Przetwarzanie danych tekstowych z reddit.com

Karolina Kotlowska

Informacje o serwisie reddit można znaleźć w Wikipedii.

- Grupa tematyczna treści publikowanych w serwisie nazywana jest *subredditem*
- Wpis może mieć etykietę (tag), tzw. *flair*
- Z reddit można pobierać dane. W jednym żądaniu można pobrać maksymalnie 100 wpisów. W ciągu minuty można wykonać żądanie co najwyżej 60 razy

Podczas laboratorium wykonamy dwa zadania:

- ZADANIE 1: pobierzemy dane z dwóch angielskojęzykowych subredditów *datascience* i *MachineLearning*, a następnie sprawdzimy, czy na podstawie treści można określić do której grupy należy wpis
- ZADANIE 2: pobierzemy dane z kilku subredditów w języku polskim znalezionych na tej stronie i analogicznie sprawdzimy, czy udało się trafnie zgadnąć źródło tekstu

0.1 Ładowanie danych

Korzystamy z gotowego API. Co 100 wpisów robimy 1-sekundową przerwę. Podajemy ile pakietów po 100 postów chcemy pobrać (parametr `length`).

In [1]:
```
# # to nie odpalamu
# import pandas as pd
# import numpy as np
# import matplotlib.pyplot as plt

# import requests
# import time
# import sys

# def pull_posts(subreddit, length):
#     """Load length x 100 posts from a subreddit"""
#     posts_list = []
#     date = None
#     while len(posts_list) < length:
#         temp_url = 'https://api.pushshift.io/reddit/search/submission'
#         temp_params = {'subreddit': subreddit, 'size': 100, 'before': d
#         temp_res = requests.get(temp_url,temp_params,timeout=None)
#         data = temp_res.json()
#         posts = data['data']
#         posts_list.append(posts)
#         earliest_utc = sys.maxsize
#         for p in posts:
#           if p['created_utc']<earliest_utc:
```

```
#             earliest_utc=p['created_utc']
#         date = earliest_utc
#         time.sleep(1)
#     return posts_list

# def flatten_list(post_groups):
#   post_list = []
#   for g in post_groups:
#       for p in g:
#           post_list.append(p)
#   return post_list

# # Próbnie pobieramy 2 pakiety po 100 wpisów

# post_groups = pull_posts('datascience',2)
# post_list = flatten_list(post_groups)
# print(len(post_list))
# print(post_list[0]['title']+' '+post_list[0]['selftext'])
```

In [2]:
```
import pandas as pd
df_ds = pd.read_csv("/content/datascience_2021-04-25.csv")
df_ml = pd.read_csv("/content/MachineLearning_2021-04-25.csv")
```

In [3]:
```
df = pd.concat([df_ds,df_ml],ignore_index=True)
df.head()
len(df)
```

Out[3]:
```
2000
```

Wyświetl załadowane dane. Teksty występują w kolumnach `title` i `selftext`

In [4]:
```
from google.colab import data_table
df = pd.DataFrame(df)
df.head()
data_table.DataTable(df, include_index=False, num_rows_per_page=10,max_co
```

```
Warning: Total number of columns (85) exceeds max_columns (81) limiting
to first (81) columns.
```

Out[4]:

	Unnamed: 0	all_awardings	allow_live_comments	author	author_flair_c
0	0	[]	False	SeaworthinessOk834	
1	1	[]	False	mfb1274	
2	2	[]	False	Ostrich_Prime	
3	3	[]	False	dex1k	
4	4	[]	False	nocturnalhustler	
...	...	...	...	...	
1995	995	[]	False	coder46	
1996	996	[]	False	robertinoc	
1997	997	[]	False	charlotteamini	
1998	998	[]	False	charlotteamini	
1999	999	[]	False	pzl07	

2000 rows × 85 columns

0.2 Przetwarzanie wstępne tekstu

Tekst powinien zostać poddany czyszczeniu i konwersji:

- usuwanie znaczników HTML i referencji (typu `&lt;`)
- usuwanie znaków przestankowych
- zastępowanie specyficznych ciągów znaków symbolem ogólnym, np. zastępowanie liczb własnym znacznikiem `#num#` , czasu `#time#`

In [5]:

```
from bs4 import BeautifulSoup
import re

def clean(text):
    # znaczniki HTML
    # text = BeautifulSoup(text,features="lxml").get_text()
    text = BeautifulSoup(text, features="html.parser").get_text()

    # różne znaki, liczby
    # text = re.sub(r"C\+\+", "cpp", text) #języki
    # text = re.sub(r"C\#", "csharp", text) #języki
    # text = re.sub(r"[#\",!?;-<>/\\*\\&-]", " ", text) #znaki
    text = re.sub(r"\[[^\]]*\]\(http.+\)", " ", text) # linki markdown
    # text = re.sub(r"\d+:\d+(:\d+)?", "#time# ", text) #czas
    # text = re.sub(r"\d+\.?\d*", "#num# ", text) #liczby
    # text = re.sub('\[[^\]]*\]',' ',text) #
    text = re.sub(r"[\.\+\=()]", " ", text) #reszta znaków
    words = text.lower().split()
    return " ".join(words)

txt = """Architektura oprogramowania.
         Framework ORM. Język programowania C++ (C#) &amp;amp; &amp;lt; &amp;gt; <htm
         \\ instrukcja if(x==-1) [removed]
         Kilka liczb: 120k$ 500000$ 23 11.53
```

```
        Time: 11:55, 12:23:00
        10MB pamieci około 2GB
        Markdown **bold title**
        p [p] empty []
        link:  [https://github.com/louisfb01/start-machine-learning-in-2
clean(txt)
```

Out[5]:
```
'architektura oprogramowania framework orm język programowania c c# & <
> \\ instrukcja if x -1 [removed] kilka liczb: 120k$ 500000$ 23 11 53 ti
me: 11:55, 12:23:00 10mb pamieci około 2gb markdown **bold title** p [p]
empty [] link:'
```

TODO 11.0.2.1

- w funkcji `load_to_data_frame`:
  - sklej kolumny 'title' i 'selftext' (dodając spację) i umieść wynik w kolumnie 'text'
  - zastosuj do kazdego elementu kolumny 'text' funkcję clean()
  - zwróć obiekt DataFrame zawierający wyłącznie kolumny 'text', 'subreddit' i 'link_flair_text'. Możesz do tego użyć funkcji DataFrame loc()

In [6]:
```
# import datetime
# def load_to_data_frame(subreddit,save=True):
#   """Załaduj listę postów do pandas DataFrame. """
#   post_groups = pull_posts(subreddit,10)
#   post_list = flatten_list(post_groups)
#   df = pd.DataFrame(post_list)
#   if save:
#     file_name = subreddit+f'_{datetime.datetime.today()}.csv'
#     df.to_csv(file_name)

#   # zabezpieczenie w przypadku pustych danych !
#   df['title']=df['title'].replace(np.nan, '', regex=True)
#   df['selftext']=df['selftext'].replace(np.nan, '', regex=True)

#   # sklej kolumny
#   df['text'] = df['title'].fillna('') + ' ' + df['selftext'].fillna('')

#   df['text']=df['text'].apply(clean(df['text']))
#   return df.loc[:, ['column1', 'column2', ...]]
```

0.3 Wizualizacja w postaci interaktywnej tabeli

In [7]:
```
# df_ds = load_to_data_frame('datascience')
# assert df_ds.shape[1]==3
# from google.colab import data_table
# data_table.DataTable(df_ds, include_index=False, num_rows_per_page=10)
```

ZADANIE 1: Data Science vs. Machine Learning

Czy *Data Science* i *Machine Learning* czymś różnią się? Czy na podstawie treści postów można je odróżnić?

Pobierzemy posty z grup datasience i MachineLearning i umieścimy w jednej tabeli DataFrame

In [8]:
```
# def concat(subredits):
#   df_list = []
#   for s in subredits:
#     df = load_to_data_frame(s)
#     print(f'{s}:{len(df)}')
#     time.sleep(1)
#     df_list.append(df)
#   return pd.concat(df_list,ignore_index=True)
#   # return df_list

# df = concat(['datascience','MachineLearning'])
```

Tabela powinna zawierać około 2200 wierszy i 3 kolumny.

In [9]:
```
data_table.DataTable(df, include_index=False, num_rows_per_page=10,max_co
```

```
Warning: Total number of columns (85) exceeds max_columns (80) limiting
to first (80) columns.
```

Out[9]:

	Unnamed: 0	all_awardings	allow_live_comments	author	author_flair_c
0	0	[]	False	SeaworthinessOk834	
1	1	[]	False	mfb1274	
2	2	[]	False	Ostrich_Prime	
3	3	[]	False	dex1k	
4	4	[]	False	nocturnalhustler	
...	...	...	...	...	
1995	995	[]	False	coder46	
1996	996	[]	False	robertinoc	
1997	997	[]	False	charlotteamini	
1998	998	[]	False	charlotteamini	
1999	999	[]	False	pzl07	

2000 rows × 85 columns

In [10]:
```
assert df.subreddit[999]=='datascience'
assert df.subreddit[1100]=='MachineLearning'
```

In [11]:
```
import numpy as np

df['title']=df['title'].replace(np.nan, '', regex=True)
df['selftext']=df['selftext'].replace(np.nan, '', regex=True)

# sklej kolumny
df['text']=df['title'] + " " + df['selftext']

# zabezpieczenie w przypadku pustych danych !
```

```
df['text']=df['text'].apply(lambda t:clean(t))

df = df.loc[:,['text','subreddit','link_flair_text']]

df.head()
```

```
<ipython-input-5-e568b04a7411>:7: MarkupResemblesLocatorWarning: The inp
ut looks more like a filename than markup. You may want to open this fil
e and pass the filehandle into Beautiful Soup.
  text = BeautifulSoup(text, features="html.parser").get_text()
```

Out[11]:

	text	subreddit	link_flair_text
0	is anaconda worth the trouble? [removed]	datascience	Discussion
1	just got accepted to an ms data science progra...	datascience	Education
2	advice on joining a bootcamp hi, i'm looking t...	datascience	Career
3	i need data to convince my professor cryptocur...	datascience	Education
4	data scientists without masters or phd, how mu...	datascience	Career

Czy posty zawierają słowo 'python'?

