform_params )
```

1.8.6 pipelinewraps.ExtractionWrap module

```
class pipelinewraps.ExtractionWrap.ExtractionWrap ( extraction, target=None )
    Bases: sklearn.base.TransformerMixin
    Performs feature extraction

    fit ( X, *args, **kwargs )

    transform ( X, y=None, **transform_params )
```

1.8.7 pipelinewraps.FunctionWrap module

```
class pipelinewraps.FunctionWrap.FunctionWrap
    Bases: sklearn.base.TransformerMixin
    Processes all function word features of the data. TransformerMixin gives it the standard fit and trans-
    form functions to transform the data

    fit ( X, y=None, **fit_params )

    transform ( X, y=None, **transform_params )
```

1.8.8 pipelinewraps.GenderWrap module

```
class pipelinewraps.GenderWrap.GenderWrap
    Bases: sklearn.base.TransformerMixin
    Transforms the gender to numerical labels TransformerMixin gives it the standard fit and transform
    functions to transform the data

    fit ( X, y=None, **fit_params )

    transform ( X, **transform_params )
```

```
pipelinewraps.GenderWrap.enrange ( x )
```

Parameters **x** -- gender of the user

Returns 0 for F and 1 for M

```
pipelinewraps.GenderWrap.getClasses ( )
```

Returns returns the gender classes

1.8.9 pipelinewraps.ItemSelector module

```
class pipelinewraps.ItemSelector.ItemSelector ( key )
    Bases: sklearn.base.BaseEstimator, sklearn.base.TransformerMixin

    fit ( x, y=None )

    transform ( data_dict )
```

1.8.10 pipelinewraps.LinkWrap module

```
class pipelinewraps.LinkWrap.LinkWrap ( target=None )
    Bases: sklearn.base.TransformerMixin
    Processes all link features of the data. TransformerMixin gives it the standard fit and transform functions to transform the data

    fit ( X, *args, **kwargs )

    transform ( X, y=None, **transform_params )
```

1.8.11 pipelinewraps.POSSeqWrap module

```
class pipelinewraps.POSSeqWrap.POSSeqWrap
    Bases: sklearn.base.TransformerMixin
    Processes all POS features of the data. TransformerMixin gives it the standard fit and transform functions to transform the data

    fit ( X, y=None, **fit_params )

    transform ( X, y=None, **transform_params )
```

```
pipelinewraps.POSSeqWrap.dfToDocument ( df )
```

1.8.12 pipelinewraps.PostTimeWrap module

```
class pipelinewraps.PostTimeWrap.PostTimeWrap
    Bases: sklearn.base.TransformerMixin
    Processes all word features of the data. TransformerMixin gives it the standard fit and transform functions to transform the data

    fit ( X, y=None, **fit_params )

    transform ( X, **transform_params )
```

```
pipelinewraps.PostTimeWrap.enrange ( x )
```

Parameters **x** -- exact hour posted

Returns time group

```
pipelinewraps.PostTimeWrap.getClasses ( )
```

Returns returns the post time classes

1.8.13 pipelinewraps.SelectionWrap module

```
class pipelinewraps.SelectionWrap.SelectionWrap ( selection, target=None )
    Bases: sklearn.base.TransformerMixin
    Performs feature selection

    fit ( X, y, *args, **kwargs )

    transform ( X, y=None, **transform_params )
```

1.8.14 pipelinewraps.StackAgeRangeWrap module

```
class pipelinewraps.StackAgeRangeWrap.StackAgeRangeWrap
    Bases: sklearn.base.TransformerMixin
    Transforms the age multiclass to multilabel binary TransformerMixin gives it the standard fit and
    transform functions to transform the data

    fit ( X, y=None, **fit_params )

    transform ( X, **transform_params )

pipelinewraps.StackAgeRangeWrap.getClasses ( )
    Returns array of the age ranges
```

1.8.15 pipelinewraps.StackGenderWrap module

```
class pipelinewraps.StackGenderWrap.StackGenderWrap
    Bases: sklearn.base.TransformerMixin
    Transforms the gender multiclass to multilabel binary TransformerMixin gives it the standard fit and
    transform functions to transform the data

    fit ( X, y=None, **fit_params )

    transform ( X, **transform_params )

pipelinewraps.StackGenderWrap.getClasses ( )
    Returns returns the gender classes
```

1.8.16 pipelinewraps.StructureWrap module

