AKADEMIA GÓRNICZO-HUTNICZA IM. STANISŁAWA STASZICA W KRAKOWIE

WYDZIAŁ INFORMATYKI, ELEKTRONIKI I TELEKOMUNIKACJI

INSTYTUT TELEKOMUNIKACJI

PRACA DYPLOMOWA

APLIKACJA KORELUJĄCA CENĘ WYBRANYCH AKCJI Z KOMUNIKATAMI TEKSTOWYMI NA PLATFORMIE TWITTER

AN APPLICATION TO CORRELATE STOCK PRICES WITH TWITTER POSTS

Autor:	*Jakub Wiercimak*
Kierunek studiow:	*Teleinformatyka*
Typ studiow:	*stacjonarne*
Opiekun pracy:	*dr inż. Jarosław Bułat*

Kraków, 2022

Spis treści

Wstęp

Celem wykonania niniejszej pracy inżynierskiej jest aplikacja korelująca ceny wybranych akcji z komunikatami tekstowymi na platformie Twitter. Psychologia uczestników rynku odgrywa kluczową rolę w inwestowaniu, a ponadto ma odzwierciedlenie w ruchach cen akcji. Ewolucja domów maklerskich doprowadziła do przyjmowania zleceń online, gdzie przy pomocy strony internetowej lub aplikacji mobilnej każdy może szybko założyć konto brokerskie oraz handlować różnymi instrumentami finansowymi w tym akcjami. Spowodowało to dynamiczny wzrost zainteresowania rynkiem akcji inwestorów detalicznych, przez co podejmowane przez nich decyzje oraz emocje jakimi się kierują przybrały na znaczeniu bardziej niż kiedykolwiek wcześniej.

Motywem przewodnim w wykonaniu takiej aplikacji było sprawdzenie jak reakcje użytkowników mediów społecznościowych wpływają na cenę wybranych akcji. Dzięki użyciu techniki uczenia maszynowego, eksploracji oraz wstępnego przetwarzania danych, możliwym jest skwantyfikowanie sentymentu panującego na rynku oraz zestawienie go z ceną danego waloru. Wykorzystując najnowsze rozwiązania w dziedzinie przetwarzania języka naturalnego rezultaty będą oddawały miarodajne nastroje panujące na rynku. Posiadając taką analizę, potencjalni inwestorzy są w stanie zmierzyć poziom euforii oraz strachu towarzyszący obrotowi danej akcji, co sprawia, że będą oni mogli podejmować lepsze decyzje inwestycyjne. Jedną z inspiracji do wykonania takiej aplikacji była akcja użytkowników Reddit przeciwko funduszom inwestycyjnym w styczniu 2021 roku, gdy poprzez skup akcji Gamestop doprowadzili fundusze zarządzające aktywami rzędu kilkunastu miliardów dolarów do zamknięcia krótkich pozycji (gry na spadek kursu) oraz strat sięgających kilku miliardów dolarów. Rezultatem był wzrost kursu akcji od 17 do blisko 350 dolarów w ciągu zaledwie kilkunastu dni. To wydarzenie pokazało siłę użytkowników mediów społecznościowych oraz uczyniło widocznym ich znaczącą rolę na rynku.

W pierwszym rozdziale zawarto szczegółowy opis wykorzystanego modelu do przetwarzania języka naturalnego - finBERT wraz z porównaniem skuteczności do klasycznych metod. Rozdział drugi oraz trzeci został poświęcony opisowi implementacji całego procesu przetwarzania danych. Ostatni rozdział to analiza otrzymanych wyników wraz z wnioskami oraz przetestowaniem przykładowej strategii inwestycyjnej opartej wyłącznie na analizie sentymentu.

1. Model przetwarzania języka naturalnego

Przetwarzanie języka naturalnego NLP (ang. *Natural Language Processing*) jest dyscypliną, której zadaniem jest rozpoznanie oraz „zrozumienie" języka naturalnego, tj. języka ludzkiego (np. polskiego, niemieckiego, angielskiego) w celu wykorzystania go do zadań praktycznych [1]. Zamiarem pracowników naukowych zajmujących się tym zagadnieniem jest zebranie wiedzy nt. tego jak człowiek przyswaja oraz przetwarza tekst. Dzięki temu możliwe jest tworzenie oraz doskonalenie oprogramowania przetwarzającego tekst a następnie jego analizę. Przetwarzanie języka naturalnego scala ze sobą wiele dyscyplin naukowych takich jak lingwistyka, matematyka oraz uczenie maszynowe [1]. Dzięki połączeniu wspomnianych dziedzin naukowych systemy komputerowe są w stanie przetwarzać język naturalny zarówno w formie tekstu jak i wiadomości głosowej oraz są w stanie „zrozumieć" znaczenie oraz kontekst zgodny z założeniami adresata. Istnieje wiele wyzwań związanych z analizą mowy oraz tekstu. W każdym języku naturalnym występuje niezliczona ilość dwuznaczności, homonimów, idiomów, które czynią pisanie oprogramowania NLP niezwykle skomplikowanym. Poprawne zastosowanie gramatyki w zdaniach lub identyfikowanie sarkazmu oraz metafor, na które przeciętna osoba poświęca lata nauki szkolnej jest niezbędne do poprawnego funkcjonowania użytecznego oprogramowania.

Aby uprościć implementację algorytmu NLP, dyscyplinę tą dzieli się na kilka zadań, które można wykonać po odpowiednim wytrenowaniu modelu. Jednym z nich jest rozpoznanie mowy, które ma na celu konwersję mowy na tekst pisany. Takie rozwiązania są stosowane w wielu aplikacjach oraz systemach, które wykorzystują zarówno głosowe wydawanie poleceń jak i odpowiedzi na zadane pytania. Największymi wyzwaniami tego obszaru badań jest zmierzenie się z niedoskonałością ludzkiej mowy, tzn. niezliczoną liczbą akcentów oraz często popełnianymi błędami gramatycznymi oraz składniowymi. Innym zadaniem NLP jest ujednoznacznienie sensu wyrazów, tak aby zostało nadane poprawne znaczenie analizowanemu słowu, przy pomocy analizy kontekstu. Przewidywanie kolejnego słowa (ang. *language modelling*) wykorzystywane m.in przy pisaniu wiadomości tekstowych w smartfonach oraz zdolność odpowiedzi na pytania zamknięte to inne zagadnienia możliwe do zrealizowania po odpowiednim wytrenowaniu modelu.

Ważnym zadaniem, które zostanie obszernie poruszone w tej pracy to analiza sentymentu tekstu, czyli przydzielenie odpowiedniego opisu emocji, które towarzyszy danemu tekstowi. Jednym z zastosowań takiego podejścia jest analiza mediów społecznościowych w celu określenia trendu nastrojów użytkowników, np. stopnia zadowolenia z danej usługi lub analizy mającej na celu sprawdzić jakie emocje towarzyszą odbiorcom najnowszej kampanii reklamowej. W tej pracy ta dziedzina NLP została zastosowana do zbadania sentymentu uczestników rynków finansowych oraz kryptowalut w celu oceny nastrojów panujących przy obrocie giełdowym na danym walorze.

Jedną z najbardziej popularnych technik konstruowania modelów w NLP to stosowanie tzw. osadzenia słów[1] (ang. *word embeddings*) [2]. Każde słowo jest reprezentowane wektorem, który reprezentuje znaczenie danego słowa na wielowymiarowej przestrzeni. Słowa, które leżą blisko

[1] stosuje się również terminy „zanurzenie" oraz „reprezentacja wektorowa słowa"

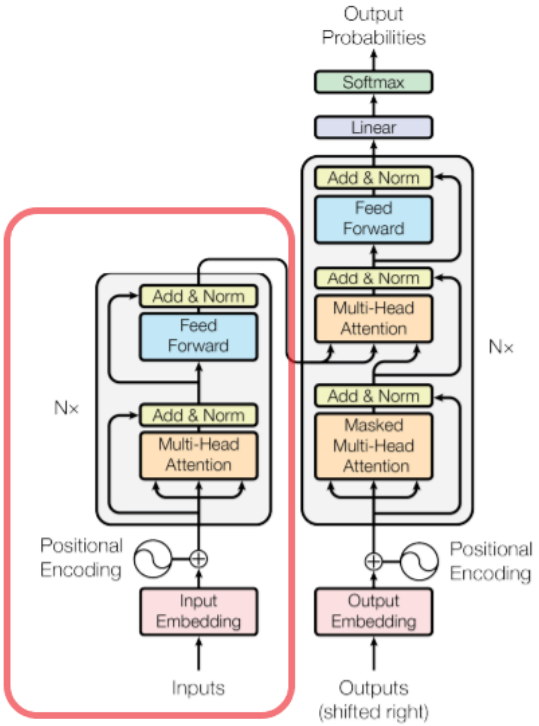
siebie są traktowane jako wyrazy o podobnym znaczeniu. Zaletą takiego rozwiązania jest dokładność w dziedzinie semantyki słów. Jeszcze innym rozwiązaniem, które zostało wykorzystane w tej pracy jest model Transformer, który zostanie opisany w poniższej sekcji.

1.1. Zasada działania modelu FINBERT

W pracy została zastosowana najnowsza implementacja modelu uczenia maszynowego finBERT, który jest modyfikacją modelu przetwarzania języka naturalnego - BERT (ang. *Bidirectional Encoder Representations from Transformers*). BERT to model językowy, który składa się z kilku enkoderów architektury Transformer (architektura enkodera zaznaczona na rysunku 1) ustawionych w stosie jeden na drugim [3]. Model po odpowiednim dostrojeniu (ang. *fine tuning*) jest dostosowany do takich zadań jak umiejętność odpowiedzi na zadane pytanie, analiza sentymentu czy też podsumowanie tekstu. Jest jednym z najnowocześniejszych modeli przetwarzania języka naturalnego. W zadaniach typu klasyfikacja tekstu czy też odpowiadanie na pytania osiąga jedne z najlepszych wyników w wymienionych gałęziach uczenia głębokiego.

Transformer jest architekturą NLP, której celem jest rozwiązanie zadań takich jak transformacja sekwencji do innej postaci (np. translacja między dwoma językami). Wykorzystuje koncept samo uwagi (ang. *self-attention*) w celu obliczenia reprezentacji wartości liczbowych sekwencji wejściowych oraz wyjściowych bez używania sieci rekurencyjnych oraz splotu [4]. Na architekturę Transformer składa się enkoder oraz dekoder, przedstawiony na rysunku 1, na wejściu enkodera podawane jest zdanie, które enkoder dzieli na wyrazy a następnie poddawane jest wspomnianemu procesowi osadzeń słów, polegające na zapisaniu słów za pomocą wektorów. W kolejnym etapie dla każdego wektora dodawany jest wektor pozycji, przez co kontekst występowania danego słowa w zdaniu jest lepiej zidentyfikowany. Mechanizm samo uwagi pozwala na „nauczenie się" kontekstu przez model, ponieważ zastosowuje iloczyn skalarny każdej kombinacji wektorów osadzenia słowa, przez co jest w stanie określić zależność kontekstową pomiędzy słowami w danej sentencji. Jednocześnie jest to słabość takiej architektury ponieważ złożoność pamięciowa wynosi N^2 co powoduje kwadratowy przyrost zapotrzebowania zasobów pamięci podręcznej wraz ze wzrostem długości wektora. W ostatnim etapie „Feed forward" przekształca wektor do odpowiedniej postaci kolejnemu enkoderowi (w przypadku modelu BERT) bądź dekoderowi. Następnie dekoder odczytuje dany wektor oraz podaje na wyjściu słowo odpowiadające danemu wektorowi ale w innym języku. Takie podejście gwarantuje tłumaczenie słów wraz z ich kontekstem ponieważ, informacja o kontekście jest zawarta w wielowymiarowym wektorze.

