02_nlp_with_textblob

September 29, 2021

1 NLP with TextBlob

TextBlob is a python library that provides a simple API for common NLP tasks and builds on the Natural Language Toolkit (nltk) and the Pattern web mining libraries. TextBlob facilitates part-of-speech tagging, noun phrase extraction, sentiment analysis, classification, translation, and others.

1.1 Imports & Settings

[1]:
```
% matplotlib inline
import warnings
from pathlib import Path

import numpy as np
import pandas as pd

# Visualization
import matplotlib.pyplot as plt
import seaborn as sns

# spacy, textblob and nltk for language processing
from textblob import TextBlob, Word
from nltk.stem.snowball import SnowballStemmer

# sklearn for feature extraction & modeling
from sklearn.feature_extraction.text import CountVectorizer, TfidfVectorizer,␣
↪TfidfTransformer
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import MultinomialNB         # Naive Bayes
from sklearn.linear_model import LogisticRegression
from sklearn import metrics
from sklearn.externals import joblib
```

[2]:
```
np.random.seed(42)
pd.set_option('float_format', '{:,.2f}'.format)
```

1.2 Load BBC Data

To illustrate the use of TextBlob, we sample a BBC sports article with the headline 'Robinson ready for difficult task'. Similar to spaCy and other libraries, the first step is to pass the document through a pipeline represented by the TextBlob object to assign annotations required for various tasks.

[3]:
```
path = Path('data', 'bbc')
files = path.glob('**/*.txt')
doc_list = []
for i, file in enumerate(files):
    topic = file.parts[-2]
    article = file.read_text(encoding='latin1').split('\n')
    heading = article[0].strip()
    body = ' '.join([l.strip() for l in article[1:]]).strip()
    doc_list.append([topic, heading, body])
```

[4]:
```
docs = pd.DataFrame(doc_list, columns=['topic', 'heading', 'body'])
docs.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2225 entries, 0 to 2224
Data columns (total 3 columns):
topic      2225 non-null object
heading    2225 non-null object
body       2225 non-null object
dtypes: object(3)
memory usage: 52.2+ KB
```

1.3 Introduction to TextBlob

You should already have downloaded TextBlob, a Python library used to explore common NLP tasks.

1.3.1 Select random article

[5]:
```
article = docs.sample(1).squeeze()
```

[6]:
```
print(f'Topic:\t{article.topic.capitalize()}\n\n{article.heading}\n')
print(article.body.strip())
```

```
Topic:  Sport

Robinson ready for difficult task

England coach Andy Robinson faces the first major test of his tenure as he tries
to get back to winning ways after the Six Nations defeat by Wales.  Robinson is
likely to make changes in the back row and centre after the 11-9 loss as he
contemplates Sunday's set-to with France at Twickenham. Lewis Moody and Martin
```

Corry could both return after missing the game with hamstring and shoulder problems. And the midfield pairing of Mathew Tait and Jamie Noon is also under threat. Olly Barkley immediately allowed England to generate better field position with his kicking game after replacing debutant Tait just before the hour. The Bath fly-half-cum-centre is likely to start against France, with either Tait or Noon dropping out. Tait, given little opportunity to shine in attack, received praise from Robinson afterwards, even if the coach admitted Cardiff was an "unforgiving place" for the teenage prodigy. Robinson now has a tricky decision over whether to withdraw from the firing line, after just one outing, a player he regards as central to England's future. Tait himself, at least outwardly, appeared unaffected by the punishing treatment dished out to him by Gavin Henson in particular. "I want more of that definitely," he said. "Hopefully I can train hard this week and get selected for next week but we'll have to look at the video and wait and see. "We were playing on our own 22 for a lot of the first half so it was quite difficult. I thought we defended reasonably well but we've just got to pick it up for France." His Newcastle team-mate Noon hardly covered himself in glory in his first major Test. He missed a tackle on Michael Owen in the build-up to Wales' try, conceded a penalty at the breakdown, was turned over in another tackle and fumbled Gavin Henson's cross-kick into touch, all inside the first quarter. His contribution improved in the second half, but England clearly need more of a playmaker in the inside centre role. Up front, the line-out remains fallible, despite a superb performance from Chris Jones, whose athleticism came to the fore after stepping into the side for Moody. It is more likely the Leicester flanker will return on the open side for the more physical challenge posed by the French forwards, with Andy Hazell likely to make way. Lock Ben Kay also justified his recall with an impressive all-round display on his return to the side, but elsewhere England positives were thin on the ground.