```
class pipelinewraps.StructureWrap.StructureWrap
    Bases: sklearn.base.TransformerMixin
    Processes all structure features of the data. TransformerMixin gives it the standard fit and transform
    functions to transform the data

    fit ( X, y=None, **fit_params )

    transform ( X, y=None, **transform_params )
```

1.8.17 pipelinewraps.WordWrap module

```
class pipelinewraps.WordWrap.WordWrap
    Bases: sklearn.base.TransformerMixin
    Processes all word features of the data. TransformerMixin gives it the standard fit and transform
    functions to transform the data

    fit ( X, y=None, **fit_params )

    transform ( X, y=None, **transform_params )
```

1.8.18 Module contents

1.9 preparedsthesi module

```
class preparedsthesi.ConnectionFactory
    Bases: object

    getConnectionThesis ( )

preparedsthesi.addposts ( )

preparedsthesi.addusers ( limit=None )
```

1.10 utility package

1.10.1 Submodules

1.10.2 utility.DataCleaner module

```
class utility.DataCleaner.DataCleaner
    Bases: object

    URL = 'URL'

    USERNAME = 'USERNAME'

    clean_data ( post_content )

    clean_email ( post_content )
```

1.10.3 utility.LanguageDetector module

```
class utility.LanguageDetector.Language
    Bases: object

    ENGLISH = 0

    FILIPINO = 1
```

TAGLISH = 2

UNKNOWN = -1

getLanguage (*code*)

Parameters **code** -- integer assigned to represent a language

Returns the meaning of the codes

class utility.LanguageDetector.**LanguageDetector**

Bases: object

englishOrTagalog (*string*)

Parameters **string** -- string to be identified as either English or Tagalog

Returns strings of "en" (English) or "tl" (Tagalog)

getLanguage (*text*)

Parameters **text** -- string to be language detected

Returns "ENGLISH", "FILIPINO" or "TAGALOG", else "UNKNOWN"

getLanguageDetailed (*text*)

Parameters **text** -- string to be language identified

Returns detailed probabilities of the languages detected, else "UNKNOWN"

1.10.4 utility.PostCleaner module

class utility.PostCleaner.**PostCleaner**

Bases: object

changeEmojisToText (*postContent*)

Parameters **postContent** -- text to be processed

Returns returns a string where the detected emojis are replaced into the label "EMOJI"

changeForeignToText (*postContent*)

Parameters **postContent** -- text to be processed

Returns returns a string where the detected foreign languages are replaced into the label "FOREIGN"

changeLinkToText (*postContent*)

Parameters **postContent** -- text to be processed

Returns returns a string where the links are replaced into the label "URL"

fixAcronymSpaces (*postContent*)

Parameters **postContent** -- text to be processed

Returns returns a string with fixed acronym spaces

getEmojis (*postContent*)

Parameters **postContent** -- text to be processed

Returns returns a list of emojis detected in the text

insertSpace (*postContent*)

Parameters **postContent** -- text to be processed

Returns returns a string formatted so that emojis that are stucked together will have a space in between them for easier processing later on

normalizeUnicode (*postContent*)

Parameters **postContent** -- text to be processed

Returns returns a text with the normalize unicode string

removeEmojis (*postContent*)

Parameters **postContent** -- text to be processed

Returns returns a string without the emojis

1.10.5 Module contents

- [Index](#)
- [Module Index](#)
- [Search Page](#)

a

addEngPOS, 1

b

batchprocessing, 1

c

combinepos, 1

d

docs, 2

f

features, 9

- features.CharacterFeatures, 2
- features.Context, 3
- features.EmojisEmoticons, 3
- features.Feature, 3
- features.FeatureExtract, 4
- features.FunctionWordCount, 4