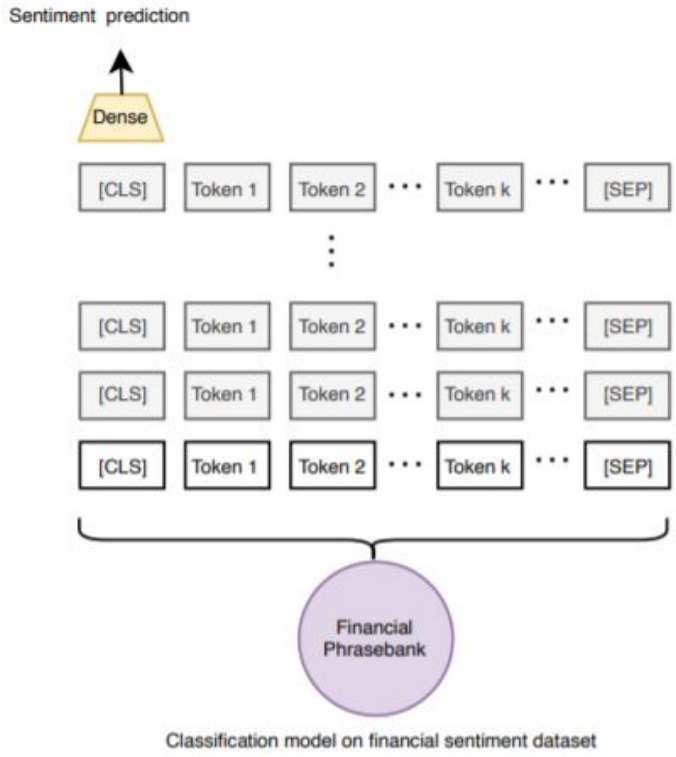
Model BERT to stos enkoderów Transformer, który został najpierw przetrenowany zdaniami z książek oraz Wikipedii w celu poznania ogólnych cech języka, a następnie zbiorem tekstów o tematyce finansowej, tak aby model „rozumiał" specjalistyczne słownictwo charakterystyczne dla branży [6]. Zbiorem danych, który został wykorzystany przez autorów rozszerzenia modelu do wstępnego treningu (ang. *pre-training*) był Reuters TRC2 [6]. Aby proces wstępnego treningu został poprawnie wykonany, przeprowadzona została procedura maskowania (ang. *masked language modelling*), polegająca na usunięciu losowych 15% wyrazów, które model musi na wyjściu odgadnąć, aby osiągnąć umiejętność rozpoznawania kontekstu. Dodatkowo została przeprowadzona predykcja kolejnego zdania (ang. *next sentence prediction*), gdzie model zna-



Rysunek 1: Enkoder oraz dekoder w architekturze Transformer [5]

jąc dwa zdania determinuje czy drugie zdanie występuje po pierwszym [3]. W kolejnym kroku model został poddany dostrojeniu - została dodana ostatnia warstwa klasyfikacyjna w celu możliwości klasyfikacji zdań według sentymentu. Warstwa ta została dodana po ostatnim ukrytym stanie tokena CLS, co zostało pokazane na rysunku 2. Następnie dokonano dostrojenia za pomocą zbioru danych o nazwie „Financial Phrasebank". To zbiór sentencji o składający się z 4500 zdań, zaczerpniętych z wiadomości oraz artykułów o tematyce ekonomii oraz rynków finansowych, w którym każde zdanie posiada swoją etykietę sentymentu. Przygotowany model to właśnie finBERT - taka modyfikacja pozwala na osiągnięcie wyników na najwyższym poziomie zgodnym z aktualnym stanem wiedzy w dziedzinie analizy finansowych treści.

Tabela 1 przedstawia procedurę tokenizacji oraz przydzielania odpowiedniego ID dla danego wyrazu. Sekwencja wejściowa jest reprezentowana przez tokeny reprezentujące słowa oraz tokeny pozycji. Tokeny o nazwie CLS oraz SEP są dodawane odpowiednio na początek oraz koniec każdej sekwencji wyrazów, odpowiednio oznaczone numerami 101 oraz 102 [7]. Następnie tokeny są zamieniane na unikalne identyfikatory (ID). W przypadku gdy model „nie zna" danego słowa, algorytm WordPiece dzieli dane słowo na „podwyrazy", które mogą dostać swój ID, ponieważ takie wyrazy znajdują się w słowniku WordPiece. Bez wykorzystania tego rozwiązania model przydzieliłby token o nazwie UNK, oraz id równym 100. Ponadto długość zdania na wejściu modelu BERT jest ściśle określona. W przypadku podania na wejście modelu zdania o krótszej długości niż wymaga tego od nas model, dodawany jest tzw. „padding" (PAD), który jest pustym tokenem w celu uzupełnienia zdania do wymaganej długości. Limitem długości sekwencji wejściowej jest 512 tokenów, w przypadku większej liczby słów, jednym z rozwiązań jest podzielenie sekwencji wyrazów na podsekcje a następnie podanie na



Rysunek 2: Dostrojenie modelu finBERT [6]

wejście modelu.

Tabela 1: Przykładowy skrypt obrazujący proces tokenizacji oraz przydzielania unikalnego identyfikatora

```
sentence = "Today was a great day to buy amazon stock"
print(sentence)

print(tokenizer.tokenize(sentence))
tokenizer.convert_tokens_to_ids(tokenizer.tokenize(sentence))
tokens_pt = tokenizer(sentence, return_tensors="pt")
for key, value in tokens_pt.items():
    print("{}:\n\t{}".format(key, value))
```

```
Today was a great day to buy amazon stock
['today', 'was', 'a', 'great', 'day', 'to', 'buy', 'amazon', 'stock']
input_ids:
  tensor([[ 101, 2651, 2001, 1037, 2307, 2154, 2000, 4965, 9733, 4518,  ↩
    102]])
token_type_ids:
  tensor([[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]])
attention_mask:
  tensor([[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]])
```

Rezultatem przetworzenia zdania przez model jest trójelementowy wektor logit. W wieloklasowym modelu jest to wejście do funkcji softmax, która służy do normalizacji prawdopodobieństwa, tak aby określić jak bardzo prawdopodobna jest każda klasa dla danej sekwencji wyrazów. Końcowym rezultatem jest trzyelementowy wektor zawierający prawdopodobieństwa dla każdej klasy - sentymentu pozytywnego, neutralnego oraz negatywnego z przedziału od 0 do 1 (widoczne w ostatniej linii w tabeli 2). Wyniki w takiej postaci są proste do zinterpretowania, ponieważ przydzielenie odpowiedniej etykiety sentymentu polega na wybraniu największego prawdopodobieństwa. Dodatkowo aby rezultat mógł zostać określony za pomocą liczb rzeczywistych, odejmujemy od prawdopodobieństwa dla sentymentu pozytywnego prawdopodobieństwo sentymentu negatywnego. Tak przygotowany wskaźnik będzie przyjmował wartości od -1 do 1, gdzie -1 to sentyment skrajnie negatywny, a 1 - pozytywny. Taka reprezentacja oceny sentymentu okaże się być znacznie użyteczniejsza przy badaniu korelacji z ceną.

Tabela 2: Przykładowy skrypt obrazujący proces dodawania na wejście modelu zdania oraz rezultat wykonania skryptu

```
import numpy as np
import torch, math

label_dict = {0: 'positive', 1: 'negative', 2: 'neutral'}
inputs = tokenizer(sentence, return_tensors="pt")

labels = torch.tensor([1]).unsqueeze(0)  # Batch size 1
outputs = model(**inputs, labels=labels)
print(outputs)
print((softmax(outputs.logits.detach().numpy())))
prediction = np.squeeze(np.argmax((softmax(outputs.logits.detach().numpy()))[0]))
print(prediction)
```

```
Today was a great day to buy amazon stock
SequenceClassifierOutput(loss=tensor(4.6432, grad_fn=<NllLossBackward>), logits=tensor([[ 1.7794, -2.6536,  0.2743]], grad_fn=<AddmmBackward>), hidden_states=None, attentions=None)
[[0.81046355 0.00962646 0.17990999]]
```

1.2. Porównanie skuteczności FINBERT do klasycznych metod

Porównanie wyników przetwarzania tekstu zostało wykonane przy pomocy narzędzi Textblob oraz Vader. Textblob[2] jest biblioteką Pythona oferującą proste API zapewniające dostęp do wielu funkcjonalności w tym do klasyfikacji tekstu. Korzysta z metody słownikowej, tzn. każdemu wyrazowi zostaje przyporządkowana wartość „polarity", która mieści się w przedziale od -1 do 1. Uśredniając kolejne wyniki dla każdego z wyrazów otrzymujemy rezultat sentymentu dla danego zdania. Vader (ang. *Valence Aware Dictionary and Sentiment Reasoner*) także jest oparty na metodzie słownikowej, ponadto, jest specjalnie dostrojony do badania sentymentu w mediach społecznościowych [8]. Wynikiem działania programu jest słownik (w nomenklaturze Pythona) z polami „pos", „neu", „neg", przyjmującymi zakres wartości od 0 do 1 oraz

[2] źródło: https://github.com/sloria/TextBlob

„compound", który jest obliczany normalizując poprzednie wyniki do przedziału od -1 do 1. Zaletą takiego podejścia jest brak konieczności posiadania dużej ilości danych potrzebnych do „nauczenia" modelu jak poprawnie rozpoznać sentyment oraz zdecydowanie szybszy czas procesu analizy zdań. Metoda słownikowa radzi sobie gorzej jeśli kontekst zdań nie jest dobrze dostosowany do oryginalnego przeznaczenia słownika. Narzędzie to nie jest oparte na uczeniu maszynowym, przez co inne zastosowanie danego słownika wiązałoby się z koniecznością dodania ręcznie kolejnych reguł, tzn. właściwej oceny intensywności sentymentu dla danego słowa w zależności od kontekstu dziedziny (np. finanse, medycyna, sport) [9].

Aby przeprowadzić badanie porównania skuteczności wymienionych narzędzi do finBERTa, posłużymy się porównaniem macierzy pomyłek, która dobrze sprawdza się przy ocenie skuteczności metod klasyfikacyjnych. Taka macierz zostanie przygotowana na podstawie wyników procesowania zdań dla każdego z modeli a następnie znormalizowana, aby poprawić czytelność rysunku. Danymi wejściowymi jest zbiór przygotowanych tweetów [3], dotyczący spółek będącymi przedmiotem obrotu publicznego na giełdzie NASDAQ oraz NYSE. Zbiór ten zawiera 1363, 2878, 604 tweetów o sentymencie odpowiednio pozytywnym, neutralnym oraz negatywnym. Tweety pochodzą od użytkowników „wpływowych" tzn. powszechnie znanych w środowisku rynków finansowych (np. Jim Cramer, który jest gospodarzem programu „Mad Money" w amerykańskiej stacji CNBC) oraz kont dostarczających informacje (np. Yahoo Finance).

Ponieważ Textblob przydziela tylko wartość liczbową sentymentu, dla celów badawczych etykieta negatywna została przydzielona dla wartości od przedziału -1 do -0.33, neutralna od -0.32 do 0.33 oraz pozytywna od 0.34 do 1. Z rysunku 3a. możemy wywnioskować, że duża część zdań o nacechowaniu pozytywnym dostaje etykietę „neutralny", zdania neutralne są przewidywane w większości poprawnie, chociaż w 27% przypadków jest to sentyment pozytywny. Negatywny sentyment w większości przypadków przydzielany jest zdaniom neutralnym.

Podobna metoda została wykorzystana w przypadku przydzielaniu sentymentu dla narzędzia Vader. Macierz pomyłek przedstawiająca skuteczność klasyfikacji zdań finansowych widoczna na rysunku 3b nie wydaję się wypełniać postawionego zadania wystarczająco dobrze. Należy zauważyć na prawidłowość w której sentyment jest niepoprawnie klasyfikowany jako neutralny, zatem konsekwencje nie powinny zauważalnie zniekształcać wyników. Wnioskując na podstawie rysunku 3c należy zauważyć, że model finBERT zdecydowanie przewyższa pozostałe narzędzia pod względem dokładności klasyfikacji. Najbardziej zauważalna jest różnica w precyzji przydzielaniu etykiet negatywnych (0.8 porównaniu z 0.33 Textbloba oraz 0.48 Vadera). Narzędzia Vader oraz Textblob są bardziej wszechstronne, lepiej poradzą sobie z tekstem o ogólnej tematyce. Nie ulega jednak wątpliwości, że model NLP lepiej radzi sobie z klasyfikacją zdań o danym kontekście, jeśli jest do tego specjalnie przystosowany.