TODO 11.1.1.1

- znajdź indeksy wierszy zawierających słowo 'python'. Użyj metody str.find()
- utwórz DataFrame zawierającą tylko te wiersze i wyświetl

Wydaje się, że słowo występuje w obu grupach postów

In [12]:
```
indexes = df[df['text'].str.find('python') != -1].index
df_python = df.iloc[indexes]
data_table.DataTable(df_python, include_index=True, num_rows_per_page=10)
```

Out[12]:

	text	subreddit	link_flair_text
14	i will do data analysis, visualization and pre...	datascience	Career
16	for europe, is an university degree necessary ...	datascience	Discussion
26	essential python libraries for data science & ...	datascience	Education
28	open source production data science examples i...	datascience	Discussion
30	how to use the requests python library to make...	datascience	NaN
...	...	...	...
1827	[p] train googlenet and inception model on nas...	MachineLearning	Project
1857	[d] best self-teaching textbook for ml for ppl...	MachineLearning	Discussion
1915	ml charades ios app project[[p] hey, guys i'm ...	MachineLearning	Project
1946	[p] implementing perceiver: general perception...	MachineLearning	Project
1979	[d] looking for some clarification on big slee...	MachineLearning	Discussion

207 rows × 3 columns

1.1. Ekstrakcja etykiet

TODO 11.1.1.2

- Za pomocą LabelEncoder zakoduj jako liczby etykiety z kolumnt `subreddit`

In [13]:
```
from sklearn import preprocessing
le = preprocessing.LabelEncoder()
y=le.fit_transform(df['subreddit'])
le.classes_
```

Out[13]:
```
array(['MachineLearning', 'datascience'], dtype=object)
```

1.2 Ekstrakcja cech

Wypróbujemy CountVectorizer i TfidfVectorizer

- Oba mogą wyodrębniać n-gramy (ciągi n symboli)
- Symbolami mogą być słowa lub znaki
- Oba pozwalają na ograniczenie liczby terminów

TODO 11.1.2.1

- Co to jest TF-IDF?
- Czym TfidfVectorizer różni się od CountVectorizer
- Obejrzyj wyjście i skasuj!

TF-IDF to Term Frequency-Inverse Document Frequency -> metoda stosowana w przetwarzaniu języka naturalnego i analizie tekstów mająca na celu ocenę istotności danego słowa w kontekście danego dokumentu w zbiorze dokumentów. TF-IDF jest używane w zadaniach takich jak wyszukiwanie informacji, klasyfikacja tekstów itp.

CountVectorizer - Zamienia tekst na macierz wystąpień słów (count matrix), gdzie każdy wiersz reprezentuje dokument, a każda kolumna reprezentuje słowo. CountVectorizer nie uwzględnia wagi słów, dlatego ważniejsze słowa, które często występują, mogą mieć większy wpływ na analizę.

TfidfVectorizer - Zamienia tekst na macierz TF-IDF, która zawiera informacje zarówno o częstości występowania słów, jak i o ich istotności w kontekście całego zbioru dokumentów. TfidfVectorizer nadaje większą wagę słowom, które są częste w danym dokumencie, ale rzadkie w innych dokumentach, co pozwala na wyodrębnienie istotnych słów specyficznych dla danego dokumentu.

In [55]:
```
#max 2 słowa
from sklearn.feature_extraction.text import CountVectorizer,TfidfVectoriz

vectorizer = TfidfVectorizer(analyzer='word',ngram_range=(1,2),max_featur
X = vectorizer.fit_transform(df['text'])
```

In [54]:
```
# znaki (max 8)
vectorizer = TfidfVectorizer(analyzer='char_wb',ngram_range=(1,8),max_fea
X = vectorizer.fit_transform(df['text'])
```

Stopwords to terminy odrzucone (występujące zbyt czesto lub zbyt rzadko)

In [56]:
```
# tylko zliczanie słów
vectorizer = CountVectorizer(analyzer='word',ngram_range=(1,2),max_featur
X = vectorizer.fit_transform(df['text'])
```

TODO 11.1.2.2

- za pomocą `CountVectorizer` wyznacz 30 najczęściej występujacych słów.
- Z reguły najczęściej występujące słowa nie przenoszą żadnaego znaczenia Zipf's law

In [17]:
```
vectorizer = CountVectorizer(max_features=30)
vectorizer.fit(df.text)
stopwords = vectorizer.vocabulary_
stopwords
```

Out[17]:
```
{'is': 14,
 'the': 24,
 'removed': 21,
 'to': 26,
 'data': 6,
 'science': 22,
 'or': 20,
 'you': 29,
 'on': 19,
 'from': 9,
 'in': 13,
 'with': 28,
 'have': 10,
 'but': 4,
 'for': 8,
 'it': 15,
 'and': 0,
 'my': 17,
 'be': 3,
 'that': 23,
 'of': 18,
 'how': 11,
 'this': 25,
 'if': 12,
 'can': 5,
 'as': 2,
 'what': 27,
 'learning': 16,
 'are': 1,
 'do': 7}
```

1.3 Klasyfikacja

Utworzony zostanie ciąg przetwarzania (ang. pipeline) składający się z dwóch kroków:

- ekstrakcji cech z tekstów za pomocą TfidfVectorizer
- klasyfikacji z użyciem MultinomialNB

TODO 11.1.3.1

- Przetestuj co najmniej 3 konfiguracje Vectorizera i wybierz tę, która wydaje się zwracać lepsze wyniki. Na przykład oblicz średnią F1 z 10 iteracji. Napisz, którą wybrałeś
- Uwaga wyniki mogą się zmieniac, ponieważ zależą od pobranych danych.

In [18]:
```
from sklearn.naive_bayes import MultinomialNB
from sklearn.pipeline import make_pipeline

# vectorizer = TfidfVectorizer(analyzer='word',ngram_range=(1,2),stop_wor
vectorizer = TfidfVectorizer(analyzer='word',ngram_range=(1,2),max_featur
cls = MultinomialNB()


print(df.shape)
print(y)

pipeline = make_pipeline(vectorizer, cls)
from sklearn.model_selection import train_test_split
import sklearn
f1s=[]
for i in range(10):
  X_train, X_test, y_train, y_test = train_test_split(df['text'], y, test
  pipeline.fit(X_train,y_train)
  y_pred = pipeline.predict(X_test)
  print(f'Accuracy:{sklearn.metrics.accuracy_score(y_test,y_pred)} F1: {s
  f1s.append(sklearn.metrics.f1_score(y_test,y_pred,average="macro"))
print(np.array(f1s).mean())
```

```
(2000, 3)
[1 1 1 ... 0 0 0]
Accuracy:0.83 F1: 0.8299829982998299
Accuracy:0.81 F1: 0.8099239695878353
Accuracy:0.785 F1: 0.7840877708317642
Accuracy:0.795 F1: 0.7948717948717949
Accuracy:0.84 F1: 0.8399359743897559
Accuracy:0.79 F1: 0.7899159663865547
Accuracy:0.805 F1: 0.8047608320192235
Accuracy:0.785 F1: 0.7847363019699132
Accuracy:0.78 F1: 0.7796474358974359
Accuracy:0.815 F1: 0.8149953748843721
0.803285841913848
```

1.4 Walidacja krzyżowa

TODO 11.1.4.1

- Oblicz i wypisz średnie wartości metryk

In [19]:
```
from sklearn.model_selection import cross_validate
scoring = ['accuracy','precision_macro','recall_macro','f1_macro']
# scoring = scoring=['accuracy','f1_macro']
cv_results = cross_validate(pipeline, df['text'], y, cv=10,scoring=scorin

acc = cv_results['test_accuracy'].mean()
prec = cv_results['test_precision_macro'].mean()
```

```
recall = cv_results['test_recall_macro'].mean()
f1 = cv_results['test_f1_macro'].mean()

print(f'acc={acc} prec={prec} recall={recall} f1={f1}')
```

```
acc=0.7875000000000001 prec=0.7883645697782165 recall=0.7875 f1=0.787347
1352116902
```

ZADANIE 2 - Z której grupy pochodzi post w języku polskim

Celem drugiego zadania jest analogiczna próba klasyfikajci tesktów pochodzących z polskich subredditów.

Wykonamy analogiczne kroki jak w Zadaniu 1.

2.1 Ładujemy i wyświetlamy dane

Wybierz co najmniej 4 źródła, niekoniecznie takie, jak w podanym przykładzie.

In [20]:
```
subreddits = ['FashionRepsPolska', 'RPGPolska', 'PolskaPolityka', 'Krakow
```

... gdybyśmy chcieli wczytać dane bez ich powtórnego ładowania

In [21]:
```
!wget https://dysk.agh.edu.pl/s/qXz2B54Ctkm7Zgp/download/various_polish.c
!unzip various_polish_csv.zip
```

```
--2023-05-20 13:52:37--  https://dysk.agh.edu.pl/s/qXz2B54Ctkm7Zgp/downl
oad/various_polish.csv.zop
Resolving dysk.agh.edu.pl (dysk.agh.edu.pl)... 149.156.96.4, 2001:6d8:1
0:1060::6004
Connecting to dysk.agh.edu.pl (dysk.agh.edu.pl)|149.156.96.4|:443... con
nected.
HTTP request sent, awaiting response... 200 OK
Length: 214084 (209K) [application/zip]
Saving to: ‘various_polish_csv.zip’

various_polish_csv. 100%[===================>] 209.07K   193KB/s    in
1.1s

2023-05-20 13:52:40 (193 KB/s) - ‘various_polish_csv.zip’ saved [214084/
214084]

Archive:  various_polish_csv.zip
  inflating: various_polish.csv
```

In [22]:
```
import pandas as pd
df = pd.read_csv("various_polish.csv")
```

Tę wersję można też pobrać z Chmury AGH

In [23]:
```
df.head(10)
```

Out[23]:

	Unnamed: 0	text	subreddit	link_flair_text
0	0	qc	FashionRepsPolska	NaN
1	1	czy ktoś korzystał z programu zwrotów za detai...	FashionRepsPolska	NaN
2	2	qc	FashionRepsPolska	NaN
3	3	pytanie siema, 2 koszulki łącznie 800g myslici...	FashionRepsPolska	NaN
4	4	za ile paka moze byc i czy nic podejrzanego si...	FashionRepsPolska	NaN
5	5	czym wyslac problem z cssbuy siema, ogolnie to...	FashionRepsPolska	NaN
6	6	siema podpowiedziałby mi ktoś na pv odnośnie p...	FashionRepsPolska	NaN
7	7	qc na te yeezuski, co myślicie?	FashionRepsPolska	LCQC
8	8	rl czy gl? z góry thank u za pomoc batch niezn...	FashionRepsPolska	LCQC
9	9	paka dhl 6kg [removed]	FashionRepsPolska	NaN

2.2 Usuwamy teksty w języku innym niż polski

Część tekstów jest w języku angielskim Nie chcemy ich używać podczas klasyfikacji.