- features.Links, 5
- features.POSFeature, 5
- features.POSSequencePattern, 6
- features.Structure, 6
- features.TFIDF, 7
- features.WordCount, 8

m

model, 12

- model.Document, 10
- model.Post, 10
- model.RootModel, 10
- model.StackModel, 11

p

pipelinewraps, 16

- pipelinewraps.AgeRangeWrap, 12
- pipelinewraps.CharacterWrap, 12
- pipelinewraps.ContextualWrap, 12
- pipelinewraps.EmojiWrap, 13
- pipelinewraps.ExtractionWrap, 13
- pipelinewraps.FunctionWrap, 13
- pipelinewraps.GenderWrap, 13
- pipelinewraps.ItemSelector, 14
- pipelinewraps.LinkWrap, 14
- pipelinewraps.POSSeqWrap, 14
- pipelinewraps.PostTimeWrap, 14
- pipelinewraps.SelectionWrap, 15
- pipelinewraps.StackAgeRangeWrap, 15
- pipelinewraps.StackGenderWrap, 15
- pipelinewraps.StructureWrap, 15
- pipelinewraps.WordWrap, 16

prepareedsthesi, 16

u

utility, 18

- utility.DataCleaner, 16
- utility.LanguageDetector, 16
- utility.PostCleaner, 17

A

ABBREVIATIONS_FILENAME (features.Structure.Structure attribute), 6
ABBREVIATIONS_FILENAME (features.WordCount.WordCount attribute), 8
add_english_pos() (in module addEngPOS), 1
addEngPOS (module), 1
addposts() (in module preparedsthesi), 16
addusers() (in module preparedsthesi), 16
AgeRangeWrap (class in pipelinewraps.AgeRangeWrap), 12
applyExtraction() (features.Feature.Feature method), 3
applySelection() (features.Feature.Feature method), 3

B

batchprocessing (module), 1

C

candidateGen() (features.POSSequencePattern.POSSequencePattern method), 6
changeEmojisToText() (utility.PostCleaner.PostCleaner method), 17
changeForeignToText() (utility.PostCleaner.PostCleaner method), 17
changeLinkToText() (utility.PostCleaner.PostCleaner method), 17
CharacterFeatures (class in features.CharacterFeatures), 2
CharacterWrap (class in pipelinewraps.CharacterWrap), 12
clean() (features.FeatureExtract.FeatureExtract method), 4
clean_data() (utility.DataCleaner.DataCleaner method), 16
clean_email() (utility.DataCleaner.DataCleaner

method), 16
combinepos (module), 1
combinePOS() (in module combinepos), 1
computeFairSCP() (features.POSSequencePattern.POSSequencePattern method), 6
ConnectionFactory (class in addEngPOS), 1
ConnectionFactory (class in preparedsthesi), 16
Context (class in features.Context), 3
ContextualWrap (class in pipelinewraps.ContextualWrap), 12

D

DataCleaner (class in utility.DataCleaner), 16
dfToDocument() (in module pipelinewraps.POSSeqWrap), 14
docs (module), 2
Document (class in model.Document), 10

E

EmojisEmoticons (class in features.EmojisEmoticons), 3
EmojiWrap (class in pipelinewraps.EmojiWrap), 13
ENGLISH (utility.LanguageDetector.LanguageDetector attribute), 16
englishOrTagalog() (utility.LanguageDetector.LanguageDetector method), 17
enrange() (in module pipelinewraps.AgeRangeWrap), 12
enrange() (in module pipelinewraps.GenderWrap), 13
enrange() (in module pipelinewraps.PostTimeWrap), 14
evaluateKfold() (model.RootModel.RootModel method), 10
evaluateKfold() (model.StackModel.StackModel method), 11
ExtractionWrap (class in pipelinewraps.Extrac-

tionWrap), 13

F

Feature (class in features.Feature), 3
FeatureExtract (class in features.FeatureExtract), 4
features (module), 9
features.CharacterFeatures (module), 2
features.Context (module), 3
features.EmojisEmoticons (module), 3
features.Feature (module), 3
features.FeatureExtract (module), 4
features.FunctionWordCount (module), 4
features.Links (module), 5
features.POSFeature (module), 5
features.POSSequencePattern (module), 6
features.Structure (module), 6
features.TFIDF (module), 7
features.WordCount (module), 8
FILIPINO (utility.LanguageDetector.Language attribute), 16
fit() (pipelinewrap-
s.AgeRangeWrap.AgeRangeWrap
method), 12