[3] źródło: https://www.kaggle.com/davidwallach/financial-tweets

Rysunek 3: Macierze pomyłek dla narzędzi: (a) Textblob, (b) Vader, (c) finBERT

2. Eksploracja oraz wstępne przetwarzanie danych w projekcie

Twitter jest jednym ze środków masowego przekazu, gdzie różni użytkownicy mogą wyrażać różne opinie oraz poglądy, co czyni go świetnym źródłem informacji potrzebnych do analizy emocji uczestników rynku. Przez ostatnie kilka lat Twitter stał się portalem społecznościowym, z którego można czerpać duże zbiory danych (ang. *big data*), a następnie poddawać analizie. Z powodu rosnącej popularności mediów społecznościowych, duża część użytkowników internetu zamieszcza niezliczone ilości danych. Jeżeli osoba zajmująca się eksploracją danych dysponuje odpowiednimi narzędziami oraz ma sprecyzowany cel odnośnie tego jakich informacji pragnie poznać, takie zbiory danych mogą stać się cennym źródłem pozyskiwania wiedzy na dany temat. Istotną cechą Twittera jest zwięzłość zamieszczanych komunikatów tekstowych, ponieważ nie mogą zawierać więcej niż 280 znaków. Innym kluczowym elementem jest tempo napływania nowych informacji, ponieważ najczęściej to na Twitterze najszybciej pojawiają się nowe wieści oraz wcześniej niż na innych platformach formują się dyskusje na dany temat.

Rysunek 4: Przykładowe wyniki wyszukiwania w zapytanie w wewnętrznej wyszukiwarce portalu Twitter

Ponadto Twitter, dzięki możliwości wyszukiwania informacji po danej frazie oraz wybierania komunikatów tekstowych z danego przedziału czasowego pozwala na precyzyjne znajdowanie danych, gdzie na innych platformach społecznościowych stanowiłoby to większą trudność.

Rysunek 4 przedstawia przykładowe wyniki wyszukiwania zaawansowanego w wewnętrznej wyszukiwarce Twittera, dzięki sprecyzowaniu dokładnych parametrów wyszukiwania, użytkownicy mogą znajdować pożądane dane.

Niewątpliwie jednym z najbardziej czasochłonnych zadań w uczeniu maszynowym jest zgromadzenie oraz przygotowanie danych. Dane pochodzące bezpośrednio ze swojego źródła są bardzo często niepełne, zawierają niepotrzebne informacje przez co nie nadają się jako wejście do modelu. Aby móc uzyskać odpowiednią ocenę sentymentu, zbiory danych wymagają odpowiedniego przygotowania. Tabela 3 przedstawia fragment danych w formacie dataframe [Pandas] z tweetami bezpośrednio po pobraniu z Twittera. Z powodu ograniczeń ilościowych oficjalnego API Twittera, do pobierania tweetów zostało użyte narzędzie Snscrape. Jako kryterium wyszukiwania tweetów została ustawiona fraza odpowiadająca nazwie danego waloru np. „Tesla", pożądany zakres czasowy oraz minimalna liczba polubień pod tweetem, która ma na celu wykluczenie spamu ze zbioru danych.

Po zakończeniu procesu pobierania otrzymujemy plik csv, który w każdym wierszu zawiera treść oraz datę i godzinę zamieszczenia tweeta. Korzystając z biblioteki Pandas w języku Python tworzymy dataframe zawierający wszystkie dane znajdujące się w pobranym pliku. Taka forma danych ułatwi proces przygotowania oraz umożliwi wykonanie dalszej analizy sentymentu.

Tabela 3: Przykładowe dane wejściowe

11

	DATE	TWEET
0	28/10/2021, 23:57:33	b'My next car is definitely a Tesla I want plug and go i dont want to get gas anymore'
6	28/10/2021, 23:53:08	b"@WR4NYGov Glad you're a Tesla fan for several reasons! As Tesla's save lives. Life is strange as you never know what your interests will change in your life."
7	28/10/2021, 23:52:13	b'Tesla market cap will be bigger than Apple in less than 5 years or even 4.'
11	28/10/2021, 23:48:21	b'Tesla is running a single stack of FSD Beta in the United States and Canada. \n\nIt just works. And if works incredibly. https://t.co/clxOXTaErB'
47	28/10/2021, 23:18:38	b'Today was all about Tesla options and DOGE soaring back! '
45	28/10/2021, 23:19:52	b'@StanphylCap Tesla is up $8B in AH on stock split hopes.\nWhat a time to be alive.'

Tabela 3 przedstawia przykładowe dane zaimportowane z pliku w formacie csv przed pierwszą obróbką. Z poniższego zbioru danych możemy zauważyć charakterystyczne cechy formatu komunikatu tekstowego. Każdy tweet pobierany jest w postaci kodu bajtowego (sekwencji bajtów), lecz po zaimportowaniu pliku do postaci dataframe tekst przybiera formę string. Aby doprowadzić każde zdanie do postaci widocznej w tabeli 6 dokonano implementacji widocznej w tabelach 4 oraz 5. Wykorzystano wyrażenia regularne w celu eliminacji niepotrzebnych znaków (reprezentacja kodu bajtowego, znaki nowych linii, niepotrzebne cudzysłowy oraz znaki hash) oraz bibliotekę Tweet-preprocessor w celu eliminacji znaczników odpowiedzi do użytkownika (np. @kuba123) oraz linków [10].

Tabela 4: Usuwanie niepotrzebnych symbolów z wykorzystaniem wyrażeń regularnych

```
tweets_list = []
for index, row in df_tweets.iterrows():
     tweets_list.append([row.DATE, row.TWEET])

df_tweets = df_tweets.replace(to_replace ="^b'", value = "", regex = True)
df_tweets = df_tweets.replace(to_replace ='^b"', value = "", regex = True)
df_tweets = df_tweets.replace(to_replace ="'\s*$", value = "", regex = ↩
    True)
df_tweets = df_tweets.replace(to_replace ='"\s*$', value = "", regex = ↩
    True)
df_tweets = df_tweets.replace(to_replace ='\\n', value = "", regex = True)
df_tweets = df_tweets.replace(to_replace ='\#', value = "", regex = True)
df_tweets
```

Tabela 5: Usuwanie linków oraz znaczników użytkowników z wykorzystaniem Tweet-preprocessor

```
import preprocessor as p

tweets_list = []
for index, row in df_tweets.iterrows():
    tweets_list.append([row['DATE'], row['TWEET']])
# # # Print the updated dataframe
tweets_list_copy = tweets_list.copy()
p.set_options(p.OPT.URL, p.OPT.MENTION)
for tweet, tw in zip(tweets_list_copy, tweets_list):
    tw[-1] = p.clean(tweet[-1])
    tw[-1] = tweet[-1].replace("\\n", "")
```

Rezultatem wykonania powyższych skryptów jest dataframe widoczny w tabeli 6.

Tabela 6: Tweety po procesie przygotowania

	TWEET	DATE	TIME
0	My next car is definitely a Tesla I want plug and go i dont want to get gas anymore	28/10/2021	23:57:33
1	Glad you're a Tesla fan for several reasons! As Tesla's save lives. Life is strange as you never know what your interests will change in your life.	28/10/2021	23:53:08
2	Tesla market cap will be bigger than Apple in less than 5 years or even 4.	28/10/2021	23:52:13
3	Tesla is running a single stack of FSD Beta in the United States and Canada. It just works. And if works incredibly.	28/10/2021	23:48:21
4	Today was all about Tesla options and DOGE soaring back!	28/10/2021	23:18:38
5	Tesla is up $8B in AH on stock split hopes.What a time to be alive.	28/10/2021	23:19:52

Dopiero po procesie eksploracji oraz przygotowania, zdania w kolumnie „TWEET” w tabeli 6. przypominają komunikaty tekstowe widoczne na platformie Twitter. Ponadto kolumna „DATE” widoczna w tabeli 3. została podzielona na dwie części zawierające osobno datę (zmieniono również format zapisu daty, aby poprawić czytelność) oraz czas wysłania tweeta. Taki format danych umożliwi grupowanie tweetów w późniejszych etapach pracy. Tak przygotowane dane są gotowe do przetworzenia przez finBERT, który oceni każdą sentencje nadając prawdopodobieństwo wystąpienia danego sentymentu, oraz nada odpowiednie etykiety, które odzwierciedlają sentyment zdania.

3. Aplikacja korelująca sentyment tekstu z ceną waloru

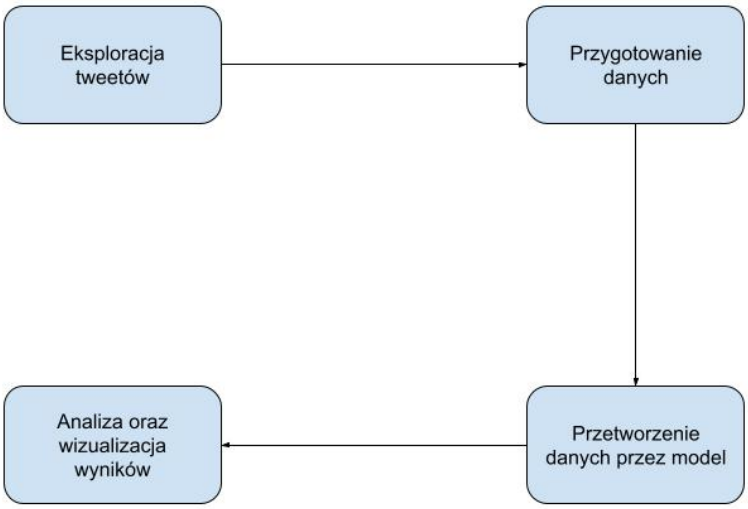
Aby sprawdzić jak zachowują się względem siebie cena oraz nastrój uczestników jaki panuje na rynku, została zaimplementowana aplikacja mająca na celu zbadanie wymienionych zależności. Sentyment na rynkach finansowych jest nastawieniem uczestników rynku względem danych instrumentów finansowych tj. akcji, obligacji czy kryptowalut. Ponieważ jest to odzwierciedlenie emocji inwestorów względem zmienności ceny, zadaniem, z którym najczęściej mierzą się inwestorzy oraz analitycy jest przewidywanie jak rynki zachowają się po upublicznieniu danej informacji [11]. Zazwyczaj, jeśli cena waloru rośnie, poziom sentymentu zachowuje się w podobny sposób, jeśli cena spada, sentyment również się pogarsza. Jednak relacja ta nie wydaje się być tak prosta, ponieważ zdarzają się sytuacje, w których cena oraz sentyment rosną lub spadają nierównomiernie tzn. sentyment rośnie lub spada bardziej dynamicznie niż cena. Takie podejście jest zaprzeczeniem popularnej, w latach 60. XX wieku hipotezy rynków efektywnych (ang. *Efficient Market Hyphotesis*), której podstawą jest twierdzenie, iż wszystkie dostępne informacje nt. papieru wartościowego mają odzwierciedlenie w jego cenie [12]. Z czasem ta teoria stała się coraz bardziej krytykowana, przez inwestorów (Warren Buffett, George Soros) oraz ekonomistów behawioralnych (m.in. Daniel Kahnemann, Richard Thaler), którzy twierdzą, że niedoskonałość rynków finansowych wynika głównie z błędów poznawczych takich jak: przesadnej pewności siebie, zbyt silnych reakcji na napływające informacje czy też efektem potwierdzenia [13]. Czynniki te mogą być podstawą do analizy walorów na rynku finansowym przez potencjalnych inwestorów, aby wyjaśnić przyczynę nienaturalnego zachowania uczestników rynku i dostrzec potencjalną okazję inwestycyjną.