Potrzebne będzie narzędzie do rozpoznawania języka. W tym celu wykorzystamy bibliotekę spaCy

In [24]:
```
!pip install spacy
!python -m spacy download en_core_web_sm
!pip install spacy_language_detection
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: spacy in /usr/local/lib/python3.10/dist-packages (3.5.2)
Requirement already satisfied: spacy-legacy<3.1.0,>=3.0.11 in /usr/local/lib/python3.10/dist-packages (from spacy) (3.0.12)
Requirement already satisfied: spacy-loggers<2.0.0,>=1.0.0 in /usr/local/lib/python3.10/dist-packages (from spacy) (1.0.4)
Requirement already satisfied: murmurhash<1.1.0,>=0.28.0 in /usr/local/lib/python3.10/dist-packages (from spacy) (1.0.9)
Requirement already satisfied: cymem<2.1.0,>=2.0.2 in /usr/local/lib/python3.10/dist-packages (from spacy) (2.0.7)
Requirement already satisfied: preshed<3.1.0,>=3.0.2 in /usr/local/lib/python3.10/dist-packages (from spacy) (3.0.8)
Requirement already satisfied: thinc<8.2.0,>=8.1.8 in /usr/local/lib/python3.10/dist-packages (from spacy) (8.1.9)
Requirement already satisfied: wasabi<1.2.0,>=0.9.1 in /usr/local/lib/python3.10/dist-packages (from spacy) (1.1.1)
Requirement already satisfied: srsly<3.0.0,>=2.4.3 in /usr/local/lib/python3.10/dist-packages (from spacy) (2.4.6)
Requirement already satisfied: catalogue<2.1.0,>=2.0.6 in /usr/local/lib/python3.10/dist-packages (from spacy) (2.0.8)
Requirement already satisfied: typer<0.8.0,>=0.3.0 in /usr/local/lib/python3.10/dist-packages (from spacy) (0.7.0)
Requirement already satisfied: pathy>=0.10.0 in /usr/local/lib/python3.10/dist-packages (from spacy) (0.10.1)
Requirement already satisfied: smart-open<7.0.0,>=5.2.1 in /usr/local/lib/python3.10/dist-packages (from spacy) (6.3.0)
Requirement already satisfied: tqdm<5.0.0,>=4.38.0 in /usr/local/lib/python3.10/dist-packages (from spacy) (4.65.0)
Requirement already satisfied: numpy>=1.15.0 in /usr/local/lib/python3.10/dist-packages (from spacy) (1.22.4)
Requirement already satisfied: requests<3.0.0,>=2.13.0 in /usr/local/lib/python3.10/dist-packages (from spacy) (2.27.1)
Requirement already satisfied: pydantic!=1.8,!=1.8.1,<1.11.0,>=1.7.4 in /usr/local/lib/python3.10/dist-packages (from spacy) (1.10.7)
Requirement already satisfied: jinja2 in /usr/local/lib/python3.10/dist-packages (from spacy) (3.1.2)
Requirement already satisfied: setuptools in /usr/local/lib/python3.10/dist-packages (from spacy) (67.7.2)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.10/dist-packages (from spacy) (23.1)
Requirement already satisfied: langcodes<4.0.0,>=3.2.0 in /usr/local/lib/python3.10/dist-packages (from spacy) (3.3.0)
Requirement already satisfied: typing-extensions>=4.2.0 in /usr/local/lib/python3.10/dist-packages (from pydantic!=1.8,!=1.8.1,<1.11.0,>=1.7.4->spacy) (4.5.0)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests<3.0.0,>=2.13.0->spacy) (1.26.15)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests<3.0.0,>=2.13.0->spacy) (2022.12.7)
Requirement already satisfied: charset-normalizer~=2.0.0 in /usr/local/lib/python3.10/dist-packages (from requests<3.0.0,>=2.13.0->spacy) (2.0.12)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests<3.0.0,>=2.13.0->spacy) (3.4)
Requirement already satisfied: blis<0.8.0,>=0.7.8 in /usr/local/lib/python3.10/dist-packages (from thinc<8.2.0,>=8.1.8->spacy) (0.7.9)
Requirement already satisfied: confection<1.0.0,>=0.0.1 in /usr/local/lib/python3.10/dist-packages (from thinc<8.2.0,>=8.1.8->spacy) (0.0.4)
```

```
Requirement already satisfied: click<9.0.0,>=7.1.1 in /usr/local/lib/pyt
hon3.10/dist-packages (from typer<0.8.0,>=0.3.0->spacy) (8.1.3)
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python
3.10/dist-packages (from jinja2->spacy) (2.1.2)
2023-05-20 13:52:50.751103: I tensorflow/core/platform/cpu_feature_guar
d.cc:182] This TensorFlow binary is optimized to use available CPU instr
uctions in performance-critical operations.
To enable the following instructions: AVX2 FMA, in other operations, reb
uild TensorFlow with the appropriate compiler flags.
2023-05-20 13:52:51.846117: W tensorflow/compiler/tf2tensorrt/utils/py_u
tils.cc:38] TF-TRT Warning: Could not find TensorRT
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/c
olab-wheels/public/simple/
Collecting en-core-web-sm==3.5.0
  Downloading https://github.com/explosion/spacy-models/releases/downloa
d/en_core_web_sm-3.5.0/en_core_web_sm-3.5.0-py3-none-any.whl (12.8 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 12.8/12.8 MB 20.6 MB/s eta
0:00:00
Requirement already satisfied: spacy<3.6.0,>=3.5.0 in /usr/local/lib/pyt
hon3.10/dist-packages (from en-core-web-sm==3.5.0) (3.5.2)
Requirement already satisfied: spacy-legacy<3.1.0,>=3.0.11 in /usr/loca
l/lib/python3.10/dist-packages (from spacy<3.6.0,>=3.5.0->en-core-web-sm
==3.5.0) (3.0.12)
Requirement already satisfied: spacy-loggers<2.0.0,>=1.0.0 in /usr/loca
l/lib/python3.10/dist-packages (from spacy<3.6.0,>=3.5.0->en-core-web-sm
==3.5.0) (1.0.4)
Requirement already satisfied: murmurhash<1.1.0,>=0.28.0 in /usr/local/l
ib/python3.10/dist-packages (from spacy<3.6.0,>=3.5.0->en-core-web-sm==
3.5.0) (1.0.9)
Requirement already satisfied: cymem<2.1.0,>=2.0.2 in /usr/local/lib/pyt
hon3.10/dist-packages (from spacy<3.6.0,>=3.5.0->en-core-web-sm==3.5.0)
(2.0.7)
Requirement already satisfied: preshed<3.1.0,>=3.0.2 in /usr/local/lib/p
ython3.10/dist-packages (from spacy<3.6.0,>=3.5.0->en-core-web-sm==3.5.
0) (3.0.8)
Requirement already satisfied: thinc<8.2.0,>=8.1.8 in /usr/local/lib/pyt
hon3.10/dist-packages (from spacy<3.6.0,>=3.5.0->en-core-web-sm==3.5.0)
(8.1.9)
Requirement already satisfied: wasabi<1.2.0,>=0.9.1 in /usr/local/lib/py
thon3.10/dist-packages (from spacy<3.6.0,>=3.5.0->en-core-web-sm==3.5.0)
(1.1.1)
Requirement already satisfied: srsly<3.0.0,>=2.4.3 in /usr/local/lib/pyt
hon3.10/dist-packages (from spacy<3.6.0,>=3.5.0->en-core-web-sm==3.5.0)
(2.4.6)
Requirement already satisfied: catalogue<2.1.0,>=2.0.6 in /usr/local/li
b/python3.10/dist-packages (from spacy<3.6.0,>=3.5.0->en-core-web-sm==3.
5.0) (2.0.8)
Requirement already satisfied: typer<0.8.0,>=0.3.0 in /usr/local/lib/pyt
hon3.10/dist-packages (from spacy<3.6.0,>=3.5.0->en-core-web-sm==3.5.0)
(0.7.0)
Requirement already satisfied: pathy>=0.10.0 in /usr/local/lib/python3.1
0/dist-packages (from spacy<3.6.0,>=3.5.0->en-core-web-sm==3.5.0) (0.10.
1)
Requirement already satisfied: smart-open<7.0.0,>=5.2.1 in /usr/local/li
b/python3.10/dist-packages (from spacy<3.6.0,>=3.5.0->en-core-web-sm==3.
5.0) (6.3.0)
Requirement already satisfied: tqdm<5.0.0,>=4.38.0 in /usr/local/lib/pyt
hon3.10/dist-packages (from spacy<3.6.0,>=3.5.0->en-core-web-sm==3.5.0)
(4.65.0)
Requirement already satisfied: numpy>=1.15.0 in /usr/local/lib/python3.1
```

```
0/dist-packages (from spacy<3.6.0,>=3.5.0->en-core-web-sm==3.5.0) (1.22.
4)
Requirement already satisfied: requests<3.0.0,>=2.13.0 in /usr/local/li
b/python3.10/dist-packages (from spacy<3.6.0,>=3.5.0->en-core-web-sm==3.
5.0) (2.27.1)
Requirement already satisfied: pydantic!=1.8,!=1.8.1,<1.11.0,>=1.7.4 in
/usr/local/lib/python3.10/dist-packages (from spacy<3.6.0,>=3.5.0->en-co
re-web-sm==3.5.0) (1.10.7)
Requirement already satisfied: jinja2 in /usr/local/lib/python3.10/dist-
packages (from spacy<3.6.0,>=3.5.0->en-core-web-sm==3.5.0) (3.1.2)
Requirement already satisfied: setuptools in /usr/local/lib/python3.10/d
ist-packages (from spacy<3.6.0,>=3.5.0->en-core-web-sm==3.5.0) (67.7.2)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python
3.10/dist-packages (from spacy<3.6.0,>=3.5.0->en-core-web-sm==3.5.0) (2
3.1)
Requirement already satisfied: langcodes<4.0.0,>=3.2.0 in /usr/local/li