[7]:
```
parsed_body = TextBlob(article.body)
```

1.3.2 Tokenization

[8]:
```
parsed_body.words
```

[8]:
```
WordList(['England', 'coach', 'Andy', 'Robinson', 'faces', 'the', 'first',
'major', 'test', 'of', 'his', 'tenure', 'as', 'he', 'tries', 'to', 'get',
'back', 'to', 'winning', 'ways', 'after', 'the', 'Six', 'Nations', 'defeat',
'by', 'Wales', 'Robinson', 'is', 'likely', 'to', 'make', 'changes', 'in', 'the',
'back', 'row', 'and', 'centre', 'after', 'the', '11-9', 'loss', 'as', 'he',
'contemplates', 'Sunday', "'s", 'set-to', 'with', 'France', 'at', 'Twickenham',
'Lewis', 'Moody', 'and', 'Martin', 'Corry', 'could', 'both', 'return', 'after',
'missing', 'the', 'game', 'with', 'hamstring', 'and', 'shoulder', 'problems',
'And', 'the', 'midfield', 'pairing', 'of', 'Mathew', 'Tait', 'and', 'Jamie',
'Noon', 'is', 'also', 'under', 'threat', 'Olly', 'Barkley', 'immediately',
'allowed', 'England', 'to', 'generate', 'better', 'field', 'position', 'with',
'his', 'kicking', 'game', 'after', 'replacing', 'debutant', 'Tait', 'just',
```

```
'before', 'the', 'hour', 'The', 'Bath', 'fly-half-cum-centre', 'is', 'likely',
'to', 'start', 'against', 'France', 'with', 'either', 'Tait', 'or', 'Noon',
'dropping', 'out', 'Tait', 'given', 'little', 'opportunity', 'to', 'shine',
'in', 'attack', 'received', 'praise', 'from', 'Robinson', 'afterwards', 'even',
'if', 'the', 'coach', 'admitted', 'Cardiff', 'was', 'an', 'unforgiving',
'place', 'for', 'the', 'teenage', 'prodigy', 'Robinson', 'now', 'has', 'a',
'tricky', 'decision', 'over', 'whether', 'to', 'withdraw', 'from', 'the',
'firing', 'line', 'after', 'just', 'one', 'outing', 'a', 'player', 'he',
'regards', 'as', 'central', 'to', 'England', "'s", 'future', 'Tait', 'himself',
'at', 'least', 'outwardly', 'appeared', 'unaffected', 'by', 'the', 'punishing',
'treatment', 'dished', 'out', 'to', 'him', 'by', 'Gavin', 'Henson', 'in',
'particular', 'I', 'want', 'more', 'of', 'that', 'definitely', 'he', 'said',
'Hopefully', 'I', 'can', 'train', 'hard', 'this', 'week', 'and', 'get',
'selected', 'for', 'next', 'week', 'but', 'we', "'ll", 'have', 'to', 'look',
'at', 'the', 'video', 'and', 'wait', 'and', 'see', 'We', 'were', 'playing',
'on', 'our', 'own', '22', 'for', 'a', 'lot', 'of', 'the', 'first', 'half', 'so',
'it', 'was', 'quite', 'difficult', 'I', 'thought', 'we', 'defended',
'reasonably', 'well', 'but', 'we', "'ve", 'just', 'got', 'to', 'pick', 'it',
'up', 'for', 'France', 'His', 'Newcastle', 'team-mate', 'Noon', 'hardly',
'covered', 'himself', 'in', 'glory', 'in', 'his', 'first', 'major', 'Test',
'He', 'missed', 'a', 'tackle', 'on', 'Michael', 'Owen', 'in', 'the', 'build-up',
'to', 'Wales', 'try', 'conceded', 'a', 'penalty', 'at', 'the', 'breakdown',
'was', 'turned', 'over', 'in', 'another', 'tackle', 'and', 'fumbled', 'Gavin',
'Henson', "'s", 'cross-kick', 'into', 'touch', 'all', 'inside', 'the', 'first',
'quarter', 'His', 'contribution', 'improved', 'in', 'the', 'second', 'half',
'but', 'England', 'clearly', 'need', 'more', 'of', 'a', 'playmaker', 'in',
'the', 'inside', 'centre', 'role', 'Up', 'front', 'the', 'line-out', 'remains',
'fallible', 'despite', 'a', 'superb', 'performance', 'from', 'Chris', 'Jones',
'whose', 'athleticism', 'came', 'to', 'the', 'fore', 'after', 'stepping',
'into', 'the', 'side', 'for', 'Moody', 'It', 'is', 'more', 'likely', 'the',
'Leicester', 'flanker', 'will', 'return', 'on', 'the', 'open', 'side', 'for',
'the', 'more', 'physical', 'challenge', 'posed', 'by', 'the', 'French',
'forwards', 'with', 'Andy', 'Hazell', 'likely', 'to', 'make', 'way', 'Lock',
'Ben', 'Kay', 'also', 'justified', 'his', 'recall', 'with', 'an', 'impressive',
'all-round', 'display', 'on', 'his', 'return', 'to', 'the', 'side', 'but',
'elsewhere', 'England', 'positives', 'were', 'thin', 'on', 'the', 'ground'])
```