fit() (pipelinewraps.CharacterWrap.Character-
Wrap method), 12
fit() (pipelinewraps.ContextualWrap.Contextual-
Wrap method), 12
fit() (pipelinewraps.EmojiWrap.EmojiWrap
method), 13
fit() (pipelinewraps.ExtractionWrap.Extraction-
Wrap method), 13
fit() (pipelinewraps.FunctionWrap.FunctionWrap
method), 13
fit() (pipelinewraps.GenderWrap.GenderWrap
method), 13
fit() (pipelinewraps.ItemSelector.ItemSelector
method), 14
fit() (pipelinewraps.LinkWrap.LinkWrap
method), 14
fit() (pipelinewraps.POSSeqWrap.POSSeqWrap
method), 14
fit() (pipelinewraps.PostTimeWrap.Post-
TimeWrap method), 14
fit() (pipelinewraps.SelectionWrap.SelectionWrap
method), 15
fit() (pipelinewraps.StackAgeRangeWrap.Stack-
AgeRangeWrap method), 15
fit() (pipelinewraps.StackGenderWrap.StackGen-
derWrap method), 15
fit() (pipelinewraps.StructureWrap.Struc-
tureWrap method), 15
fit() (pipelinewraps.WordWrap.WordWrap
method), 16

fit_transform() (features.FeatureExtract.Feature-
Extract method), 4
fixAcronymSpaces() (utility.PostCleaner.Post-
Cleaner method), 17
FunctionWordCount (class in features.Function-
WordCount), 4
FUNCTIONWORDS_FILENAME (features.Func-
tionWordCount.FunctionWordCount
attribute), 4
FunctionWrap (class in pipelinewraps.Function-
Wrap), 13

G

GenderWrap (class in pipelinewraps.Gender-
Wrap), 13
get_keywords() (features.Links.Links method), 5
get_links() (features.Links.Links method), 5
get_list_keywords() (features.Links.Links
method), 5
get_liwc() (features.FeatureExtract.FeatureExtract
method), 4
get_testing_TFIDF() (features.TFIDF.TFIDF
method), 7
get_title() (features.Links.Links method), 5
get_training_TFIDF() (features.TFIDF.TFIDF
method), 7
getAdpositionCount() (features.FunctionWord-
Count.FunctionWordCount method), 4
getAllFunctionWordCount() (features.Function-
WordCount.FunctionWordCount
method), 4
getArticleCount() (features.FunctionWordCount.-
FunctionWordCount method), 5
getAuxillaryCount() (features.FunctionWord-
Count.FunctionWordCount method), 5
getAveLengthWords() (features.WordCount.-
WordCount method), 8
getAvgNCharacterPerParagraph() (features.Struc-
ture.Structure method), 6
getAvgNSentencePerParagraph() (features.Struc-
ture.Structure method), 6
getAvgNWordPerParagraph() (features.Struc-
ture.Structure method), 7
getAvgNWordPerSentence() (features.Struc-
ture.Structure method), 7
getClasses() (in module pipelinewrap-
s.AgeRangeWrap), 12
getClasses() (in module pipelinewraps.Gender-
Wrap), 13
getClasses() (in module pipelinewraps.Post-
TimeWrap), 14
getClasses() (in module pipelinewraps.Stack-
AgeRangeWrap), 15

- getClasses() (in module pipelinewraps.StackGenderWrap), 15
- getCombinedPOSTag() (features.POSFeature.POSFeature method), 5
- getConjunctionCount() (features.FunctionWordCount.FunctionWordCount method), 5
- getConnectionThesis() (addEngPOS.ConnectionFactory method), 1
- getConnectionThesis() (prepareedstthesis.ConnectionFactory method), 16
- getDictOfWordsMappedToOccurrence() (features.WordCount.WordCount method), 8
- getEmojis() (utility.PostCleaner.PostCleaner method), 17
- getEmojiTFIDF() (features.EmojisEmoticons.EmojisEmoticons method), 3
- getEnglishPOS() (features.POSFeature.POSFeature method), 5
- getEntropy() (features.WordCount.WordCount method), 8
- getFeatureNames() (features.TFIDF.TFIDF method), 7
- getFeatures() (features.Feature.Feature method), 3
- getHapaxDislegomena() (features.WordCount.WordCount method), 8
- getHapaxLegomena() (features.WordCount.WordCount method), 8
- getHonoresR() (features.WordCount.WordCount method), 8
- getInterjectionCount() (features.FunctionWordCount.FunctionWordCount method), 5
- getLabels() (features.EmojisEmoticons.EmojisEmoticons method), 3