b/python3.10/dist-packages (from spacy<3.6.0,>=3.5.0->en-core-web-sm==3.
5.0) (3.3.0)
Requirement already satisfied: typing-extensions>=4.2.0 in /usr/local/li
b/python3.10/dist-packages (from pydantic!=1.8,!=1.8.1,<1.11.0,>=1.7.4->
spacy<3.6.0,>=3.5.0->en-core-web-sm==3.5.0) (4.5.0)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in /usr/local/lib/p
ython3.10/dist-packages (from requests<3.0.0,>=2.13.0->spacy<3.6.0,>=3.
5.0->en-core-web-sm==3.5.0) (1.26.15)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/pyth
on3.10/dist-packages (from requests<3.0.0,>=2.13.0->spacy<3.6.0,>=3.5.0-
>en-core-web-sm==3.5.0) (2022.12.7)
Requirement already satisfied: charset-normalizer~=2.0.0 in /usr/local/l
ib/python3.10/dist-packages (from requests<3.0.0,>=2.13.0->spacy<3.6.0,>
=3.5.0->en-core-web-sm==3.5.0) (2.0.12)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.1
0/dist-packages (from requests<3.0.0,>=2.13.0->spacy<3.6.0,>=3.5.0->en-c
ore-web-sm==3.5.0) (3.4)
Requirement already satisfied: blis<0.8.0,>=0.7.8 in /usr/local/lib/pyth
on3.10/dist-packages (from thinc<8.2.0,>=8.1.8->spacy<3.6.0,>=3.5.0->en-
core-web-sm==3.5.0) (0.7.9)
Requirement already satisfied: confection<1.0.0,>=0.0.1 in /usr/local/li
b/python3.10/dist-packages (from thinc<8.2.0,>=8.1.8->spacy<3.6.0,>=3.5.
0->en-core-web-sm==3.5.0) (0.0.4)
Requirement already satisfied: click<9.0.0,>=7.1.1 in /usr/local/lib/pyt
hon3.10/dist-packages (from typer<0.8.0,>=0.3.0->spacy<3.6.0,>=3.5.0->en
-core-web-sm==3.5.0) (8.1.3)
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python
3.10/dist-packages (from jinja2->spacy<3.6.0,>=3.5.0->en-core-web-sm==3.
5.0) (2.1.2)
✔ Download and installation successful
You can now load the package via spacy.load('en_core_web_sm')
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/c
olab-wheels/public/simple/
Collecting spacy_language_detection
  Downloading spacy_language_detection-0.2.1-py3-none-any.whl (6.5 kB)
Requirement already satisfied: spacy>=3.0.0 in /usr/local/lib/python3.1
0/dist-packages (from spacy_language_detection) (3.5.2)
Collecting langdetect==1.0.9 (from spacy_language_detection)
  Downloading langdetect-1.0.9.tar.gz (981 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 981.5/981.5 kB 13.2 MB/s eta
0:00:00
  Preparing metadata (setup.py) ... done
Requirement already satisfied: six in /usr/local/lib/python3.10/dist-pac
kages (from langdetect==1.0.9->spacy_language_detection) (1.16.0)
```

```
Requirement already satisfied: spacy-legacy<3.1.0,>=3.0.11 in /usr/loca
l/lib/python3.10/dist-packages (from spacy>=3.0.0->spacy_language_detect
ion) (3.0.12)
Requirement already satisfied: spacy-loggers<2.0.0,>=1.0.0 in /usr/loca
l/lib/python3.10/dist-packages (from spacy>=3.0.0->spacy_language_detect
ion) (1.0.4)
Requirement already satisfied: murmurhash<1.1.0,>=0.28.0 in /usr/local/l
ib/python3.10/dist-packages (from spacy>=3.0.0->spacy_language_detectio
n) (1.0.9)
Requirement already satisfied: cymem<2.1.0,>=2.0.2 in /usr/local/lib/pyt
hon3.10/dist-packages (from spacy>=3.0.0->spacy_language_detection) (2.
0.7)
Requirement already satisfied: preshed<3.1.0,>=3.0.2 in /usr/local/lib/p
ython3.10/dist-packages (from spacy>=3.0.0->spacy_language_detection)
(3.0.8)
Requirement already satisfied: thinc<8.2.0,>=8.1.8 in /usr/local/lib/pyt
hon3.10/dist-packages (from spacy>=3.0.0->spacy_language_detection) (8.
1.9)
Requirement already satisfied: wasabi<1.2.0,>=0.9.1 in /usr/local/lib/py
thon3.10/dist-packages (from spacy>=3.0.0->spacy_language_detection) (1.
1.1)
Requirement already satisfied: srsly<3.0.0,>=2.4.3 in /usr/local/lib/pyt
hon3.10/dist-packages (from spacy>=3.0.0->spacy_language_detection) (2.
4.6)
Requirement already satisfied: catalogue<2.1.0,>=2.0.6 in /usr/local/li
b/python3.10/dist-packages (from spacy>=3.0.0->spacy_language_detection)
(2.0.8)
Requirement already satisfied: typer<0.8.0,>=0.3.0 in /usr/local/lib/pyt
hon3.10/dist-packages (from spacy>=3.0.0->spacy_language_detection) (0.
7.0)
Requirement already satisfied: pathy>=0.10.0 in /usr/local/lib/python3.1
0/dist-packages (from spacy>=3.0.0->spacy_language_detection) (0.10.1)
Requirement already satisfied: smart-open<7.0.0,>=5.2.1 in /usr/local/li
b/python3.10/dist-packages (from spacy>=3.0.0->spacy_language_detection)
(6.3.0)
Requirement already satisfied: tqdm<5.0.0,>=4.38.0 in /usr/local/lib/pyt
hon3.10/dist-packages (from spacy>=3.0.0->spacy_language_detection) (4.6
5.0)
Requirement already satisfied: numpy>=1.15.0 in /usr/local/lib/python3.1
0/dist-packages (from spacy>=3.0.0->spacy_language_detection) (1.22.4)
Requirement already satisfied: requests<3.0.0,>=2.13.0 in /usr/local/li
b/python3.10/dist-packages (from spacy>=3.0.0->spacy_language_detection)
(2.27.1)
Requirement already satisfied: pydantic!=1.8,!=1.8.1,<1.11.0,>=1.7.4 in
/usr/local/lib/python3.10/dist-packages (from spacy>=3.0.0->spacy_langua
ge_detection) (1.10.7)
Requirement already satisfied: jinja2 in /usr/local/lib/python3.10/dist-
packages (from spacy>=3.0.0->spacy_language_detection) (3.1.2)
Requirement already satisfied: setuptools in /usr/local/lib/python3.10/d
ist-packages (from spacy>=3.0.0->spacy_language_detection) (67.7.2)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python
3.10/dist-packages (from spacy>=3.0.0->spacy_language_detection) (23.1)
Requirement already satisfied: langcodes<4.0.0,>=3.2.0 in /usr/local/li
b/python3.10/dist-packages (from spacy>=3.0.0->spacy_language_detection)
(3.3.0)
Requirement already satisfied: typing-extensions>=4.2.0 in /usr/local/li
b/python3.10/dist-packages (from pydantic!=1.8,!=1.8.1,<1.11.0,>=1.7.4->
spacy>=3.0.0->spacy_language_detection) (4.5.0)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in /usr/local/lib/p
ython3.10/dist-packages (from requests<3.0.0,>=2.13.0->spacy>=3.0.0->spa
```

```
cy_language_detection) (1.26.15)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/pyth
on3.10/dist-packages (from requests<3.0.0,>=2.13.0->spacy>=3.0.0->spacy_
language_detection) (2022.12.7)
Requirement already satisfied: charset-normalizer~=2.0.0 in /usr/local/l
ib/python3.10/dist-packages (from requests<3.0.0,>=2.13.0->spacy>=3.0.0-
>spacy_language_detection) (2.0.12)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.1
0/dist-packages (from requests<3.0.0,>=2.13.0->spacy>=3.0.0->spacy_langu
age_detection) (3.4)
Requirement already satisfied: blis<0.8.0,>=0.7.8 in /usr/local/lib/pyth
on3.10/dist-packages (from thinc<8.2.0,>=8.1.8->spacy>=3.0.0->spacy_lang
uage_detection) (0.7.9)
Requirement already satisfied: confection<1.0.0,>=0.0.1 in /usr/local/li
b/python3.10/dist-packages (from thinc<8.2.0,>=8.1.8->spacy>=3.0.0->spac
y_language_detection) (0.0.4)
Requirement already satisfied: click<9.0.0,>=7.1.1 in /usr/local/lib/pyt
hon3.10/dist-packages (from typer<0.8.0,>=0.3.0->spacy>=3.0.0->spacy_lan
guage_detection) (8.1.3)
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python
3.10/dist-packages (from jinja2->spacy>=3.0.0->spacy_language_detection)
(2.1.2)
Building wheels for collected packages: langdetect
  Building wheel for langdetect (setup.py) ... done
  Created wheel for langdetect: filename=langdetect-1.0.9-py3-none-any.w
hl size=993224 sha256=1c806026719d3548e9f9cecd60bd88a5d6fc4d05346929ecd9
a1b58d98146d1d
  Stored in directory: /root/.cache/pip/wheels/95/03/7d/59ea870c70ce4e5a
370638b5462a7711ab78fba2f655d05106
Successfully built langdetect
Installing collected packages: langdetect, spacy_language_detection
Successfully installed langdetect-1.0.9 spacy_language_detection-0.2.1
```

Jeżeli pojawi się informacja o konieczności ponownego uruchomienia środowiska wykonawczego - możesz wczytać zapisany wcześniej plik.