Taka zależność była inspiracją do napisania aplikacji, która miała na celu zestawić sentyment uczestników rynku, tj. użytkowników tweetujących na temat danego waloru oraz cenę. Dysponując takimi narzędziami jak Snscrape[4], możliwe jest zagregowanie dużych zbiorów danych a następnie poddanie ich ocenie modelowi NLP. Wynikiem takiego działania jest zbiór tweetów, w którym każdy tweet ma swoją ocenę sentymentu. W tej aplikacji porównanie sentymentu z ceną będzie dokonywane na wykresie dziennym, tzn. dla każdego dnia z określonego przedziału czasowego będzie narysowany punkt odpowiadający cenie oraz inny punkt odpowiadający sentymentowi. Punkt ten będzie średnią wszystkich wyników liczbowych oceny sentymentu z danego dnia. Istnieje wiele potencjalnych przeznaczeń tak wykonanej aplikacji. Jednym z zastosowań może być ocena długoterminowego trendu panującego na danym walorze. Wiele strategii inwestycyjnych podejmowanych przez fundusze inwestycyjne czy też indywidualnych inwestorów polega na identyfikacji trendu a następnie „grę z trendem". Innym przykładem może być podejście stosowane przez fundusze ilościowe, które często używają agregacji wskaźników technicznych, sentymentu oraz zmienności w celu opracowania modelu tradingowego. W tej pracy pole analizy zostanie ograniczone do zbadania jak sentyment koreluje się z ceną oraz jakie wartościowe wnioski można wyciągnąć w celu budowy strategii opartej na inwestowaniu wyłącznie przy wykorzystaniu analizy sentymentu. Jednym z obszarów takiej analizy jest sprawdzenie współczynnika korelacji Pearsona pomiędzy wymienionymi dwoma zmiennymi, czyli określenie miary jak bardzo para zmiennych jest ze sobą współzależna [14].

[4] https://github.com/JustAnotherArchivist/snscrape

3.1. Struktura projektu

Projekt został podzielony na cztery części. Na pierwszą część składa się pobranie danych, następnie inny skrypt odpowiada za właściwe przygotowanie danych. Kolejnym krokiem jest podanie danych na wejście modelu, a następnie przetworzenie tweetów. Ostatnim etapem projektu jest wizualizacja wyników oraz analiza osiągniętych rezultatów. Aplikacja składa się z czterech skryptów, przepływ informacji pomiędzy skryptami przedstawiono na rysunku 5.



Rysunek 5: Diagram struktury projektu

3.2. Implementacja

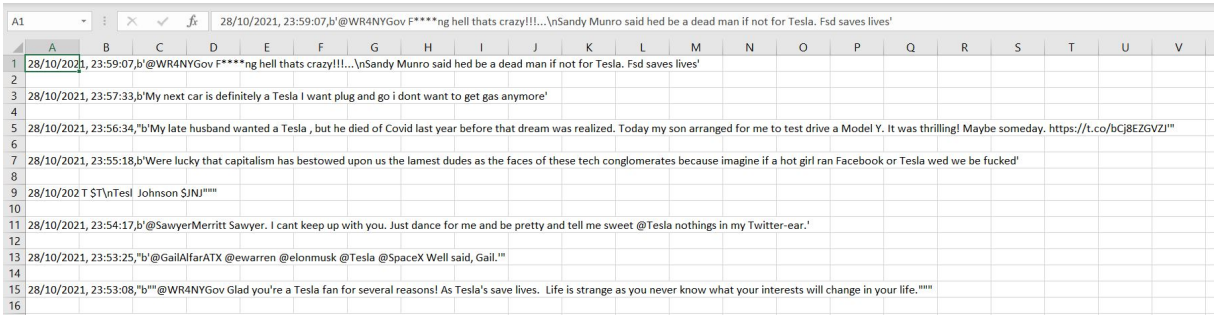
Projekt został zaimplementowany w języku Python, ponieważ język ten dostarcza wiele bibliotek umożliwiających operowanie narzędziami w dziedzinie nauki o danych (*data science*). Ponadto, jego powszechnie znana łatwość użycia oraz prosta składnia sprawiają, że jest to jeden z najbardziej popularnych języków w gałęzi eksploracji danych oraz uczenia maszynowego. Do pobranie tweetów wykorzystane została biblioteka Snscrape. Jest to narzędzie służące do eksploracji danych z mediów społecznościowych (np. Twitter, Facebook, Reddit), umożliwiające ściąganie postów, profilów użytkowników, liczba polubień oraz komunikatów tekstowych według kryterium wyszukiwania. Planowane było użycie oficjalnego API Twittera, niestety okazało się być niemożliwe pobieranie tweetów z przedziału czasowego dłuższego niż tydzień wstecz, a ponadto czasowy limit ilości żądań pobrania tweetów (100 tweetów na żądanie GET wysyłane co 5 sekund) nie pozwoliłby na efektywne pobranie setek tysięcy tweetów. Takie limity nie występują w Snscrape, co pozwoliło na efektywne pobranie dużych zbiorów danych.

Tabela 7: Skrypt pobierający dane z tweetera

```
import snscrape.modules.twitter as sntwitter
import csv
import unicodedata
import datetime

# Creating list to append tweet data to
tweets_list = []
query = "Tesla"
min_faves = 5
since_date = "2021-08-01"
until_date = "2021-10-31"
filename = "{}_snscrape_{}_{}_favs{}".format(query, since_date, until_date, min_faves)
csvFile = open(filename + '.csv', 'w')

csvWriter = csv.writer(csvFile)

# Using TwitterSearchScraper to scrape data and append tweets to list

for i, tweet in enumerate(sntwitter.TwitterSearchScraper('{} min_faves:{} lang:en since:{} until:{}'.format(query, min_faves, since_date, until_date)).get_items()):
    created_at = datetime.datetime.strptime(str(tweet.date), "%Y-%m-%d %H:%M:%S+%f:00").strftime('%d/%m/%Y, %H:%M:%S')
    tw = unicodedata.normalize('NFKD', tweet.content).encode('ascii', 'ignore')
    csvWriter.writerow([created_at, tw])
    print(i, created_at, tw)
```

Skrypt widoczny w tabeli 7 przedstawia kod służący do eksploracji danych z portalu Twitter, wraz z wprowadzonymi kryteriami: wyszukiwana fraza, minimalna liczba polubień oraz przedział czasowy.

Po średnio trwającym kilkadziesiąt minut procesie pobierania, otrzymujemy plik w formacie csv. Przykładowa treść takiego pliku jest widoczna na rysunku 6.



Rysunek 6: Komunikaty tekstowe w formacie csv, bezpośrednio po ukończeniu procesu pobierania

Tabela 8: Importowanie oraz wczytywanie modelu za pomocą biblioteki Transformers

```
from transformers import AutoTokenizer, AutoModelForSequenceClassification

tokenizer = AutoTokenizer.from_pretrained("ProsusAI/finbert")

model = AutoModelForSequenceClassification.from_pretrained("ProsusAI/↩
    finbert")
```

Tak otrzymane dane należy teraz wczytać do formatu danych dataframe. Umożliwia to biblioteka Pandas, służąca głównie do przechowywania oraz reprezentacji danych. W następnym kroku tweety obsługiwał będzie skrypt odpowiedzialny za wstępne przygotowanie danych. Kolejnym krokiem jest przetworzenie danych przez model finBERT. Model został szczegółowo opisany w rozdziale 1, natomiast w tej części zostanie opisana jego implementacja. Model został pobrany ze strony HuggingFace[5], która jest otwartym źródłem zawierającym modele, zbiory danych oraz inne zasoby uczenia maszynowego. Implementacja kodu została przeprowadzona z wykorzystaniem biblioteki Transformers, która dostarcza API do prostego pobrania i wykorzystania dostępnych modeli oraz jest wspierana przez popularne biblioteki uczenia głębokiego takie jak Jax, Pytorch, Tensorflow. Po zainstalowaniu biblioteki Transformers, model oraz tokenizer został zapisany do zmiennych widocznych w tabeli 8.

Następnie została zaimplementowana funkcja softmax, często używana jako ostateczna funkcja aktywacji sieci neuronowej w celu znormalizowania wyników do postaci rozkładu prawdopodobieństwa przewidywanych klas. Wynikiem wykonania takiej funkcji jest prawdopodobieństwo przypisane do danej klasy z przedziału od 0 do 1. Przykładowa implementacja funkcji została przedstawiona w tabeli 9. Aby funkcja działała poprawnie, należy zaimportować bibliotekę NumPy, która służy do obróbki danych numerycznych, ponadto posiada własny typ danych, który jest stosowany m.in. do efektywnego przetwarzania wielowymiarowych tabel oraz macierzy.

Tabela 9: Implementacja funkcji softmax

```
import numpy as np
def softmax(z): return np.exp(z)/((np.exp(z)).sum())
```

Kolejny etap to przetworzenie sentencji przez model oraz odpowiednie zapisanie danych wyjściowych do użytecznego formatu danych. W tym celu została napisana funkcja get_output, widocznej w tabeli 10, która przyjmuję ramkę danych oraz argumenty opcjonalne, które mogą wydrukować przetworzone zdania wraz z wynikowymi klasą oraz zwracająca gotowy dataframe, wraz z ocenami sentymentu dla każdego zdania oraz odpowiednią etykietą.

[5] źródło: https://huggingface.co/ProsusAI/finbert

Tabela 10: Implementacja funkcji get_output

```
import torch, math

def get_output(df, print_sentences = True, print_output = True):
  df_sentences = df["TWEET"]
  positive = []
  negative = []
  neutral = []
  predictions = []
  sentiment_score = []
  label_dict = {0: 'positive', 1: 'negative', 2: 'neutral'}

  for sentence in df_sentences:
    if print_sentences:
      print(sentence)
    inputs = tokenizer(sentence, return_tensors="pt")
    labels = torch.tensor([1]).unsqueeze(0)  # Batch size 1
    outputs = model(**inputs, labels=labels)

    if print_output:
      print(outputs)
      print((softmax(outputs.logits.detach().numpy())))

    positive.append((softmax(outputs.logits.detach().numpy()))[0][0])
    negative.append((softmax(outputs.logits.detach().numpy()))[0][1])
    neutral.append((softmax(outputs.logits.detach().numpy()))[0][2])

    prediction = np.squeeze(np.argmax((softmax(outputs.logits.detach().↩
   numpy()))[0]))
    predictions.append(label_dict[prediction])

  df['positive'] = positive
  df['negative'] = negative
  df['neutral'] = neutral

  for pos, neg in zip(positive, negative):
    sentiment_score.append(pos-neg)
  df['sentiment_score'] = sentiment_score
  df['predictions'] = predictions

  return df
```

Wstępnie przygotowana ramka danych jest argumentem funkcji get_output, gdzie odbywa się przetwarzanie sentencji, w celu uzyskania ocen sentymentu. Kilka przykładowych obiektów na wyjściu modelu zostało przedstawione w tabeli 11. Przetworzenie bloku danych o pojemności 10 tysięcy tweetów trwało około 10 minut z wykorzystaniem laptopa z podzespołami takimi jak 6 rdzeniowy procesor Intel Core i7 oraz 16 GB pamięci RAM. Rezultatem przetwarzania sentencji przez model finBERT jest klasa SequenceClassifierOutput. Jest to bazowa klasa dla wyjścia modeli klasyfikacyjnych. Składa się ona z takich pól jak loss, grad_fn, logits, hid-

Tabela 11: Przykładowe wyniki bezpośrednio na wyjściu modelu finBERT

```
Soon it will seem like all rotation will be from other stocks to Tesla!
SequenceClassifierOutput(loss=tensor(4.1060, grad_fn=<NllLossBackward>), ↩
   logits=tensor([[-0.5063, -1.6223,  2.4146]], grad_fn=<AddmmBackward>),↩
    hidden_states=None, attentions=None)
[[0.0502894  0.01647428 0.9332363 ]]
Screw Elon Musk, the selfish bastard. This seals the deal for me - I'll ↩
   never own a Tesla even if I could.
SequenceClassifierOutput(loss=tensor(4.1650, grad_fn=<NllLossBackward>), ↩
   logits=tensor([[-0.1610, -1.8843,  2.1726]], grad_fn=<AddmmBackward>),↩
    hidden_states=None, attentions=None)
[[0.08700471 0.01552959 0.8974657 ]]
Detroit automakers would be able to offer $4,500 more in tax credits to U.↩
   S. electric car buyers than rivals such as Tesla and Toyota under ↩
   Biden's new proposal
SequenceClassifierOutput(loss=tensor(4.7405, grad_fn=<NllLossBackward>), ↩
   logits=tensor([[ 1.9029, -2.6954, -0.0440]], grad_fn=<AddmmBackward>),↩
    hidden_states=None, attentions=None)
[[0.8674605  0.00873468 0.1238049 ]]
A lot of people and 401ks made a lot of money that day too from owning ↩
   Tesla stock.
SequenceClassifierOutput(loss=tensor(3.9846, grad_fn=<NllLossBackward>), ↩
   logits=tensor([[-0.2679, -1.7037,  2.1791]], grad_fn=<AddmmBackward>),↩
    hidden_states=None, attentions=None)
[[0.07817888 0.01860072 0.90322036]]
```

den_states czy attentions. Najbardziej interesującym dla nas polem jest tensor logits, który zawiera prawdopodobieństwa wystąpienia danej klasy sentymentu (pozytywny, negatywny, neutralny) w przedziale od minus nieskończoności do plus nieskończoności. Aby prawdopodobieństwo zostało wyrażone w przedziale od 0 do 1, została użyta funkcja softmax, która konwertuje przedział do pożądanych wielkości.