1.3.3 Sentence boundary detection

[9]:
```
parsed_body.sentences
```

[9]:
```
[Sentence("England coach Andy Robinson faces the first major test of his tenure
as he tries to get back to winning ways after the Six Nations defeat by
Wales."),
 Sentence("Robinson is likely to make changes in the back row and centre after
the 11-9 loss as he contemplates Sunday's set-to with France at Twickenham."),
 Sentence("Lewis Moody and Martin Corry could both return after missing the game
```

```
with hamstring and shoulder problems."),
 Sentence("And the midfield pairing of Mathew Tait and Jamie Noon is also under
threat."),
 Sentence("Olly Barkley immediately allowed England to generate better field
position with his kicking game after replacing debutant Tait just before the
hour."),
 Sentence("The Bath fly-half-cum-centre is likely to start against France, with
either Tait or Noon dropping out."),
 Sentence("Tait, given little opportunity to shine in attack, received praise
from Robinson afterwards, even if the coach admitted Cardiff was an "unforgiving
place" for the teenage prodigy."),
 Sentence("Robinson now has a tricky decision over whether to withdraw from the
firing line, after just one outing, a player he regards as central to England's
future."),
 Sentence("Tait himself, at least outwardly, appeared unaffected by the
punishing treatment dished out to him by Gavin Henson in particular."),
 Sentence(""I want more of that definitely," he said."),
 Sentence(""Hopefully I can train hard this week and get selected for next week
but we'll have to look at the video and wait and see."),
 Sentence(""We were playing on our own 22 for a lot of the first half so it was
quite difficult."),
 Sentence("I thought we defended reasonably well but we've just got to pick it
up for France.""),
 Sentence("His Newcastle team-mate Noon hardly covered himself in glory in his
first major Test."),
 Sentence("He missed a tackle on Michael Owen in the build-up to Wales' try,
conceded a penalty at the breakdown, was turned over in another tackle and
fumbled Gavin Henson's cross-kick into touch, all inside the first quarter."),
 Sentence("His contribution improved in the second half, but England clearly
need more of a playmaker in the inside centre role."),
 Sentence("Up front, the line-out remains fallible, despite a superb performance
from Chris Jones, whose athleticism came to the fore after stepping into the
side for Moody."),
 Sentence("It is more likely the Leicester flanker will return on the open side
for the more physical challenge posed by the French forwards, with Andy Hazell
likely to make way."),
 Sentence("Lock Ben Kay also justified his recall with an impressive all-round
display on his return to the side, but elsewhere England positives were thin on
the ground.")]
```

1.3.4 Stemming

To perform stemming, we instantiate the SnowballStemmer from the nltk library, call its .stem() method on each token and display tokens that were modified as a result:

[10]:
```
# Initialize stemmer.
stemmer = SnowballStemmer('english')
```

```
# Stem each word.
[(word, stemmer.stem(word)) for i, word in enumerate(parsed_body.words)
 if word.lower() != stemmer.stem(parsed_body.words[i])]
```