In [25]:
```python
import spacy
from spacy.language import Language
from spacy_language_detection import LanguageDetector

def get_lang_detector(nlp, name):
    return LanguageDetector(seed=42)  # We use the seed 42


nlp_model = spacy.load("en_core_web_sm")
# Language.factory("language_detector", func=get_lang_detector)
# nlp_model.add_pipe('language_detector', last=True)
```

Zarejestruj rozszerzenie - detektor języka (tylko raz podczas sesji).

Obiekt `nlp_model` jest singletonem zawierającym min. model języka oraz standardowy ciąg przetwarzania tekstu składający się z:

- tokenizera (czyli odpowiednik klasy Vectorizer)
- taggera (przypisywanie znaczników części mowy do symboli)
- parsera (buduje graf syntaktycznych zależności)
- NER - rozpoznawanie nazwanych terminów (takich, jak Bill Gates lub USA)

Language detection to dodatkowy etap, jego powtórne zarejestrowanie powodowałoby błędy.

In [26]:
```
# Zarejestruj rozszerzenie - detektor języka (tylko raz podczas sesji)

Language.factory("language_detector", func=get_lang_detector)
nlp_model.add_pipe('language_detector', last=True)
```

Out[26]:
```
<spacy_language_detection.spacy_language_detector.LanguageDetector at 0x7f8326759030>
```

Zobacz, jak działa rozpoznawanie języka...

In [27]:
```
# Document level language detection
job_title = "Senior NLP Research Engineer"
doc = nlp_model(job_title)
language = doc._.language
print(language)
```

```
{'language': 'en', 'score': 0.9999944616311092}
```

In [28]:
```
# Sentence level language detection
text = """This is English text.
        Er lebt mit seinen Eltern und seiner Schwester in Berlin.
        Yo me divierto todos los días en el parque.
        Je m'appelle Angélica Summer, j'ai 12 ans et je suis canadienne.
        Cześć, jestem Jan Kowalski z Warszawy.
        Ahoj, tady Pavel.
        Dobrý deň, tu je Pavel."""
doc = nlp_model(text)
for i, sent in enumerate(doc.sents):
    print(sent, sent._.language)
```

```
This is English text.
         {'language': 'en', 'score': 0.9999987929307772}
Er lebt mit seinen Eltern und seiner Schwester in Berlin.
         {'language': 'de', 'score': 0.999996045846908}
Yo me divierto todos los días en el parque.
         {'language': 'es', 'score': 0.9999960751128255}
Je m'appelle Angélica Summer, j'ai 12 ans et je suis canadienne.
         {'language': 'fr', 'score': 0.9999960488878061}
Cześć, jestem Jan Kowalski z Warszawy.
         {'language': 'pl', 'score': 0.9999984483477706}
Ahoj, tady Pavel.
         {'language': 'cs', 'score': 0.9999944828639359}
Dobrý deň, tu je Pavel. {'language': 'sk', 'score': 0.7440013197807269}
```

Jak są rozpoznawane zgromadzone teksty?

TODO 11.2.2.1 Wyznacz język dla dowolnych 100 tekstów zgromadzonych w tabeli `df`.

In [29]:
```
texts = df['text'].head(100)

for t in texts:
    doc = nlp_model(t)
    language = doc._.language['language']
```

```
    print(language, end=" ")
    print(t)
```

ca qc
pl czy ktoś korzystał z programu zwrotów za detailed review na cssbuy i może wyjaśnić jak to działa? chodzi mi dokładnie o podpunkt z 2 zdjęcia napisałem post na reddicie i zastanawiam się gdzie podać info na podstaw ie którego css będzie mógł przelać kasę na moje konto
ca qc
pl pytanie siema, 2 koszulki łącznie 800g myslicie ze warto brać gd-eub?
pl za ile paka moze byc i czy nic podejrzanego sie nie dzieje ? https://preview redd it/ouzaa690p2xa1 png?width 1054&format png&auto web p&v enabled&s 6d5d07230496ebe766fac73768cae6d677ead6d2
pl czym wyslac problem z cssbuy siema, ogolnie to kiedys w 2018 zamawial em rzeczy z chin superbuy i teraz po przerwie potrzebuje zamowic buty pr oblem z tym ze tutja polecacie cssbuy i leci ono w chuja od dwoch tygodn i paka miala status wyslanej do warehouse a po napisaniu do supportu odp isali, ze sprzedawca jeszcze jej nie wyslal wiec dalem refunda na ktoreg o sie zgodzili i czekam na hajs: pytanie 1 dostane kase do tygodnia tak jak to maja zapisane czy znowu leca w chuja, ktos wie jak to wyglada? py tanie 2 teraz kupilem to samo na superbuy, paka po 3 dniach juz w magazy nie xd , pytanie jak wyslac buty o wadze okolo 3kg zeby wszystko sprawni e poszlo? pomozcie prosze
pl siema podpowiedziałby mi ktoś na pv odnośnie przesyłki z 🐼 chodzi o deklaracje, przesyłkę itp
pl qc na te yeezuski, co myślicie?
pl rl czy gl? z góry thank u za pomoc batch nieznany :/
sl paka dhl 6kg [removed]
pl qc atrybutu każdego drillowca 🔥🔪
pl ktoś wie gdzie i do kogo mam się zwrócić w sprawie tego?
it [qc] polo moncler 😻😻😻
pl ma ktoś linka do tego? siema, potrzbuje link do tego
pl ma ktoś jakieś linki 1:1 butów z dhgate?
pl bache wytłumaczy ktoś te bache np : pk og itp
ca [qc] jordan 4 black cat
pl podrzucam wiecej zdjec bratki pomocy
pl w ile dni wam przyszla paka wyslana dhl na cssbuy ? i jakie wrażenia ? dobrej majówki życzę
pl pomocy siema miski interesuje sie już fejami od dłuższego czasu ale z a chûja nie wiem jak zamawiac, chciałbym sobie sprawic nike tn na wakacj e ale nie mam zielonego pojecia jak to zamówić, od kogo kupowac w miare wiem ale z zamówieniem juz moge miec problem i szukam kogos kto by mi to wytłumaczyl od podstaw jak dla debila :
pl cześć co mam zrobić jeżeli moją paczkę z pandabuy zatrzymała odprawa celna ?
sq qc revenge t-shirt
sq qc revenge t-shirt
pl przesyłka dhl status [removed]
pl odradzam kazdemu zamawianie od goata!!! tragedia! 5 raz wymieniam
pl poprosze o qc przyjaciele
sl n [removed]
pl trapstar bag 1 0 qc hej prosze o qc na tego baga :
en [qc] palm angels shark classic t-shirt
pl jaki batch polecacie zamawiać aj1x travis? zależy mi żeby stosunek ja kość- cena były spoko
pl czy wysyłanie zegarka eub-em to dobry pomysł?
en qc olive revers mocha travis scott jordan 1 low fk siema, wrzucam fot ki qc olive i revers mocha fk od tmf https://imgur io/a/lbobxwq https:// imgur io/a/i9rghk6
pl ktoś wytłumaczy lub odeślę o co chodzi z batch czyn się różnią jestem w tym zielony
sv qc jordan 1 high dark mocha
pl dostawa do czego moga sie doczepic na granicy kupujac: bluze stone, b

luze ralph, af1, yeezy i czapke cp jakies rady? bo to moje pierwsze zaku
py na takich stronach, czym wysyłać? i jak to robic zeby nie byto zadneg
o problemu po drodze
en dostawa [removed]
en qc burberry bluza
en pandabuy [removed]
pl co mogę zrobić? zignorować to i stracić, czy spróbować?
pl można kupić blikiem na hagobuy?
sk ma ktoś link do spodenek trapstar rozmiar s
et siema ma ktoś linka na tees i track pants od nike tn?
pl deklaracja wartości przesyłki i wysylka [removed]
pl wymiana fejurami wwa ma ktoś dobre podróbki air maxow w rozmiarze oko
ło 42 i chciałby się wymienić za fake yeezy boost 350 v2 core black rozm
iar okolo 42 dobra jakoś
en qc na jordany 4 edycja what the pls na moje oko zajebiscie ale nwm ht
tps://preview redd it/w0s1hn5y6hwa1 jpg?width 4032&format pjpg&auto webp
&v enabled&s 6b91f9da7337169a58f23bbb575ce75c546f1c47 https://preview re
dd it/au8z26gy6hwa1 jpg?width 4032&format pjpg&auto webp&v enabled&s fb0
5c1dcb425f5a407ac803ce0432803ebd8909a https://preview redd it/gphxc7gy6h
wa1 jpg?width 4032&format pjpg&auto webp&v enabled&s 69481192a30d93b979e
3f30ee2c98984db8150f5 https://preview redd it/x0yd87gy6hwa1 jpg?width 40
32&format pjpg&auto webp&v enabled&s 855d238fba7bf4ebd7b5328616225e0105c
66377 https://preview redd it/gh66k7gy6hwa1 jpg?width 4032&format pjpg&a
uto webp&v enabled&s 4b98599d2f3b8c7cbaa4c395a1b93a102cdd837d
ca qc
pl siemka która opcja przesyłki najlepsza żeby było szybko i „bezpieczni
e” dhl? dpd? [removed]
pl zatrzymmali pake pomoze ktos
pl qc na te budzetowe jordan4
en qc fog essential hoodie [removed]
en travis scott x air jordan 1 low top- qc
pl pytanko siemanko wszystkim, mam pytanie odnosnie replik vapormaxow pl
us, jaki jest najlepszy batch oraz czy warto pozdrawiam
vi qc?
pl gd-e-ems 5 8 kg yo ziomki, pierwszy haul do polski krk , gd e ems 5 8
kg, deklaruje 69 5$, ioss 118 45 cny, z uznanych brandow buty balenciaga
i spodnie burberry, wydalem okolo 170$ za wszystko bez dostawy , czy bed
zie git?
pl gd-e-ems vat siemano : chce zapytac u was jak sie liczy vat czy ioss,
i z czym w ogole moge sie spotkac przy dostawie posrednictwem gd-e-ems m
am 5 8 kg, sporo oszczedzam korzystajac z gd-e-ems w porownaniu z europe
tariffless line-b wiec jestem zainteresowany zeby wyslac taniej oczywisc
ie, i czy to sie oplaca na przedmiot dodatkowych oplat celnych : dzieki
pl pomoc siema, chciałbym zamówić pakę około 17kg i potrzebuje pomocy ja
k to ogarnąć żeby bezproblemowo dotarło rzeczy w pace to jakieś nike, tr
apstar, essential, jordan
pl qc yeezy 350 lw batch
pl czy ta kurtka istnieje? czy jest to stworzony byt przez chińczyków? j
edyne miejsce gdzie udało mi się ją znaleźć : https://preview redd it/ou
m6ijdwqawa1 png?width 1156&format png&auto webp&v enabled&s ba1d0a8a3c28
228ee7c01bffab58c409f42ed672 https://preview redd it/bkukdewirawa1 png?w
idth 1151&format png&auto webp&v enabled&s 48b4605afb097395b498d13b800a1
e8cfb066482
pl pytanko do doświadczonych wojowników [removed]
pl salehe bembury crocsy ktoś coś dobre repy? miło by było jakby był duż
y wybór kolorów :
pl siema, czy z moją paczką jest wszystko w porządku? waga paczki to 130
0g pierwszy raz zamawialem coś z pandabuy i trochę się martwie
pl oryginał czy podróbka af1
pl witam, zrobi ktos qc na yeezy 350 zebra? batch g5