- getLanguage() (utility.LanguageDetector.Language method), 17
- getLanguage() (utility.LanguageDetector.LanguageDetector method), 17
- getLanguageDetailed() (utility.LanguageDetector.LanguageDetector method), 17
- getLolHmmCount() (features.WordCount.WordCount method), 8
- getNDifferentWords() (features.WordCount.WordCount method), 8
- getNParagraphs() (features.Structure.Structure method), 7
- getNSentenceBegLower() (features.Structure.Structure method), 7
- getNSentenceBegUpper() (features.Structure.Structure method), 7
- getNSentences() (features.Structure.Structure method), 7
- getNumberOfRepeatedPunctuationMarks() (features.CharacterFeatures.CharacterFeatures method), 2
- getNumberOfRepetitiveAlphaCharacters() (features.CharacterFeatures.CharacterFeatures method), 2
- getNumberOfSpecialChars() (features.CharacterFeatures.CharacterFeatures method), 2
- getNumberOfWhiteSpaces() (features.CharacterFeatures.CharacterFeatures method), 2
- getNWordsBegCapital() (features.WordCount.WordCount method), 8
- getNWordsWithRepLetters() (features.WordCount.WordCount method), 8
- getOccurrenceArray() (features.WordCount.WordCount method), 9
- getParagraphs() (features.Structure.Structure method), 7
- getPosts() (in module batchprocessing), 1
- getPostsFromFile() (in module batchprocessing), 1
- getPredictions() (model.RootModel.RootModel method), 10
- getPredictions() (model.StackModel.StackModel method), 11
- getPronounCount() (features.FunctionWordCount.FunctionWordCount method), 5
- getProSentenceCount() (features.FunctionWordCount.FunctionWordCount method), 5
- getRatioOfHapaxDislegomena() (features.WordCount.WordCount method), 9
- getRatioOfHapaxLegomena() (features.WordCount.WordCount method), 9
- getRatioOfNetAbbrev() (features.WordCount.WordCount method), 9
- getRatioOfShortWords() (features.WordCount.WordCount method), 9
- getRatioOfUniqueWords() (features.WordCount.WordCount method), 9
- getSichelsS() (features.WordCount.WordCount method), 9
- getSimpsonsD() (features.WordCount.WordCount method), 9
- getTestingUser() (model.RootModel.RootModel method), 10
- getTestingUser() (model.StackModel.StackModel method), 11
- getTestingX() (model.RootModel.RootModel method), 10
- getTestingX() (model.StackModel.StackModel method), 11
- getTestingy() (model.RootModel.RootModel method), 10
- getTestingy() (model.StackModel.StackModel method), 11

getTotalNumberOfCharacters() (features.CharacterFeatures.CharacterFeatures method), 2
getTotalNumberOfDigitalNumbers() (features.CharacterFeatures.CharacterFeatures method), 2
getTotalNumberOfLetters() (features.CharacterFeatures.CharacterFeatures method), 2
getTotalNumberOfUppercase() (features.CharacterFeatures.CharacterFeatures method), 3
getTotalNumberOfWords() (features.WordCount.WordCount method), 9
getTrainingUser() (model.RootModel.RootModel method), 10
getTrainingUser() (model.StackModel.StackModel method), 11
getTrainingX() (model.RootModel.RootModel method), 11
getTrainingX() (model.StackModel.StackModel method), 11
getTrainingy() (model.RootModel.RootModel method), 11
getTrainingy() (model.StackModel.StackModel method), 11
getWordLengthFreqDist() (features.WordCount.WordCount method), 9
getYulesK() (features.WordCount.WordCount method), 9

I

insertSpace() (utility.PostCleaner.PostCleaner method), 18
ItemSelector (class in pipelinewraps.ItemSelector), 14

L

Language (class in utility.LanguageDetector), 16
LanguageDetector (class in utility.LanguageDetector), 17
Links (class in features.Links), 5
LinkWrap (class in pipelinewraps.LinkWrap), 14

M

MAX_LENGTH (features.POSSequencePattern.POSSequencePattern attribute), 6
minePOSPatterns() (features.POSSequencePattern.POSSequencePattern method), 6