[10]:
```
[('Andy', 'andi'),
 ('faces', 'face'),
 ('tenure', 'tenur'),
 ('tries', 'tri'),
 ('winning', 'win'),
 ('ways', 'way'),
 ('Nations', 'nation'),
 ('Wales', 'wale'),
 ('likely', 'like'),
 ('changes', 'chang'),
 ('centre', 'centr'),
 ('contemplates', 'contempl'),
 ('France', 'franc'),
 ('Lewis', 'lewi'),
 ('Moody', 'moodi'),
 ('Corry', 'corri'),
 ('missing', 'miss'),
 ('hamstring', 'hamstr'),
 ('problems', 'problem'),
 ('pairing', 'pair'),
 ('Jamie', 'jami'),
 ('Olly', 'olli'),
 ('immediately', 'immedi'),
 ('allowed', 'allow'),
 ('generate', 'generat'),
 ('position', 'posit'),
 ('kicking', 'kick'),
 ('replacing', 'replac'),
 ('debutant', 'debut'),
 ('before', 'befor'),
 ('fly-half-cum-centre', 'fly-half-cum-centr'),
 ('likely', 'like'),
 ('France', 'franc'),
 ('dropping', 'drop'),
 ('little', 'littl'),
 ('opportunity', 'opportun'),
 ('received', 'receiv'),
 ('praise', 'prais'),
 ('afterwards', 'afterward'),
 ('admitted', 'admit'),
 ('unforgiving', 'unforgiv'),
 ('teenage', 'teenag'),
```

```
('prodigy', 'prodigi'),
('tricky', 'tricki'),
('decision', 'decis'),
('firing', 'fire'),
('regards', 'regard'),
('future', 'futur'),
('outwardly', 'outward'),
('appeared', 'appear'),
('unaffected', 'unaffect'),
('punishing', 'punish'),
('dished', 'dish'),
('definitely', 'definit'),
('Hopefully', 'hope'),
('selected', 'select'),
("'ll", 'll'),
('playing', 'play'),
('quite', 'quit'),
('defended', 'defend'),
('reasonably', 'reason'),
("'ve", 've'),
('France', 'franc'),
('Newcastle', 'newcastl'),
('team-mate', 'team-mat'),
('hardly', 'hard'),
('covered', 'cover'),
('glory', 'glori'),
('missed', 'miss'),
('tackle', 'tackl'),
('Wales', 'wale'),
('try', 'tri'),
('conceded', 'conced'),
('penalty', 'penalti'),
('turned', 'turn'),
('another', 'anoth'),
('tackle', 'tackl'),
('fumbled', 'fumbl'),
('inside', 'insid'),
('contribution', 'contribut'),
('improved', 'improv'),
('clearly', 'clear'),
('playmaker', 'playmak'),
('inside', 'insid'),
('centre', 'centr'),
('remains', 'remain'),
('fallible', 'fallibl'),
('despite', 'despit'),
('performance', 'perform'),
```

```
 ('Jones', 'jone'),
 ('athleticism', 'athletic'),
 ('stepping', 'step'),
 ('Moody', 'moodi'),
 ('likely', 'like'),
 ('Leicester', 'leicest'),
 ('physical', 'physic'),
 ('challenge', 'challeng'),
 ('posed', 'pose'),
 ('forwards', 'forward'),
 ('Andy', 'andi'),
 ('Hazell', 'hazel'),
 ('likely', 'like'),
 ('justified', 'justifi'),
 ('recall', 'recal'),
 ('impressive', 'impress'),
 ('elsewhere', 'elsewher'),
 ('positives', 'posit')]
```

1.3.5 Lemmatization

[11]:
```
[(word, word.lemmatize()) for i, word in enumerate(parsed_body.words)
 if word != parsed_body.words[i].lemmatize()]
```

[11]:
```
[('faces', 'face'),
 ('as', 'a'),
 ('tries', 'try'),
 ('ways', 'way'),
 ('changes', 'change'),
 ('as', 'a'),
 ('problems', 'problem'),
 ('was', 'wa'),
 ('has', 'ha'),
 ('regards', 'regard'),
 ('as', 'a'),
 ('was', 'wa'),
 ('was', 'wa'),
 ('forwards', 'forward'),
 ('positives', 'positive')]
```

Lemmatization relies on parts-of-speech (POS) tagging; `spaCy` performs POS tagging, here we make assumptions, e.g. that each token is verb.