pl uzupełnienie dokumentów w odprawie celnej podesłałem im na początku s
creenshot z paypala, teraz babka mi każe jeszcze te rzeczy podesłać tylk
o do końca nie wiem jak zrobić, jakiś tip?: * potwierdzenie zamówienia z
rzut ze strony sklepu https://preview redd it/5j17g2qhl9wa1 jpg?width 95
2&format pjpg&auto webp&v enabled&s ed260413cdf76c6e3e53eac4d8d1e3a51114
bf91
en najlepszy batch na travis scott x air jordan 1 low og olive best batc
h bratku
pl zgłaszanie celne jest tu może ktoś kto pomógł by mi wypełnić list do
zgłoszenia celnego tak żeby była jak największą szansa dostać paczkę bo
pierwszy raz to będę robił i wolę dopytać
pl lc na bluze vlone cześć! znalazłem bluze vlone na vinted i nwm czy to
fake jest czy legit potrzebuje odpowiedz asap
pl wysyłka 4 parek [removed]
en qc yeezy 350 onyx [removed]
en qc yeezy 350 lol2021 [removed]
sk qc na te travisy
pl znacie jakiegoś sprzedawcę z jesnsami luźniejszymi nie baggy?
pl wie ktoś gdzie kupie repy korków piłkarskich?
pl koszulki piłkarskie wiecie może u kogo można kupić dobre jakościowo k
oszulki piłkarskie?
en trapstar summer set from goat qc??
id qc - jordan 1 obsidian cz batch
pl szukam kogos kto sie zna na listach potrzebuje pomocy z tym pierwszy
raz dostalem i idk
pl potrzebuje kogoś z aktywnym kontem wechat siemanko, posiada ktoś akty
wne konto wechat i pomógł by mi je potwierdzić przy rejestracji?
pl eub gdzie przekierować paczkę [removed]
pl w czym jest lepszy cssbuy on pandabuy?
pl w2c dokładnie taki sam wariant kolorów, z szarym jumpmanem na tongue
en agent [removed]
ca qc essentials, af1 x drake
pl czym teraz shipować? będę niedługo shipowac wielką pake 2 parki jorda
nów z boxami, 4 pary klapek i tracki zastanawiam sie czym wysłać i jak t
o zadeklarować możecie mi jakoś doradzić?
pl gx po pięciu dniach noszenia xd ponoć w legitach też sie robi crackin
g na midsolach ale bez przesady ze po 5 dniach, zamowilem juz farbe ange
lus white i finisher matte i mam pytanie czy normalnie moge malowac tylk
o białą czesc midsolu czy musze tez czarne zmywac by sie git trzymalo
en somebody wanna help with custom clearance? blik/revolut payment for s
uccessful helper no ioss, e-ems text me on reddit or discord simozwloch#
2504
pl pokażcie swoje najdrozsze itemy
pl paka w chinach siemano, mam problem z paczka bo już 22 stoi w chinach
i nic się nie zmienia nigdy nie miałem sytuacji w której zajmowało to ty
le czasu żadnych maili, powiadomien na css czy komuś też się zdarzyło że
by tak długo paka stała w miejscu na praktycznie początku dostawy?
en qc yeezy 350 black reflective od lol2021
pl szukam tego sharka, size l/m
pl qc bapesta baby blue poprosze qc na te bapesty imgur: w2c:
en ship [removed]
pl jaki batch najlepszy dla aj5?
en qc ts mocha jordan one high - og batch - jmdy - 169 yuan qc https://p
review redd it/c3dqespfsvva1 jpg?width 960&format pjpg&auto webp&v enabl
ed&s 485a134eeaa79c82f12efbff4831e3aac25259a5 https://preview redd it/jy
tiwtpfsvva1 jpg?width 960&format pjpg&auto webp&v enabled&s 26240b627b14
015b78a0965681821a3dd2326bf8 https://preview redd it/mv3tiwpfsvva1 jpg?w
idth 960&format pjpg&auto webp&v enabled&s 95d67143d54632d97a50002874d8b
162fa1999d7 https://preview redd it/3g5sctpfsvva1 jpg?width 960&format p
jpg&auto webp&v enabled&s 36470a8acd1d19190ae044067a9f3e551dd37d79 http

```
s://preview redd it/aotdawpfsvva1 jpg?width 960&format pjpg&auto webp&v
enabled&s 465140db7d06b12505cb419f4833bb4c13f570bb https://preview redd
it/25wwqaqfsvva1 jpg?width 960&format pjpg&auto webp&v enabled&s 8b45659
fe783df09096ad13dbd81326fa8cd58e2 https://preview redd it/t4b8u1rfsvva1
jpg?width 960&format pjpg&auto webp&v enabled&s 4e8d88ae0f9fe715242232a5
9a61789f322facb5 https://preview redd it/aaxjq1qfsvva1 jpg?width 960&for
mat pjpg&auto webp&v enabled&s 9ff775f74764cfa218b5bc2a9693ddfd5b170eaf
fr e-ems czy europe tariffless lines? &#x200b;
hu szukam, fit185
en zrobi ktos qc na te white cement'y
en ship [removed]
pl siema, pierwszy raz mam taki dziwny tracking, wszystko okej czy liści
k będzie? 2kg eub
sk siemano, zrobi ktos qc aj4 military black? batch sk
```

TODO 11.2.2.2 Napisz funkcję, accept_language(t). Funkcja ma zwrócić True, jeżeli rozpoznanym językiem był polski, słowacji, czeski (i także może hu oraz hr).

In [30]:
```
#zostawmy pl, sk, cs (bo podobne) i hu oraz hr

def accept_language(text):
    if isinstance(text, str):
      doc = nlp_model(text)
      language = doc._.language['language']

      accepted_languages = ['pl', 'sk', 'cs', 'hu', 'hr']

      if language in accepted_languages:
          return True
      return False

df[:10]['text'].apply(accept_language)
```

Out[30]:
```
0    False
1     True
2    False
3     True
4     True
5     True
6     True
7     True
8     True
9    False
Name: text, dtype: bool
```

TODO 11.2.2.3 Wykorzystaj wartość zwracaną przez df.text.apply(accept_language), aby utworzyć ramkę df_pl z tekstami w akceptowanych językach

In [31]:
```
print(df['text'])
```

```
0                                                    qc
1       czy ktoś korzystał z programu zwrotów za detai...
2                                                    qc
3       pytanie siema, 2 koszulki łącznie 800g myslici...
4       za ile paka moze byc i czy nic podejrzanego si...
                              ...
2891    tourists eating their overpriced food around a...
2892    w krakowie uruchomiono "kacze bufety" najlepsz...
2893    aktywiści: "budować tramwaj na alei słowackieg...
2894    odkopane z kwietnia, kopiec krakusa podejrzewa...
2895    calling all dj’s i am looking for dj’s to play...
Name: text, Length: 2896, dtype: object
```

In [34]:
```
# df_pl = df[:2895]['text'].apply(accept_language)
df = df.dropna()
df_pl = df[df['text'].apply(accept_language)]
df_pl.head()
```

Out[34]:

	Unnamed: 0	text	subreddit	link_flair_text
7	7	qc na te yeezuski, co myślicie?	FashionRepsPolska	LCQC
8	8	rl czy gl? z góry thank u za pomoc batch niezn...	FashionRepsPolska	LCQC
10	10	qc atrybutu każdego drillowca 🔥🔪	FashionRepsPolska	LCQC
13	13	ma ktoś linka do tego? siema, potrzbuje link d...	FashionRepsPolska	LCQC
15	15	bache wytłumaczy ktoś te bache np : pk og itp	FashionRepsPolska	LCQC

2.3 Ekstrakcja etykiet

TODO 11.2.3.1 Analogicznie, jak wcześniej utwórz zbiór etykiet

In [35]:
```
from sklearn import preprocessing

le = preprocessing.LabelEncoder()
y=le.fit_transform(df_pl['subreddit'])

le.classes_
```

Out[35]:
```
array(['FashionRepsPolska', 'PolskaPolityka', 'krakow'], dtype=object)
```

2.4 Wyznaczamy stopwords - słowa funkcyjne dla języka polskiego

TODO 11.2.4.1 Wyznacz 30 słów funkcyjnych na podstawie atrybutu vocabulary_

In [59]:
```
from sklearn.feature_extraction.text import TfidfVectorizer

vectorizer = CountVectorizer(max_features=30)
vectorizer.fit(df_pl.text)
```

```
functional_words = vectorizer.vocabulary_
functional_words
```

Out[59]:
```
{'na': 12,
 'co': 4,
 'czy': 5,
 'za': 28,
 'ktoś': 10,
 'do': 7,
 'tego': 25,
 'od': 14,
 'jest': 9,
 'to': 26,
 'przez': 20,
 'się': 22,
 'po': 16,
 'ale': 0,
 'dla': 6,
 'jak': 8,
 'są': 23,
 'nie': 13,
 'może': 11,
 'chce': 3,
 'tym': 27,
 'tak': 24,
 'będzie': 2,
 'że': 29,
 'pis': 15,
 'bardzo': 1,
 'polska': 18,
 'polsce': 17,
 'polski': 19,
 'rosja': 21}
```

In [60]:
```
stopwords=[k for k in functional_words]
```

2.5 Wybór konfiguracji dla obiektu Vectorizer

TODO 11.2.5.1

- Analogicznie, jak wcześniej eksperymentalnie wybierz konfigurację dla klasy TfidfVectorizer (ze stopwords lub bez). Wyniki mogą być rózne w zależności od pobranych postów...