Aby przeprowadzić analizę danych, należy zapisać dane w postaci ramki danych (dataframe) przy pomocy biblioteki Pandas, tak aby przetwarzanie danych było możliwie jak najmniej czasochłonne. Ramka danych zawierająca sentencje z gotowymi wynikami została przedstawiona w tabeli 12.

Tabela 12: Ramka danych po przetworzeniu przez model

13

	TWEET	DATE	TIME	positive	negative	neutral	sentiment_score	predictions
0	Soon it will seem like all rotation will be fr...	28/10/2021	21:28:23	0.050289	0.016474	0.933236	0.033815	neutral
1	Screw Elon Musk, the selfish bastard. This sea...	28/10/2021	21:27:17	0.087005	0.015530	0.897466	0.071475	neutral
2	Detroit automakers would be able to offer $4,5...	28/10/2021	21:27:03	0.867460	0.008735	0.123805	0.858726	positive
3	A lot of people and 401ks made a lot of money ...	28/10/2021	18:18:40	0.078179	0.018601	0.903220	0.059578	neutral

Ramka widoczna w tabeli 12 zawiera wszystkie potrzebne dane do sprawdzenia korelacji pomiędzy ceną a sentymentem panującym wśród użytkowników Twittera. Z tak przygotowanych danych obliczamy średnią arytmetyczną oceny sentymentu dla każdego zdania („senti-

ment_score” w tabeli 12) opublikowanego danego dnia, a następnie zestawiamy go z ceną na jednym wykresie.

Ostatnim etapem projektu jest wizualizacja otrzymanych wyników. Aby móc wyciągnąć użyteczne wnioski należy nanieść wszystkie dane na wykres. W celu osiągnięcia zamierzonych celów została wykorzystana inna biblioteka Pythona - Matplotlib, która służy do rysowania wykresów, a także jest rozszerzeniem biblioteki NumPy. Wyniki zostaną szczegółowo opisane w rozdziale 4.

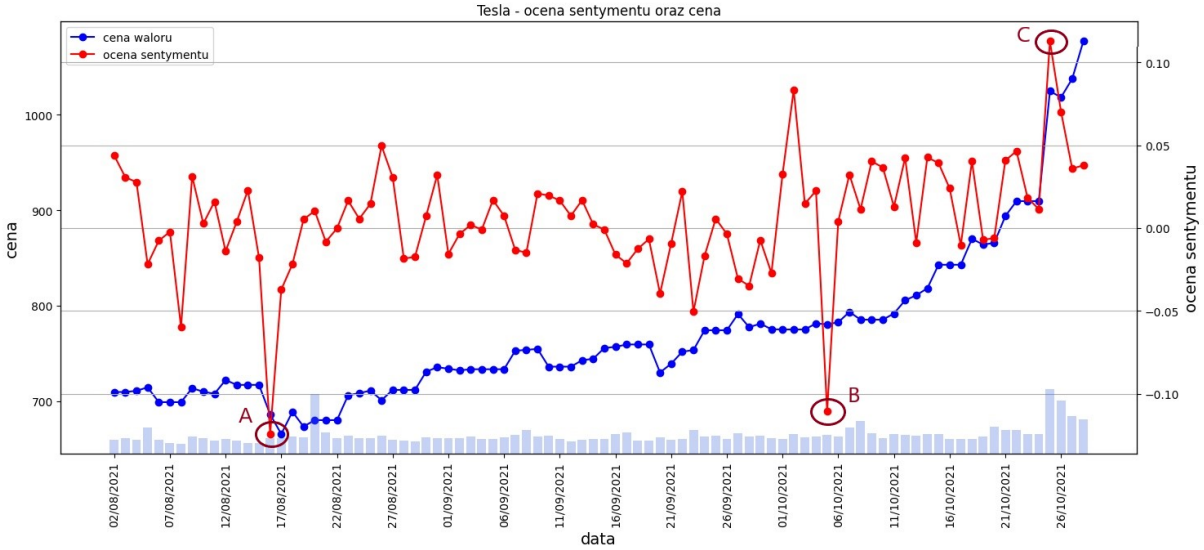
4. Analiza otrzymanych wyników

Na tym etapie projektu zostanie przeprowadzona analiza otrzymanych wyników poprzez naniesienie rezultatów na wykres oraz porównanie ich z ceną waloru. Dla każdego waloru zostaną przedstawione dwa wykresy porównujące ocenę sentymentu wraz z ceną w odmiennych formach, pierwszy przedstawiający dzienną ocenę sentymentu oraz drugi uśredniający ocenę sentymentu za pomocą dwóch średnich kroczących: pięciodniowej oraz dziesięciodniowej. Takie średnie kroczące zostały wybrane ze względu na najwyższe uzyskane współczynniki korelacji z ceną. Cena dla każdego omówionego poniżej przypadku jest wyrażona w dolarze amerykańskim. W każdej poniższej podsekcji zostały wymienione czynniki takie jak liczba tweetów, przedział czasowy oraz współczynnik korelacji Pearsona między ceną a oceną sentymentu. W analizie zostały wykorzystane spółki takie jak Tesla, Apple, Facebook oraz Gamestop, a także kryptowaluta Bitcoin. Każdy z walorów zostanie osobno opisany w kolejnych sekcjach.

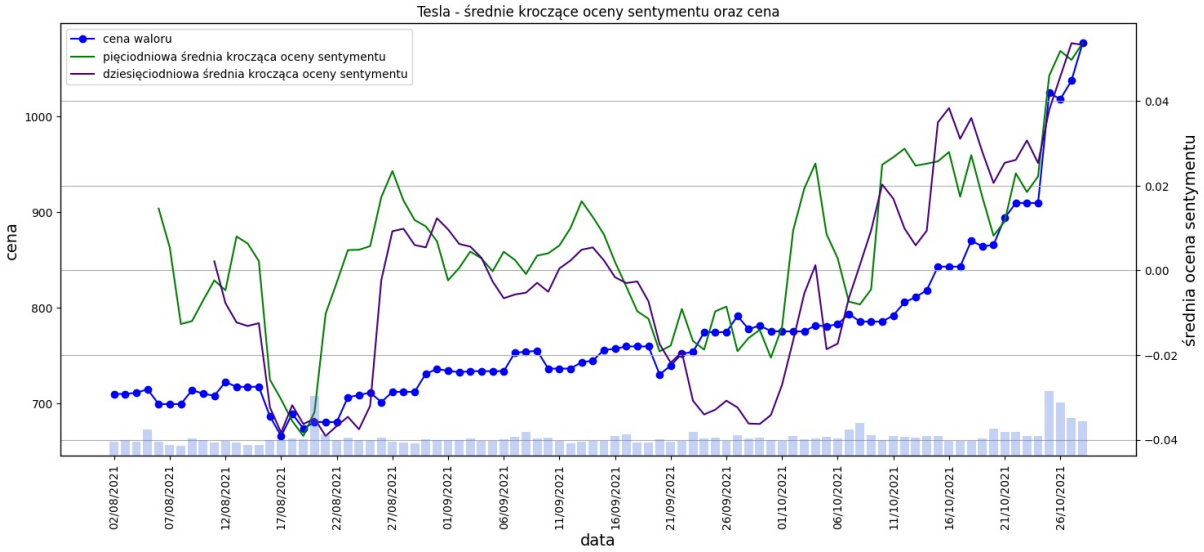
Tesla

Tesla jest to spółka produkująca samochody elektryczne, notowana na giełdzie NASDAQ od 2010 roku. Większościowym udziałowcem Tesli jest Elon Musk, najbogatszy człowiek na świecie (stan: 12.12.2021). Musk jest osobą medialną, aktywną na portalu Twitter, co przejawia się dużym zainteresowaniem zarówno jego osobą jak i jego działalnościami biznesowymi. Konsekwencją tego jest duża liczba tweetów na temat samego Muska oraz Tesli, co powoduje, że spółka ta jest dobrym walorem do analizy sentymentu. Do wykonania eksperymentu zostało wykorzystane ponad 110 tys. tweetów wysłanych od wszystkich użytkowników, z kryterium minimalnej liczby polubień 5 (takie kryterium obowiązywać będzie również dla Apple oraz Facebook). Przedziałem czasowym w poniższej analizie jest zakres od 01.08.2021 do 30.10.2021. Na wykresie widać 3 charakterystyczne punkty. Pierwszy miał miejsce 16 sierpnia (punkt A na rysunku 7), gdy sentyment na wykresie osiągnął minimum. Wydarzeniem w tle było wszczęcie śledztwa po zderzeniu Tesli prowadzonej przez autopilota z pojazdem uprzywilejowanym. Kolejnym dołkiem widocznym na wykresie (punkt B) była informacja o karze 137 milionów dolarów, którą spółka musi zapłacić za złe traktowanie afroamerykańskich pracowników.

Rysunek 7 zestawiający ze sobą cenę akcji oraz ocenę sentymentu, na pierwszy rzut oka nie przedstawia żadnej zależności pomiędzy wymienionymi zmiennymi. Na dole wykresu umieszczony został wolumen liczby tweetów przetworzonych danego dnia - tak narysowany wolumen będzie widoczny na każdym kolejnym wykresie w tym rozdziale. Maksimum wolumenu tweetów zostało osiągnięte 25 października, a tego samego dnia ocena sentymentu osiągneła swój szczyt na wykresie (punkt C) kiedy to pojawiła się informacja, że spółka Hertz kupi 100 000 elektrycznych samochodów Tesli, jednocześnie powodując wzrostu kursu spółki do poziomu kapitalizacji równemu biliona dolarów. Dzienne minimum to 673 tweety, a średnia dla każdego dnia wynosi 1251 komunikatów tekstowych. Współczynnik korelacji wynosi 0.43, zatem jest to korelacja przeciętna. W kolejnym kroku, dla każdego z analizowanych walorów zostaną wyciągnięte dwie średnie kroczące - pięcio oraz dziesięciodniowa.



Rysunek 7: Dzienna ocena sentymentu oraz cena spółki Tesla



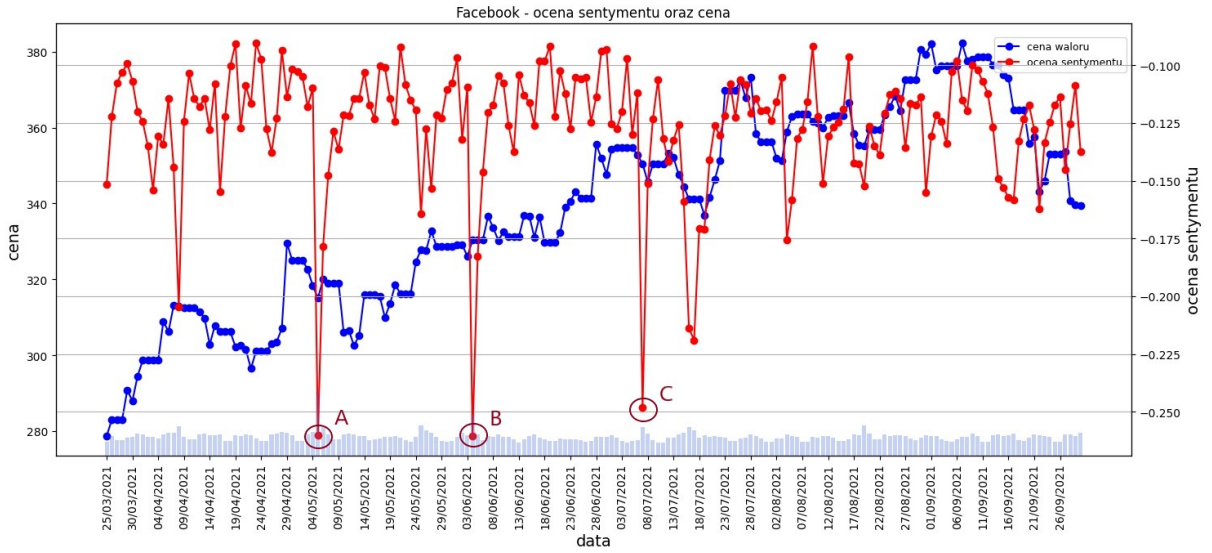
Rysunek 8: Pięciodniowa i dziesięciodniowa średnia krocząca oceny sentymentu oraz cena spółki Tesla

Na rysunku 8 obydwie średnie ruchome, wydają się mocno korelować cenę Tesli. Współczynnik korelacji Pearsona dla średnich pięciodniowej oraz dziesięciodniowej wynosi odpowiednio 0.69 oraz 0.74, zatem jest to korelacja wysoka oraz bardzo wysoka.