[12]:
```
[(word, word.lemmatize(pos='v')) for i, word in enumerate(parsed_body.words)
 if word != parsed_body.words[i].lemmatize(pos='v')]
```

```
[12]: [('faces', 'face'),
       ('tries', 'try'),
       ('winning', 'win'),
       ('is', 'be'),
       ('changes', 'change'),
       ('contemplates', 'contemplate'),
       ('missing', 'miss'),
       ('pairing', 'pair'),
       ('is', 'be'),
       ('allowed', 'allow'),
       ('kicking', 'kick'),
       ('replacing', 'replace'),
       ('is', 'be'),
       ('dropping', 'drop'),
       ('given', 'give'),
       ('received', 'receive'),
       ('admitted', 'admit'),
       ('was', 'be'),
       ('has', 'have'),
       ('firing', 'fire'),
       ('outing', 'out'),
       ('regards', 'regard'),
       ('appeared', 'appear'),
       ('punishing', 'punish'),
       ('dished', 'dish'),
       ('said', 'say'),
       ('selected', 'select'),
       ('were', 'be'),
       ('playing', 'play'),
       ('was', 'be'),
       ('thought', 'think'),
       ('defended', 'defend'),
       ('got', 'get'),
       ('covered', 'cover'),
       ('missed', 'miss'),
       ('conceded', 'concede'),
       ('was', 'be'),
       ('turned', 'turn'),
       ('fumbled', 'fumble'),
       ('improved', 'improve'),
       ('remains', 'remain'),
       ('came', 'come'),
       ('stepping', 'step'),
       ('is', 'be'),
       ('posed', 'pose'),
       ('forwards', 'forward'),
       ('justified', 'justify'),
```

```
('were', 'be'),
('ground', 'grind')]
```

1.3.6 Sentiment & Polarity

TextBlob provides polarity and subjectivity estimates for parsed documents using dictionaries provided by the Pattern library. These dictionaries lexicon map adjectives frequently found in product reviews to sentiment polarity scores, ranging from -1 to +1 (negative positive) and a similar subjectivity score (objective subjective).

The .sentiment attribute provides the average for each over the relevant tokens, whereas the .sentiment_assessments attribute lists the underlying values for each token

[15]:
```
parsed_body.sentiment
```

[15]:
```
Sentiment(polarity=0.088031914893617, subjectivity=0.46456433637284694)
```

[14]:
```
parsed_body.sentiment_assessments
```

[14]:
```
Sentiment(polarity=0.088031914893617, subjectivity=0.46456433637284694,
assessments=[(['first'], 0.25, 0.3333333333333333, None), (['major'], 0.0625,
0.5, None), (['tries'], -0.1, 0.4, None), (['back'], 0.0, 0.0, None),
(['winning'], 0.5, 0.75, None), (['likely'], 0.0, 1.0, None), (['back'], 0.0,
0.0, None), (['missing'], -0.2, 0.05, None), (['game'], -0.4, 0.4, None),
(['better'], 0.5, 0.5, None), (['game'], -0.4, 0.4, None), (['likely'], 0.0,
1.0, None), (['little'], -0.1875, 0.5, None), (['teenage'], 0.0, 0.0, None),
(['central'], 0.0, 0.25, None), (['future'], 0.0, 0.125, None), (['least'],
-0.3, 0.4, None), (['unaffected'], -0.05, 0.1, None), (['particular'],
0.16666666666666666, 0.3333333333333333, None), (['more'], 0.5, 0.5, None),
(['definitely'], 0.0, 0.5, None), (['hard'], -0.2916666666666667,
0.5416666666666666, None), (['next'], 0.0, 0.0, None), (['own'], 0.6, 1.0,
None), (['first'], 0.25, 0.3333333333333333, None), (['half'],
-0.16666666666666666, 0.16666666666666666, None), (['difficult'], -0.5, 1.0,
None), (['reasonably'], 0.2, 0.6, None), (['hardly'], -0.2916666666666667,
0.5416666666666666, None), (['first'], 0.25, 0.3333333333333333, None),
(['major'], 0.0625, 0.5, None), (['first'], 0.25, 0.3333333333333333, None),
(['second'], 0.0, 0.0, None), (['half'], -0.16666666666666666,
0.16666666666666666, None), (['clearly'], 0.10000000000000002,
0.3833333333333333, None), (['more'], 0.5, 0.5, None), (['superb'], 1.0, 1.0,
None), (['more'], 0.5, 0.5, None), (['likely'], 0.0, 1.0, None), (['open'], 0.0,
0.5, None), (['more'], 0.5, 0.5, None), (['physical'], 0.0, 0.14285714285714285,
None), (['french'], 0.0, 0.0, None), (['likely'], 0.0, 1.0, None),
(['justified'], 0.4, 0.9, None), (['impressive'], 1.0, 1.0, None), (['thin'],
-0.4, 0.8500000000000001, None)])
```

1.3.7 Combine Textblob Lemmatization with `CountVectorizer`

[13]:
```
def lemmatizer(text):
    words = TextBlob(text.lower()).words
    return [word.lemmatize() for word in words]
```

[14]:
```
vectorizer = CountVectorizer(analyzer=lemmatizer, decode_error='replace')
```