model (module), 12
model.Document (module), 10
model.Post (module), 10
model.RootModel (module), 10
model.StackModel (module), 11

N

normalizeUnicode() (utility.PostCleaner.PostCleaner method), 18

P

pipelinewraps (module), 16
pipelinewraps.AgeRangeWrap (module), 12
pipelinewraps.CharacterWrap (module), 12
pipelinewraps.ContextualWrap (module), 12
pipelinewraps.EmojiWrap (module), 13
pipelinewraps.ExtractionWrap (module), 13
pipelinewraps.FunctionWrap (module), 13
pipelinewraps.GenderWrap (module), 13
pipelinewraps.ItemSelector (module), 14
pipelinewraps.LinkWrap (module), 14
pipelinewraps.POSSeqWrap (module), 14
pipelinewraps.PostTimeWrap (module), 14
pipelinewraps.SelectionWrap (module), 15
pipelinewraps.StackAgeRangeWrap (module), 15
pipelinewraps.StackGenderWrap (module), 15
pipelinewraps.StructureWrap (module), 15
pipelinewraps.WordWrap (module), 16
populateMappingDictionary() (features.POSFeature.POSFeature method), 6
POSFeature (class in features.POSFeature), 5
POSSequencePattern (class in features.POSSequencePattern), 6
POSSeqWrap (class in pipelinewraps.POSSeqWrap), 14
Post (class in model.Post), 10
PostCleaner (class in utility.PostCleaner), 17
PostTimeWrap (class in pipelinewraps.PostTimeWrap), 14
preparedsthesiis (module), 16
process() (features.Context.Context method), 3

R

removeEmojis() (utility.PostCleaner.PostCleaner method), 18
retrievePOSTags_docFrequency() (features.POSSequencePattern.POSSequencePattern method), 6
RootModel (class in model.RootModel), 10

S

SelectionWrap (class in pipelinewraps.SelectionWrap), 15
StackAgeRangeWrap (class in pipelinewraps.StackAgeRangeWrap), 15
StackGenderWrap (class in pipelinewraps.StackGenderWrap), 15

StackModel (class in model.StackModel), 11
 Structure (class in features.Structure), 6
 StructureWrap (class in pipelinewraps.StructureWrap), 15

T

TAGLISH (utility.LanguageDetector.Language attribute), 17
 TFIDF (class in features.TFIDF), 7
 transform() (features.FeatureExtract.FeatureExtract method), 4
 transform() (pipelinewraps.AgeRangeWrap.AgeRangeWrap method), 12
 transform() (pipelinewraps.CharacterWrap.CharacterWrap method), 12
 transform() (pipelinewraps.ContextualWrap.ContextualWrap method), 12
 transform() (pipelinewraps.EmojiWrap.EmojiWrap method), 13
 transform() (pipelinewraps.ExtractionWrap.ExtractionWrap method), 13
 transform() (pipelinewraps.FunctionWrap.FunctionWrap method), 13
 transform() (pipelinewraps.GenderWrap.GenderWrap method), 13
 transform() (pipelinewraps.ItemSelector.ItemSelector method), 14
 transform() (pipelinewraps.LinkWrap.LinkWrap method), 14
 transform() (pipelinewraps.POSSeqWrap.POSSeqWrap method), 14
 transform() (pipelinewraps.PostTimeWrap.PostTimeWrap method), 14
 transform() (pipelinewraps.SelectionWrap.SelectionWrap method), 15
 transform() (pipelinewraps.StackAgeRangeWrap.StackAgeRangeWrap method), 15
 transform() (pipelinewraps.StackGenderWrap.StackGenderWrap method), 15
 transform() (pipelinewraps.StructureWrap.StructureWrap method), 15
 transform() (pipelinewraps.WordWrap.WordWrap method), 16

U

UNKNOWN (utility.LanguageDetector.Language attribute), 17
 updateEngPOS() (in module batchprocessing), 1
 updatePosts() (in module batchprocessing), 1
 URL (utility.DataCleaner.DataCleaner attribute), 16

useLasso() (features.Feature.Feature method), 4
 USERNAME (utility.DataCleaner.DataCleaner attribute), 16
 utility (module), 18
 utility.DataCleaner (module), 16
 utility.LanguageDetector (module), 16
 utility.PostCleaner (module), 17

W

WordCount (class in features.WordCount), 8
 WordWrap (class in pipelinewraps.WordWrap), 16
 writePostsToFile() (in module batchprocessing), 1