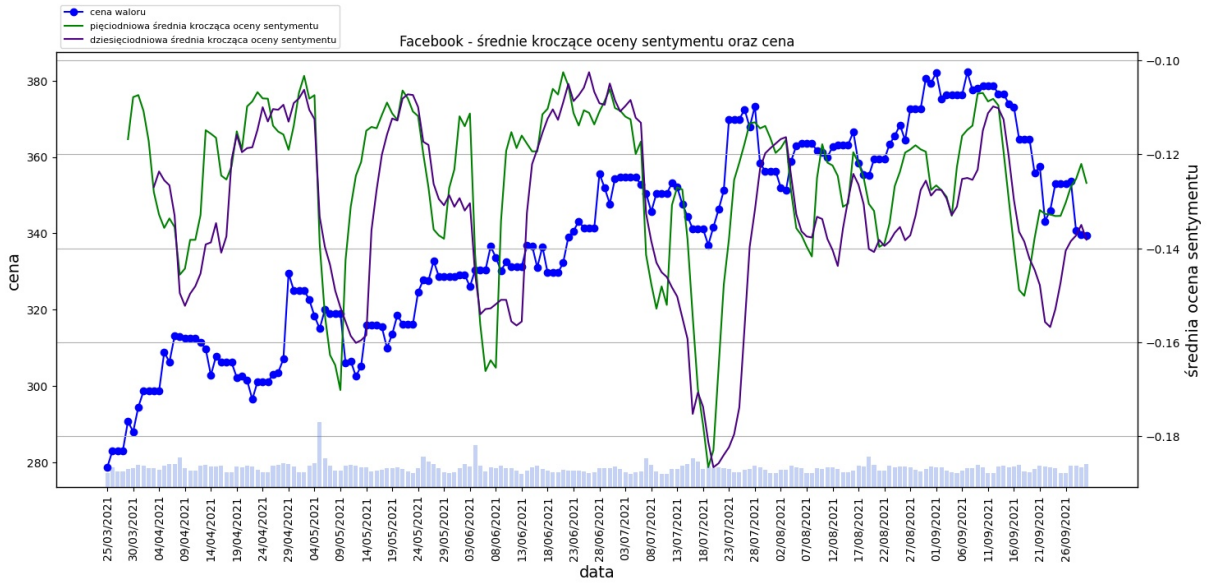
Facebook

Facebook jest potentatem w branży mediów społecznościowych zarządzanym przez spółkę Meta Platforms, do niedawna znaną także pod tą samą nazwą - Facebook. Spółka ta zadebiutowała na giełdzie NASDAQ 12 marca 2012 roku, gdzie od tamtej pory stała się jedną z największych firm Big Tech na świecie z kapitalizacją rynkową o wartości bliską jednego biliona dolarów, posiadająca bazę użytkowników liczącą około trzech miliardów. Analizowanym

okresem był przedział czasu, począwszy od 25 marca 2021 do 31 października 2021 roku, w którym zebrano około 332 tys. tweetów. Średnio na każdy dzień przypada około 1750 tweetów, gdzie minimum to 1169, a maksimum 5968.



Rysunek 9: Dzienna ocena sentymentu oraz cena spółki Facebook



Rysunek 10: Pięciodniowa i dziesięciodniowa średnia krocząca oceny sentymentu oraz cena spółki Facebook

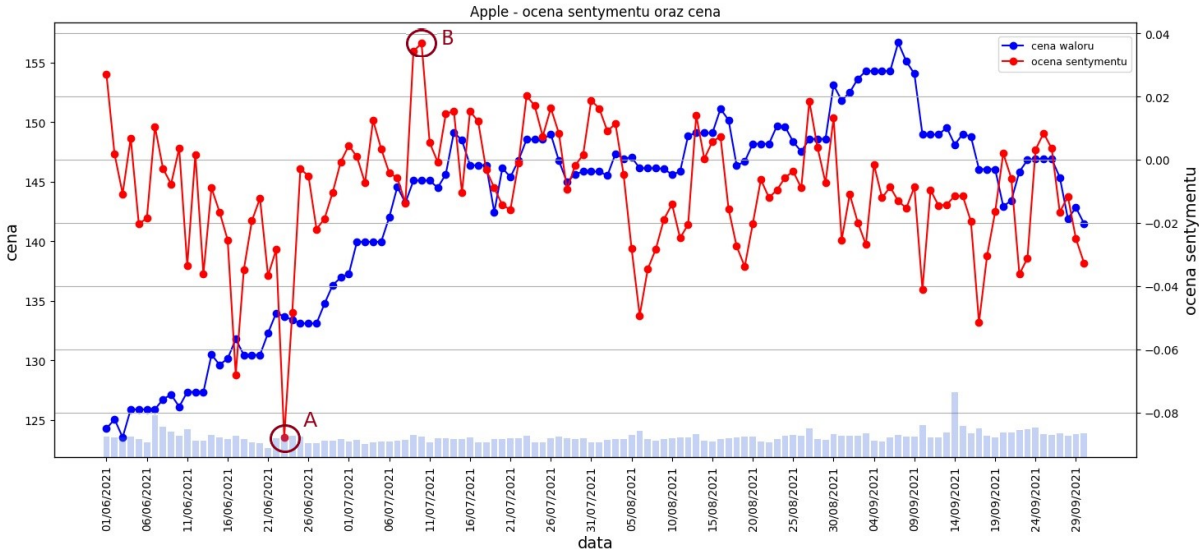
Rysunek 9 przedstawia zestawioną ocenę sentymentu oraz cenę akcji. Powyższy wykres nie wykazuje żadnej korelacji pomiędzy ceną a oceną sentymentu, a współczynnik korelacji wynosi -0.04. Należy zwrócić uwagę na fakt, iż w całym analizowanym okresie ocena sentymentu nie przekracza -0.05. Ponadto na rysunku widoczne są trzy wyraźne spadki do poziomów poniżej -0.25. Co ciekawe w każdym z przypadków, przyczyną tak negatywnego sentymentu były informacje dotyczące Donalda Trumpa. 5 maja (punkt A), kiedy to pojawiła się informacja o przedłużeniu zakazu publikowania treści dla byłego prezydenta Stanów Zjednoczonych, 4

czerwca (punkt B), gdy Facebook poinformował o przedłużenie zakazu dla Trumpa do stycznia 2023 roku oraz 7 lipca (punkt C) gdy ten sam Trump poinformował, że pozywa firmy Facebook, Twitter oraz Google, oskarżając je o cenzurę.

Na rysunku 10 widoczna jest wyraźna zmienność sentymentu, jednak trudno znaleźć odzwierciedlenie owego zjawiska w cenie. Zarówno dla średniej pięciodniowej jak i dziesięciodniowej korelacja po zaokrągleniu do drugiego miejsca po przecinku wynosi -0.07.

Apple

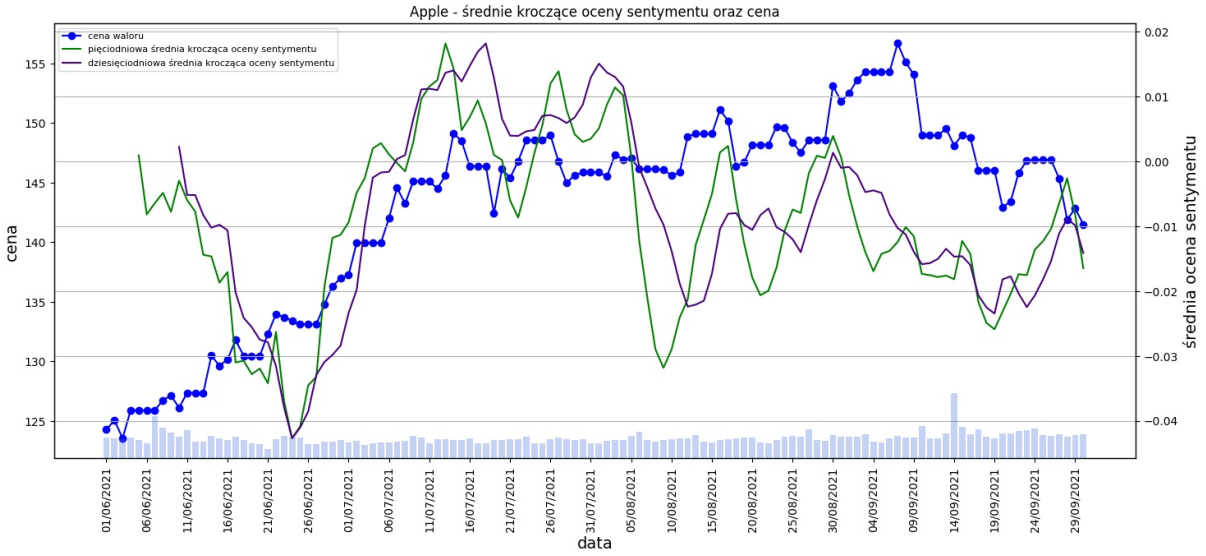
Kolejną spółką która została poddana analizie jest Apple, czyli dobrze znane przedsiębiorstwo słynące z produkcji sprzętu komputerowego oraz mobilnego. Tak jak poprzednie spółki - firma notowana jest na giełdzie NASDAQ od 1980 roku. Dziś spółka jest największą firmą na świecie z kapitalizacją rynkową bliską 3 bilionów dolarów. Analizie sentymentu panującego wokół spółki zostało poddane 420 tysięcy tweetów, opublikowanych od 1 czerwca do 30 września 2021 roku. Dzienna średnia tweetów dla analizowanego okresu wynosi 3488, gdzie 11220 to maksimum (tego samego dnia odbyła się premiera nowego iPada) a 1510 to dzienne minimum.



Rysunek 11: Dzienna ocena sentymentu oraz cena spółki Apple

Współczynnik korelacji między zbiorem ocen sentymentów a ceną wynosi 0.28 co wskazuje na słabą korelację. Analizując rysunek 11 należy zwrócić uwagę na dwie skrajne wartości pojawiające się najpierw 23 czerwca (punkt A), kiedy pojawiła się informacja o zamknięciu gazety z siedzibą w Hong Kongu o tej samej nazwie co analizowana spółka (wydarzenie to zostało uznane przez wiele mediów za koniec epoki wolnych mediów w Hong Kongu) oraz 10 lipca (punkt B), w tym przypadku brak jednoznacznej przyczyny wzrostu sentymentu. Pierwsze wydarzenie sugeruje, że wyszukiwanie tweetów po frazie „Apple" jest niedoskonałe, ponieważ nie wszystkie powiązania frazy związane są z firmą produkującą sprzęt oraz oprogramowanie. W analizowanym zbiorze danych takie frazy występują także w kontekście przepisów kulinarnych bądź innych nazw własnych. Rysunek 12 przedstawia średnie kroczące na tle ceny spółki

w trendzie wzrostowym. Korelacja dla pięciodniowej średniej kroczącej wynosi 0.28, zaś dla dziesięciodniowej 0.37. Są to współczynniki opisywane w skali statystycznej odpowiednio jako korelacja słaba oraz przeciętna.