In [61]:
```
from sklearn.naive_bayes import MultinomialNB
from sklearn.pipeline import make_pipeline
from sklearn.model_selection import train_test_split
import sklearn

from sklearn.svm import SVC

# vectorizer =
# vectorizer =
# vectorizer =
# vectorizer = ???

vectorizer = TfidfVectorizer(analyzer='word',ngram_range=(1,2),max_featur
```

```
cls = MultinomialNB()

pipeline = make_pipeline(vectorizer, cls)
for i in range(10):
  X_train, X_test, y_train, y_test = train_test_split(df_pl.text, y, test
  pipeline.fit(X_train,y_train)
  y_pred = pipeline.predict(X_test)
  print(f'Accuracy:{sklearn.metrics.accuracy_score(y_test,y_pred)} F1: {s
```

```
Accuracy:0.8833333333333333 F1: 0.4604604604604605
Accuracy:0.95 F1: 0.7133726647000983
Accuracy:0.9333333333333333 F1: 0.5897209985315712
Accuracy:0.9166666666666666 F1: 0.5409356725146198
Accuracy:0.9 F1: 0.5057797708021924
Accuracy:0.8416666666666667 F1: 0.3724770642201835
Accuracy:0.9083333333333333 F1: 0.4666005291005291
Accuracy:0.9083333333333333 F1: 0.5123917748917749
Accuracy:0.8583333333333333 F1: 0.30930930930930933
Accuracy:0.925 F1: 0.57
```

2.6 Walidacja krzyżowa i wyświetlanie macierzy pomyłek

In [39]:
```
# Funkcja pobrana z https://github.com/DTrimarchi10/confusion_matrix/blob

import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

def make_confusion_matrix(cf,
                          group_names=None,
                          categories='auto',
                          count=True,
                          percent=True,
                          cbar=True,
                          xyticks=True,
                          xyplotlabels=True,
                          sum_stats=True,
                          figsize=None,
                          cmap='Blues',
                          title=None):
    '''
    This function will make a pretty plot of an sklearn Confusion Matrix
    Arguments
    ---------
    cf:            confusion matrix to be passed in
    group_names:   List of strings that represent the labels row by row t
    categories:    List of strings containing the categories to be displa
    count:         If True, show the raw number in the confusion matrix.
    normalize:     If True, show the proportions for each category. Defau
    cbar:          If True, show the color bar. The cbar values are based
                   Default is True.
    xyticks:       If True, show x and y ticks. Default is True.
    xyplotlabels:  If True, show 'True Label' and 'Predicted Label' on th
    sum_stats:     If True, display summary statistics below the figure.
    figsize:       Tuple representing the figure size. Default will be th
    cmap:          Colormap of the values displayed from matplotlib.pyplo
```

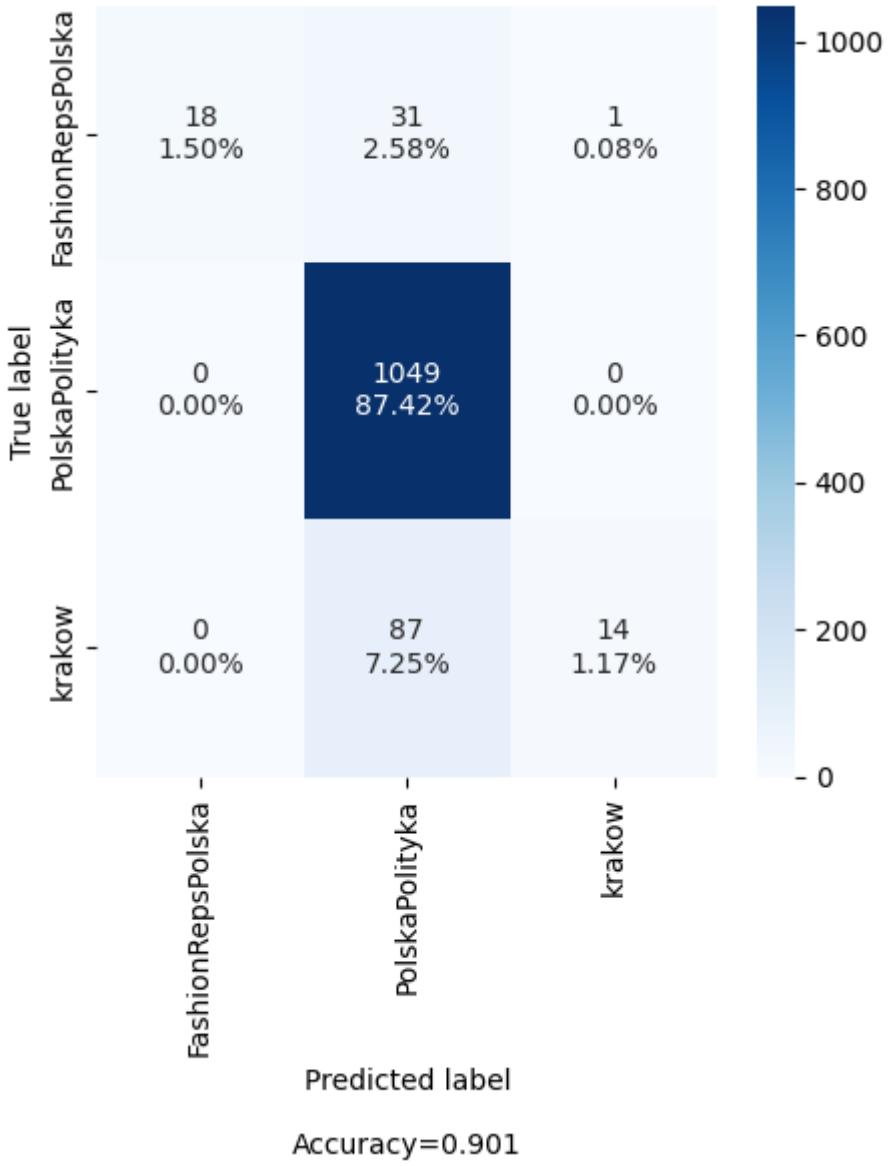
```
                See http://matplotlib.org/examples/color/colormaps_ref

    title:      Title for the heatmap. Default is None.
    '''


    # CODE TO GENERATE TEXT INSIDE EACH SQUARE
    blanks = ['' for i in range(cf.size)]

    if group_names and len(group_names)==cf.size:
        group_labels = ["{}\n".format(value) for value in group_names]
    else:
        group_labels = blanks

    if count:
        group_counts = ["{0:0.0f}\n".format(value) for value in cf.flatte
    else:
        group_counts = blanks

    if percent:
        group_percentages = ["{0:.2%}".format(value) for value in cf.flat
    else:
        group_percentages = blanks

    box_labels = [f"{v1}{v2}{v3}".strip() for v1, v2, v3 in zip(group_lab
    box_labels = np.asarray(box_labels).reshape(cf.shape[0],cf.shape[1])


    # CODE TO GENERATE SUMMARY STATISTICS & TEXT FOR SUMMARY STATS
    if sum_stats:
        #Accuracy is sum of diagonal divided by total observations
        accuracy  = np.trace(cf) / float(np.sum(cf))

        #if it is a binary confusion matrix, show some more stats
        if len(cf)==2:
            #Metrics for Binary Confusion Matrices
            precision = cf[1,1] / sum(cf[:,1])
            recall    = cf[1,1] / sum(cf[1,:])
            f1_score  = 2*precision*recall / (precision + recall)
            stats_text = "\n\nAccuracy={:0.3f}\nPrecision={:0.3f}\nRecall
                accuracy,precision,recall,f1_score)
        else:
            stats_text = "\n\nAccuracy={:0.3f}".format(accuracy)
    else:
        stats_text = ""


    # SET FIGURE PARAMETERS ACCORDING TO OTHER ARGUMENTS
    if figsize==None:
        #Get default figure size if not set
        figsize = plt.rcParams.get('figure.figsize')

    if xyticks==False:
        #Do not show categories if xyticks is False
        categories=False


    # MAKE THE HEATMAP VISUALIZATION
    plt.figure(figsize=figsize)
    sns.heatmap(cf,annot=box_labels,fmt="",cmap=cmap,cbar=cbar,xticklabel
```

```
    if xyplotlabels:
        plt.ylabel('True label')
        plt.xlabel('Predicted label' + stats_text)
    else:
        plt.xlabel(stats_text)

    if title:
        plt.title(title)
```

TODO 11.2.6.1 Wywołaj funcję `cross_val_predict` dostarczając do niej odpowiednie dane (pipeline, teksty i etykiety). Wyznacz macierz pomyłek i wyświetl ją.

In [40]:
```
from sklearn.model_selection import cross_val_predict
from sklearn.metrics import confusion_matrix


y_pred = cross_val_predict(pipeline, df_pl['text'], y, cv=10)
conf_mat = confusion_matrix(y, y_pred)
```

In [41]:
```
plt.rcParams["figure.figsize"] = (5,5)
make_confusion_matrix(conf_mat,categories=le.classes_);
```



2.7 Kiedy klasyfiaktor pomylił się

Z `df_pl` wybierz teksty, dla których było sporo błednych klasyfikacji. W naszym przykładzie może to b yć `krakow`.