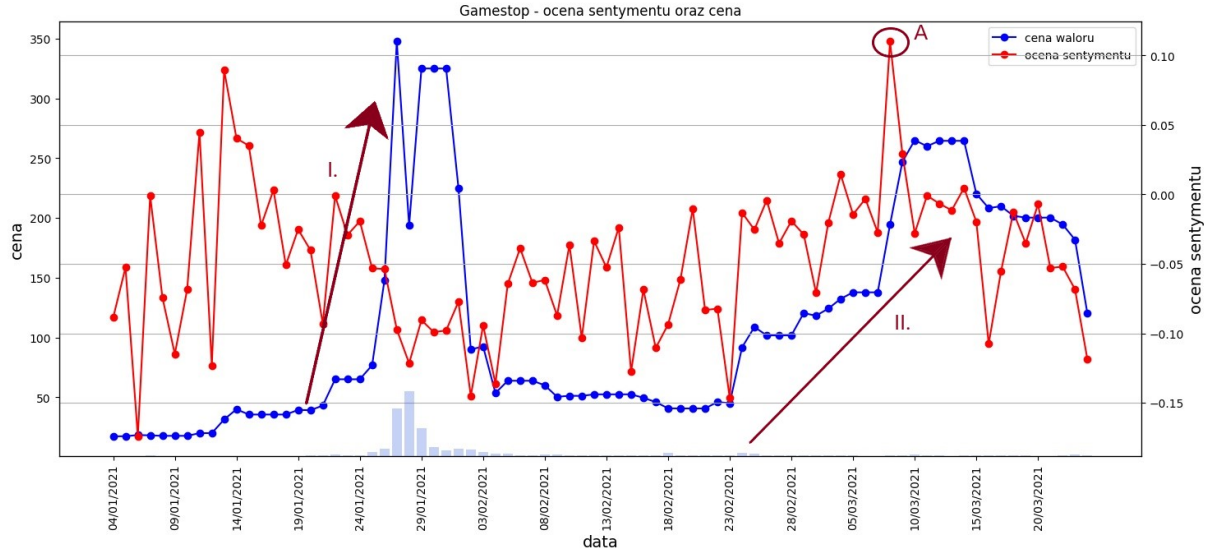
Rysunek 12: Pięciodniowa i dziesięciodniowa średnia krocząca oceny sentymentu oraz cena spółki Apple

Gamestop

Kolejną analizowaną spółką jest Gamestop, firma sprzedająca gry komputerowe oraz elektronikę. Akcje tej spółki stały się popularne w styczniu 2021 roku, gdy wśród użytkowników forum internetowego Reddit powstał ruch mający na celu grę przeciwko funduszom hedgingowym grającym na spadek kursu akcji. Prywatni inwestorzy zaczęli skupować udziały Gamestop, windując cenę do najwyższych historycznie poziomów, jednocześnie zmuszając duże fundusze do zamknięcia swoich krótkich pozycji (ang. *short squeeze*) oraz poniesienia gigantycznych strat [15]. Akcja ta ma również odzwierciedlenie w analizowanym zbiorze danych, ponieważ jeszcze na początku stycznia Gamestop dla przeciętnego inwestora był nieznaną spółką. Minimum dziennego wolumenu tweetów wynosi zaledwie 38, zaś maksimum blisko 24 tysięcy. W sumie analizie zostało poddane 87 tysięcy sentencji, z kryterium minimalnej liczby polubień równym 2. Efekty badania widoczne są na poniższych rysunkach.

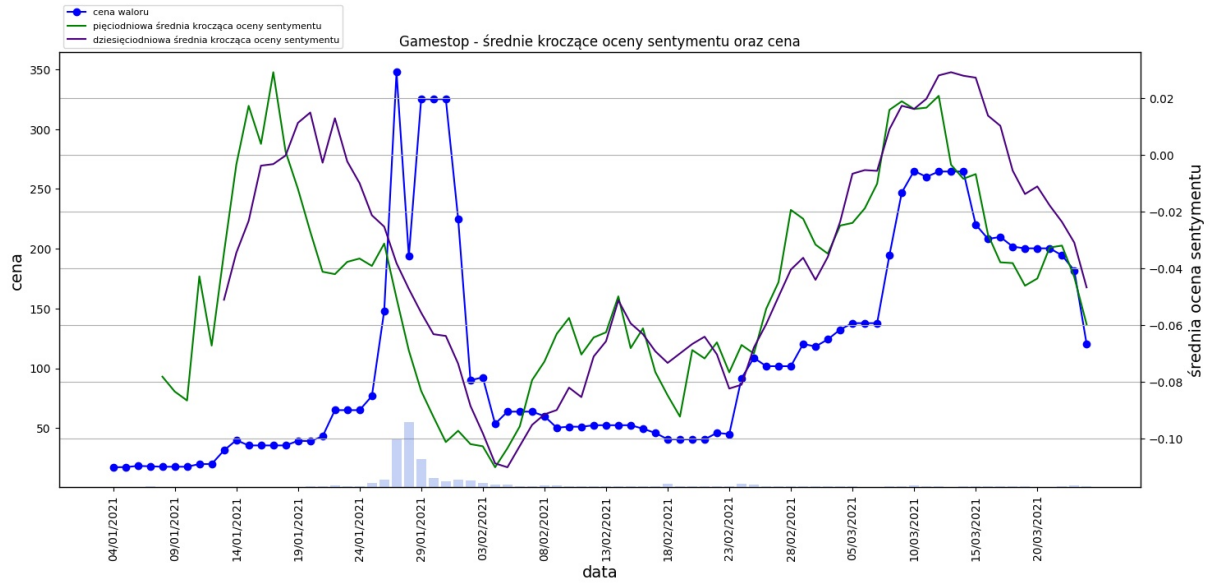
Dyskusja na forach nt. Gamestopu rozpoczęła się około 11 stycznia, gdy ogłoszono informacje o zatrudnieniu 3 dyrektorów wchodzących w skład zarządu. 13 stycznia spółka zanotowała blisko 50 procentowy wzrost - tak rozpoczął się kilkunastodniowy rajd, który spowodował wzrost wartości akcji spółki do poziomów bliskich 350 dolarów (strzałka I. na rysunku 13). Po kilkudniowym okresie wyznaczania szczytów cena spółki spadła o ponad 80%, co nie miało swojego odzwierciedlenia w analizie sentymentu. W drugiej połowie wykresu, gdy cena powróciła do poziomów bliskich styczniowych szczytów (strzałka II.), średnie kroczące sentymentu dobrze odzwierciedliły powrót optymizmu wśród inwestorów. Punkt A na wykresie to moment, w którym ogłoszono, że spółka planuje rozpoczęcie sprzedaży online, a ocena sentymentu osią-

gnęła szczyt na wykresie. Kilka dni po tym wydarzeniu cena waloru zaczęła spadać - tak jak średnie krocząc oceny sentymentu.



Rysunek 13: Dzienna ocena sentymentu oraz cena spółki Gamestop

Na pierwszy rzut oka powyższy wykres nie przedstawia widocznej korelacji. Obliczony współczynnik korelacji wynosi zaledwie 0.08. Z rysunku 13 można wyciągnąć wniosek, że pozytywny sentyment wyprzedza przyszły wzrost ceny, jednak należy zwrócić uwagę, że spośród wszystkich analizowanych spółek jest to jedynie odosobniony przypadek.

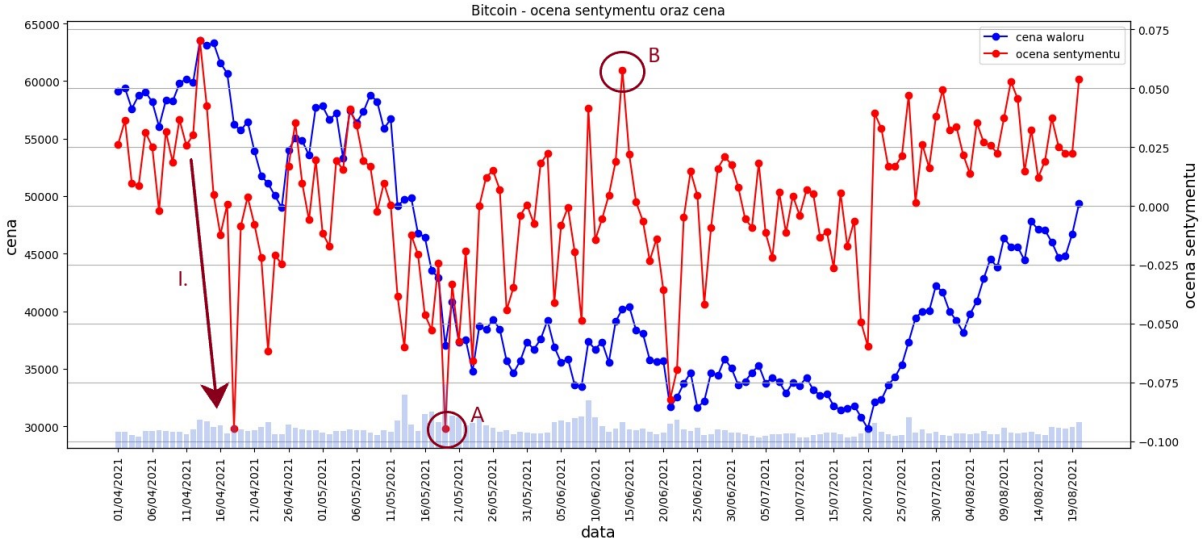


Rysunek 14: Pięciodniowa i dziesięciodniowa średnia krocząca oceny sentymentu oraz cena spółki Gamestop

Kształt wszystkich trzech wykresów na rysunku 14 wygląda bardzo podobnie, korelacja pięciodniowej średniej ruchomej do ceny jest równa 0.22, a dziesięciodniowa - 0.4 (korelacja odpowiednio: słaba oraz przeciętna).

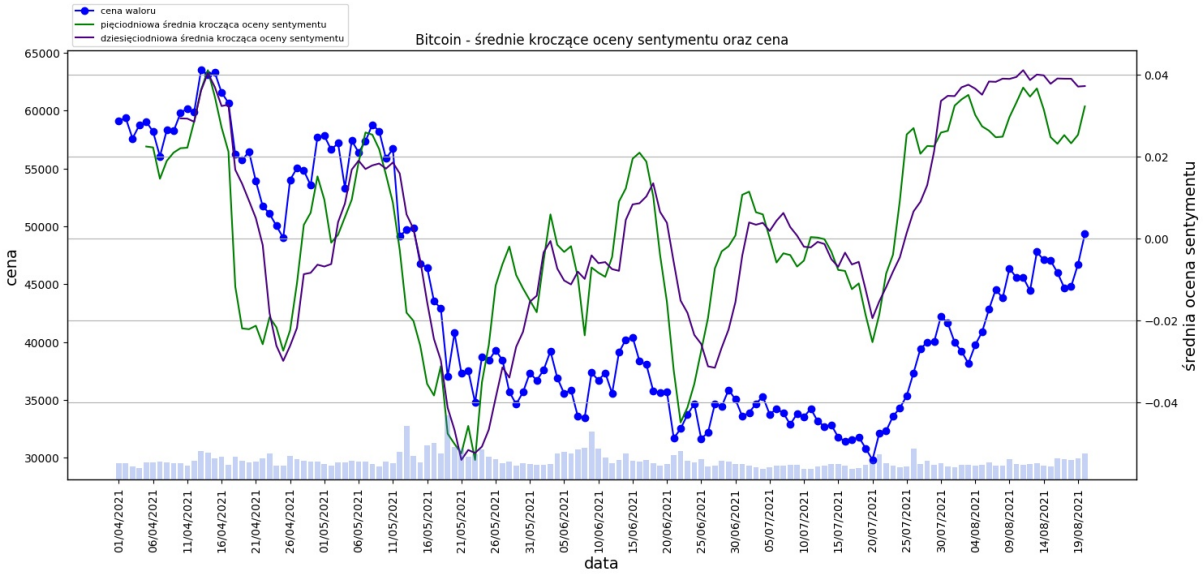
Bitcoin

Ostatnim przykładem wykorzystanym do porównania oceny sentymentu do ceny jest kryptowaluta Bitcoin. Bitcoin jest zdecentralizowaną kryptowalutą, którą można przesyłać w sieci peer-to-peer o tej samej nazwie. Notowany na największych giełdach kryptowalutowych na świecie, jest liderem rynku pod względem kapitalizacji rynkowej. Taki przykład został wybrany do analizy ze względu na dużą aktywność jaką przejawiają użytkownicy Twittera względem kryptowalut. Od kwietnia 2021 roku do końca sierpnia zostało pobrane blisko 380 tysięcy tweetów, z kryterium minimalnej liczby polubień pod tweetem równym 8. Ponadto, Bitcoin jest najbardziej zmiennym ze zbioru analizowanych walorów, co sprawia, iż możliwe jest szczegółowe zbadanie jak każdy nagły wzrost lub spadek ceny wpływa na nastroje uczestników rynku.



Rysunek 15: Dzienna ocena sentymentu oraz cena kryptowaluty Bitcoin

Rysunek 15 obrazuje korelację między danymi komunikatami tekstowymi, gdzie dzienne minimum tweetów to 1442, maksimum jest równe 9042, a średnia to 2677 tweetów dziennie. Współczynnik korelacji wynosi 0.28 co oznacza, że jest to korelacja słaba. Na uwagę zasługuje gwałtowny spadek sentymentu (zaznaczony strzałką I. na wykresie) trwający od 13 (szczyt cenowy oraz wydarzenie towarzyszące - oczekiwany debiut giełdowy giełdy kryptowalut Coinbase) do 18 kwietnia (największy dzienny spadek ceny od 2 miesięcy). Innym interesującym wydarzeniem, jest moment w którym poziom sentymentu (punkt A) był najniższy na wykresie gdy Bitcoin spadł poniżej 40 tysięcy dolarów oraz 14 czerwca (punkt B) gdy kurs powrócił na poziomy 40 tysięcy po tweecie Elona Muska.



Rysunek 16: Pięciodniowa i dziesięciodniowa średnia krocząca oceny sentymentu oraz cena kryptowaluty Bitcoin

Pięciodniowa średnia krocząca wykazała korelację z ceną na poziomie 0.38 natomiast dziesięciodniowa - 0.43. Rysunek 16 można podzielić na dwie części, pierwszą do końca maja, gdzie średnie kroczące niemal idealnie nakładają się na wykres ceny oraz pozostałą część, gdzie ruch średnich kroczących ma podobny kierunek co cena, jednak pojawia się zauważalna dywergencja pomiędzy sentymentem a ceną. Sentyment użytkowników rośnie silniej aniżeli cena, gdzie pod koniec sierpnia osiąga podobne wartości do poziomów na początku kwietnia, natomiast cena Bitcoina nie zdołała osiągnąć kwietniowych szczytów w badanym okresie czasu.

Wyniki strategii inwestycyjnej opartej na analizie sentymentu

Aby sprawdzić czy analiza sentymentu jest w stanie zapewnić inwestorom zyski, została zrealizowana prosta strategia inwestycyjna, która polega na kupnie danego waloru, gdy pięciodniowa średnia krocząca sentymentu przecina od dołu cenę, natomiast sprzedaż odbywa się gdy ta sama średnia krocząca przecina wykres ceny od góry. Taki plan inwestycyjny opiera się grze zgodnej z trendem sentymentu, gdy pojawia się trend wzrostowy - kupujemy, gdy spadkowy - sprzedajemy. Kupno oraz sprzedaż odbywało się tylko w analizowanym przedziale czasowym, co oznacza, że gdy znajdował się sygnał kupna ale brakowało sygnału sprzedaży (brak przecięcia linii średniej kroczącej od góry) taka transakcja nie została ujęta w poniższej tabeli.

Tabela 13 przedstawia wszystkie transakcje na podstawie przecięć pięciodniowej średniej kroczącej. W momencie wystąpienia sygnału kupna lub sprzedaży w sobotę lub niedzielę, kiedy obrót giełdowy jest wyłączony, została ujęta cena poniedziałkowego otwarcia. Skumulowany zysk ze wszystkich wymienionych transakcji wyniósł 350%, natomiast w żadnej transakcji nie została ujęta opłata transakcyjna, która z reguły wynosi około 0.1%. Ponadto, największy udział w zyskach ma transakcja na spółce Gamestop, której zysk wyniósł 283.3%. Odliczając taką transakcję zysk wynosi blisko 70%, natomiast należy zwrócić uwagę, że cena wszystkich analizowanych walorów (poza Bitcoinem) jest w trendzie wzrostowym, co także ma istotny wpływ

Tabela 13: Wyniki strategii kupna sprzedaży na podstawie analizy sentymentu

Walor	Data Kupna	Data Sprzedaży	Cena Kupna	Cena Sprzedaży	Zysk/Strata (%)
Tesla	20/08/2021	24/09/2021	680.26	708.49	4.1%
Tesla	25/09/2021	27/09/2021	773.12	791.36	2.4%
Tesla	01/10/2021	21/10/2021	775.22	894.00	15.3%
Tesla	22/10/2021	28/10/2021	909.68	1077.04	18.4%
Apple	28/06/2021	21/07/2021	134.78	145.40	7.9%
Apple	25/07/2021	05/08/2021	148.27	147.06	-0.8%
Apple	28/09/2021	29/09/2021	141.91	142.83	0.6%
Facebook	10/05/2021	04/06/2021	305.97	330.35	8.0%
Facebook	09/06/2021	07/07/2021	330.25	350.49	6.1%
Facebook	14/07/2021	15/07/2021	347.63	344.46	-0,9%
Facebook	29/07/2021	04/08/2021	358.32	358.92	0.2%
Facebook	11/08/2021	12/08/2021	359.96	362.65	0.7%
Facebook	22/09/2021	23/09/2021	343.21	345.96	0.8%
Gamestop	06/02/2021	13/03/2021	72.41	277.52	283.3%
Bitcoin	06/05/2021	08/05/2021	56396.52	58803.78	4.3%

na rezultat analizowanej strategii. Rezultatem stosowania takiej strategii są wysokie zyski, natomiast należy mieć na uwadze, że zbiór danych nie jest wystarczająco duży aby sprawdzić zachowanie średnich kroczących na większym przedziale czasowym oraz w różnych cyklach rynkowych (hossa oraz bessa), co czyni ryzykownym podejmowanie decyzji inwestycyjnych tylko na podstawie wyciągniętych wniosków.

Podsumowanie

W przeprowadzonym badaniu została zobrazowana zależność między nastrojem użytkowników Twittera a ceną akcji oraz Bitcoina. Dzięki odpowiedniemu przygotowaniu danych oraz wykorzystaniu modelu przetwarzania języka naturalnego finBERT rezultaty eksperymentu są zadowalające.

Analiza sentencji wykonana za pomocą NLP jest najnowocześniejszym narzędziem w dziedzinie badania kontekstu języka naturalnego a także klasyfikacji zdań co pokazało porównanie przeprowadzone w rozdziale 1.2. Zastosowując odpowiednie biblioteki programistyczne do procesu eksploracji, wstępnego przygotowania oraz wizualizacji danych, otrzymane wyniki są cennym narzędziem do badania sentymentu rynków finansowych, a także opracowania strategii inwestycyjnych na bazie wykazanych zależności. Próba wykazania korelacji między ceną a sentymentem rynku wykazała w większości przypadków korelację minimum przeciętną, co jest potwierdzeniem tezy wystosowanej w wielu publikacjach o tematyce ekonomicznej, że cena jest ściśle skorelowana z emocjami panującymi wśród inwestorów. Istotną kwestią jest jednak sprawdzenie poziomu nastrojów, tzn. obliczenie oraz uśrednienie oceny sentymentu, a następnie zbadanie czy nie nastąpiła zbyt duża zmiana w odniesieniu do ceny waloru, ze względu na pojawiający się na rynku strach bądź chciwość.

Analiza komunikatów tekstowych na podstawie wyszukiwania tweetów po słowie kluczowym niesie ze sobą potencjalne pułapki np. w przykładzie Apple nie wszystkie tweety wyszukiwane po słowie kluczowym dotyczyły samej spółki. Innym przykładem jest fraza „Bitcoin" której wyniki wyszukiwania bardzo często dotyczyły innych kryptowalut, a najczęściej na końcu pojawiał się hashtag Bitcoin w celu zwiększenia zasięgu odbiorców. Niemniej jednak, zdecydowana większość tweetów dotyczyła tematu badanych spółek, na co miał wpływ fakt, iż omawiane spółki są jednymi z najpopularniejszych wśród inwestorów młodszej generacji, często udzielających się na portalach społecznościowych.

Pomimo tego, iż trudno przewidywać cenę wyłącznie na podstawie sentymentu, inwestorzy są w stanie określić trend emocji towarzyszący obrotem danej akcji, przez co prawdopodobne jest uniknięcie kupna bądź sprzedaży w stanie ekstremalnej euforii bądź niepewności rynku dotyczącej ceny. Potencjalnymi kolejnymi krokami do udoskonalenia narzędzia jest aplikacja takich funkcjonalności jak: pomiar liczby wyszukiwań nazwy danego waloru (np. w wyszukiwarce Google), sprawdzenie sentymentu artykułów pojawiających się w serwisach informacyjnych (np. Yahoo, New York Times itp.), czy też zbadanie sentymentu osób dyskutujących na łamach innych portali społecznościowych (np. Reddit, Facebook). Tak zdobyte dane połączone ze wskaźnikami analizy technicznej bądź fundamentalnej, mogą być składnikami potencjalnego modelu tradingowego, handlującego na różnych interwałach akcjami bądź kryptowalutami istotnie przyczyniając się do rozwoju zyskownego inwestowania na rynkach finansowych.

Bibliografia

[1] *Natural Language Processing (by IBM Cloud Education)*. https://www.ibm.com/cloud/learn/natural-language-processing. Odwiedzono: 2021-12-20.

[2] Dan Jurafsky i James H. Martin. *Speech and language processing : an introduction to natural language processing, computational linguistics, and speech recognition*. Upper Saddle River, N.J.: Pearson Prentice Hall, 2009. ISBN: 9780131873216 0131873210. URL: http://www.amazon.com/Speech-Language-Processing-2nd-Edition/dp/0131873210/ref=pd_bxgy_b_img_y.

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee i Kristina Toutanova. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. 2019. arXiv: 1810.04805 [cs.CL].

[4] *Transformers*. https://towardsdatascience.com/transformers-89034557de14. Odwiedzono: 2021-12-28.

[5] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser i Illia Polosukhin. „Attention Is All You Need". W: *CoRR* abs/1706.03762 (2017). arXiv: 1706.03762. URL: http://arxiv.org/abs/1706.03762.

[6] Dogu Araci. „FinBERT: Financial Sentiment Analysis with Pre-trained Language Models". W: *CoRR* abs/1908.10063 (2019). arXiv: 1908.10063. URL: http://arxiv.org/abs/1908.10063.

[7] *BERT - transformers documentation*. https://huggingface.co/docs/transformers/model_doc/bert. Odwiedzono: 2021-12-20.

[8] C. Hutto i Eric Gilbert. „VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text". W: *Proceedings of the International AAAI Conference on Web and Social Media* 8.1 (maj 2014), s. 216–225. URL: https://ojs.aaai.org/index.php/ICWSM/article/view/14550.

[9] *Sentiment analysis - the lexicon based approach*. https://www.alphabold.com/sentiment-analysis-the-lexicon-based-approach/. Odwiedzono: 2021-12-20.

[10] Saurav Pradha, Malka N. Halgamuge i Nguyen Tran Quoc Vinh. „Effective Text Data Preprocessing Technique for Sentiment Analysis in Social Media Data". W: *2019 11th International Conference on Knowledge and Systems Engineering (KSE)*. 2019, s. 1–8. DOI: 10.1109/KSE.2019.8919368.

[11] Xiaodong LI, Haoran XIE, Li CHEN, Jianping WANG i Xiaotie DENG. „News impact on stock price return via sentiment analysis". English. W: *Knowledge-Based Systems* 69.1 (paź. 2014), s. 14–23. ISSN: 0950-7051. DOI: 10.1016/j.knosys.2014.04.022.

[12] Burton G Malkiel. „The efficient market hypothesis and its critics". W: *Journal of economic perspectives* 17.1 (2003), s. 59–82.

[13] *Efficient-market hypothesis*. https://en.wikipedia.org/wiki/Efficient-market_hypothesis. Odwiedzono: 2021-12-20.

[14] Jacob Benesty, Jingdong Chen, Yiteng Huang i Israel Cohen. „Pearson correlation coefficient". W: *Noise reduction in speech processing*. Springer, 2009, s. 1–4.

[15] *Historia GameStop a przyszłość rynków finansowych*. https://businessinsider.com.pl/biznes/inwestowanie/gamestop-historia-gamestop-a-przyszlosc-rynkow-finansowych/dgr109p. Odwiedzono: 2021-12-20.