In [42]:
```
df_pl.head()
```

Out[42]:

	Unnamed: 0	text	subreddit	link_flair_text
7	7	qc na te yeezuski, co myślicie?	FashionRepsPolska	LCQC
8	8	rl czy gl? z góry thank u za pomoc batch niezn...	FashionRepsPolska	LCQC
10	10	qc atrybutu każdego drillowca 🔥🔪	FashionRepsPolska	LCQC
13	13	ma ktoś linka do tego? siema, potrzbuje link d...	FashionRepsPolska	LCQC
15	15	bache wytłumaczy ktoś te bache np : pk og itp	FashionRepsPolska	LCQC

In [43]:
```
df_kr = df_pl[df_pl.subreddit=='krakow'].copy()
df_kr.head()
```

Out[43]:

	Unnamed: 0	text	subreddit	link_flair_text
1803	1803	znalezione na kazimierzu https://prev...	krakow	Photo
1804	1804	gryl w krakowie hej! majówka przyszła, więc se...	krakow	Question
1810	1810	fajne, mniej uczęszczane spoty na deskę? nie m...	krakow	Question
1814	1814	szukam ludzi do ankiety - poglądy, nastroje sp...	krakow	Question
1816	1816	życie wróciło do parku bednarskiego otwarty w ...	krakow	Local news

In [44]:
```
pipeline.predict(df_kr.text)
```

Out[44]:
```
array([2, 2, 2, 1, 1, 2, 1, 1, 1, 1, 1, 2, 1, 2, 2, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 2, 1, 2, 2, 2, 1, 1, 1, 2, 2, 1, 1, 2, 1, 1, 1, 1, 1,
       2, 1, 1, 2, 1, 2, 2, 2, 2, 2, 2, 1, 1, 1, 2, 2, 1, 1, 1, 1, 2, 1,
       1, 1, 1, 2, 1, 2, 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 2, 1, 1, 2, 1,
       1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1])
```

TODO 11.2.7.1 Dokonaj klasyfkiacji danych w `df_kr`. Wyniki w postaci etykiet tekstowych dodaj jako kolumnę o nazwie `pred`.

In [45]:
```
vectorizer = TfidfVectorizer(analyzer='word', ngram_range=(1, 2), max_fea

X_train = df_pl['text']
y_train = y

X_test = df_kr['text']

vectorized_X_train = vectorizer.fit_transform(X_train)
vectorized_X_test = vectorizer.transform(X_test)

cls = MultinomialNB()
cls.fit(vectorized_X_train, y_train)
p = cls.predict(vectorized_X_test)

df_kr['pred'] = p
print(df_kr.head())
```

```
      Unnamed: 0                                               text subr
eddit  \
1803        1803  znalezione na kazimierzu &#x200b; https://prev...    k
rakow
1804        1804  gryl w krakowie hej! majówka przyszła, więc se...    k
rakow
1810        1810  fajne, mniej uczęszczane spoty na deskę? nie m...    k
rakow
1814        1814  szukam ludzi do ankiety - poglądy, nastroje sp...    k
rakow
1816        1816  życie wróciło do parku bednarskiego otwarty w ...    k
rakow

     link_flair_text  pred
1803           Photo     2
1804        Question     2
1810        Question     1
1814        Question     2
1816      Local news     1
```

TODO 11.2.7.2 utwórz tabelę danych zawierającą tylko te wiersze, dla których etykiety subreddit i pred są rózne.

Przeglądnij przykłady i dla kilku z nich spróbuj uzasadnić decyzję klasyfikatora.

Klasyfikator może mieć trudności w przewidywaniu odpowiedniej etykiety w przypadku tekstów o podobnej zawartości (dotyczace tego samego tematu). Na przykład, teksty dotyczące krakowskiego parku Bednarskiego (1816) i Krakowa Głównego (2864) mogą mieć podobne treści, ale różnią się słownictwem lub stylem, co może wprowadzać klasyfikator w błąd. Rowniez nalezy wziac pod uwagę subiektywność klasyfikacji.

In [46]:
```
df_kr_diff = df_kr[df_kr['subreddit'] != df_kr['pred']]
df_kr_diff.head(100)
```

Out[46]:

	Unnamed: 0	text	subreddit	link_flair_text	pred
1803	1803	znalezione na kazimierzu https://prev...	krakow	Photo	2
1804	1804	gryl w krakowie hej! majówka przyszła, więc se...	krakow	Question	2
1810	1810	fajne, mniej uczęszczane spoty na deskę? nie m...	krakow	Question	1
1814	1814	szukam ludzi do ankiety - poglądy, nastroje sp...	krakow	Question	2
1816	1816	życie wróciło do parku bednarskiego otwarty w ...	krakow	Local news	1
...	...	...	...	...	...
2813	2813	na lekcje religii kraków wydaje 30 mln złotych...	krakow	Local news	1
2854	2854	trudna sytuacja krakowskiego mpk na czas nie d...	krakow	Local news	1
2864	2864	kraków główny	krakow	Photo	1
2877	2877	czy możecie polecić krakowskie galerie sztuki ...	krakow	Question	1
2884	2884	co ciekawego na poniedziałek? hejka, z powodu ...	krakow	Question	1

100 rows × 5 columns

2.8 Strojenie parametrów (grid search)

Przeprowadzimy strojenie parametru `alpha` klasyfikatora używając wartości z podanego zakresu.

In [47]:
```
alpha = np.linspace(0.001,1,100,endpoint=True)
print(alpha)
```

```
[0.001      0.01109091 0.02118182 0.03127273 0.04136364 0.05145455
 0.06154545 0.07163636 0.08172727 0.09181818 0.10190909 0.112
 0.12209091 0.13218182 0.14227273 0.15236364 0.16245455 0.17254545
 0.18263636 0.19272727 0.20281818 0.21290909 0.223      0.23309091
 0.24318182 0.25327273 0.26336364 0.27345455 0.28354545 0.29363636
 0.30372727 0.31381818 0.32390909 0.334      0.34409091 0.35418182
 0.36427273 0.37436364 0.38445455 0.39454545 0.40463636 0.41472727
 0.42481818 0.43490909 0.445      0.45509091 0.46518182 0.47527273
 0.48536364 0.49545455 0.50554545 0.51563636 0.52572727 0.53581818
 0.54590909 0.556      0.56609091 0.57618182 0.58627273 0.59636364
 0.60645455 0.61654545 0.62663636 0.63672727 0.64681818 0.65690909
 0.667      0.67709091 0.68718182 0.69727273 0.70736364 0.71745455
 0.72754545 0.73763636 0.74772727 0.75781818 0.76790909 0.778
 0.78809091 0.79818182 0.80827273 0.81836364 0.82845455 0.83854545
 0.84863636 0.85872727 0.86881818 0.87890909 0.889      0.89909091
 0.90918182 0.91927273 0.92936364 0.93945455 0.94954545 0.95963636
 0.96972727 0.97981818 0.98990909 1.        ]
```

In [48]:
```
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report
from sklearn.model_selection import GridSearchCV
from sklearn.model_selection import cross_validate

params = {'multinomialnb__alpha':alpha}
grid = GridSearchCV(pipeline, params, scoring="f1_macro",cv=10, verbose=1
grid.fit(df_pl['text'], y)
grid.best_params_
```

```
Fitting 10 folds for each of 100 candidates, totalling 1000 fits
```

Out[48]:
```
{'multinomialnb__alpha': 0.05145454545454546}
```

TODO 11.2.8.1 Wykorzystaj wyznaczoną wartość `alpha` aby:

- przeprowadzić walidację krzyżową dla odpowiednio skonfigurowanego klasyfikatora
- obliczyć wartości średnie metryk
- wyświetlić wynikową macierz pomyłek

In [49]:
```
from sklearn.model_selection import cross_validate
cls = MultinomialNB(alpha=1.0)

pipeline = make_pipeline(vectorizer, cls)
scoring = ['accuracy','precision_macro','recall_macro','f1_macro']
cv_results = cross_validate(pipeline, df_pl['text'], y, cv=10,scoring=sco
print(cv_results)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.
py:1344: UndefinedMetricWarning: Precision is ill-defined and being set
to 0.0 in labels with no predicted samples. Use `zero_division` paramete
r to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
/usr/local/lib/python3.10/dist-packages/sklearn/metrics/_classification.
py:1344: UndefinedMetricWarning: Precision is ill-defined and being set
to 0.0 in labels with no predicted samples. Use `zero_division` paramete
r to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
{'fit_time': array([0.08659649, 0.09088469, 0.09165525, 0.0929122 , 0.09
914303,
       0.0912509 , 0.09081841, 0.09859657, 0.10091281, 0.07889628]), 'sc
ore_time': array([0.01075935, 0.00851178, 0.0107975 , 0.00843263, 0.0076
6397,
       0.00825357, 0.0075295 , 0.01035595, 0.00861788, 0.01568842]), 'te
st_accuracy': array([0.90833333, 0.9       , 0.9       , 0.9       , 0.9
3333333,
       0.90833333, 0.9       , 0.875     , 0.89166667, 0.89166667]), 'te
st_precision_macro': array([0.9683908 , 0.80172414, 0.96581197, 0.965811
97, 0.97640118,
       0.9683908 , 0.96581197, 0.29166667, 0.6299435 , 0.96296296]), 'te
st_recall_macro': array([0.53333333, 0.5       , 0.5       , 0.46666667,
0.7       ,
       0.5       , 0.5       , 0.33333333, 0.46666667, 0.4969697 ]), 'te
st_f1_macro': array([0.61832938, 0.56277383, 0.56639757, 0.53753754, 0.7
7124337,
       0.58169935, 0.56639757, 0.31111111, 0.50437754, 0.55975724])}
```

In [50]:
```
acc = cv_results['test_accuracy'].mean()
prec = cv_results['test_precision_macro'].mean()
recall = cv_results['test_recall_macro'].mean()
f1 = cv_results['test_f1_macro'].mean()

print(f'acc={acc} prec={prec} recall={recall} f1={f1}')
```

```
acc=0.9008333333333335 prec=0.8496915956957827 recall=0.4996969696969697
f1=0.5579624477933123
```

In [51]:
```
plt.rcParams["figure.figsize"] = (5,5)
make_confusion_matrix(conf_mat,categories=le.classes_);
```