Zastosowanie analizy SHAP w analizie sentymentu metodami NLP

Jakub Karaś i Karol Kozłowski

16 czerwca 2024

1 Treść zadania

Zastosowanie analizy SHAP w analizie sentymentu metodami NLP

Proszę zbudować model służący do analizy sentymentu tweetów zebranych w zbiorze danych:

https://www.kaggle.com/datasets/austinreese/trump-tweets

(podobnymi metodami jak przedstawione na zajęciach). Następnie dla zbudowanego modelu proszę przeprowadzić analizę SHAP i przeprowadzić dyskusję uzyskanych wyników.

2 Link do rozwiązania

https://github.com/KarolKozlowski22/MIO_projekt2024

3 Wykonanie

3.1 Nauczenie modelu

W ramach projektu zdecydowaliśmy się użyć zbioru danych traindata.csv dostępnego na platformie Kaggle do wytrenowania modelu.

Model sekwencyjny Keras składa się z następujących warstw:

1. Embedding: Warstwa osadzająca, zamienia słowa na wektory liczbowe.

2. SpatialDropout1D: Warstwa dropout, zapobiega przeuczeniu poprzez losowe wyłączanie neuronów
3. LSTM: Warstwa długiej krótkoterminowej pamięci (Long Short-Term Memory), zdolna do uczenia się zależności czasowych.
4. Dense: Warstwa gęsta z 512 neuronami i aktywacją ReLU.
5. Dense: Warstwa wyjściowa z 3 neuronami (klasy sentymentu) i aktywacją softmax do klasyfikacji.

Cały zbudowany model można zobaczyć w pliku train_model.ipynb.

Analiza wyników:

- Model osiągnął dokładność 66.7% na danych testowych, co wskazuje na umiarkowaną skuteczność w klasyfikacji sentymentu tweetów. Strata wynosząca 0.7770 sugeruje, że model ma jeszcze pewne pole do poprawy.
- Podjęliśmy próby poprawy wyników poprzez zwiększenie liczby neuronów oraz dodanie dodatkowych warstw do modelu. Niestety, te działania nie przyniosły znaczącej poprawy wyników.
- Możliwe, że poprawę performance'u można by osiągnąć poprzez zastosowanie innej architektury modelu, takiej jak np. Transformer, BERT lub innych zaawansowanych sieci neuronowych specjalizujących się w przetwarzaniu języka naturalnego. Jednak ze względu na ograniczony czas, nie zrealizowaliśmy tych zmian w ramach tego projektu.

3.2 Preprocessing danych

Następnie należało pobrać plik realdonaldtrump.csv z Kaggle i wykorzystać na nim nauczony model. Zanim jednak można było to zrobić należało wyczyścić niepotrzebne dane, które mogłyby negatywnie wpłynąć na uzyskany wynik:

- Usunięcie Nieistotnych Kolumn: usunęliśmy kolumny link, mentions oraz hashtags, ponieważ nie wnosiły one wartościowej informacji do analizy sentymentu.
- Usunięcie stopwords: pobraliśmy listę angielskich stop słów z pakietu NLTK. Stop słowa to często używane słowa, które nie mają dużego znaczenia informacyjnego (np. "and", "the").

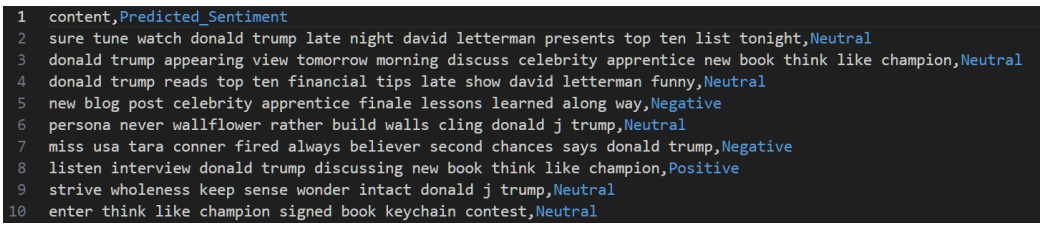
- Stemming: użyliśmy obiektu SnowballStemmer do sprowadzania słów do ich podstawowej formy. Stemming redukuje różne formy tego samego słowa do wspólnej podstawy (np. "running", "ran"i "runs"do "run")
- Zastosowaliśmy również:
    1. Usunięcie nazw użytkowników Twittera, linków oraz znaków specjalnych.
    2. Konwersja tekstu na małe litery, aby ujednolicić format tekstu.

Przeprowadziliśmy następujące operacje na tekście:

Podsumowując, preprocessing danych pozwolił na oczyszczenie i ujednolicenie tekstu, usunięcie nieistotnych informacji oraz ekstrakcję kluczowych cech czasowych, co jest kluczowe dla skutecznej analizy sentymentu.

3.3 Użycie modelu

W notatniku own_model_and_shap.ipynb użyliśmy naszego modelu na danych z twittera, które zawierały sam kontent tweetów.



```
content,Predicted_Sentiment
sure tune watch donald trump late night david letterman presents top ten list tonight,Neutral
donald trump appearing view tomorrow morning discuss celebrity apprentice new book think like champion,Neutral
donald trump reads top ten financial tips late show david letterman funny,Neutral
new blog post celebrity apprentice finale lessons learned along way,Negative
persona never wallflower rather build walls cling donald j trump,Neutral
miss usa tara conner fired always believer second chances says donald trump,Negative
listen interview donald trump discussing new book think like champion,Positive
strive wholeness keep sense wonder intact donald j trump,Neutral
enter think like champion signed book keychain contest,Neutral
```

Rysunek 1: Rezultat

Wynik jest zapisywany w osobnej kolumnie jako Negative/Neutral/Positive. Uzyskane wyniki można porównać z gotowym modelem SentimentIntensityAnalizer w pliku sia_and_shap.ipynb. Uzyskane wyniki trochę się różnią co jest spowodowane użyciem innego modelu, innych danych do uczenia i wielu innych czynników.

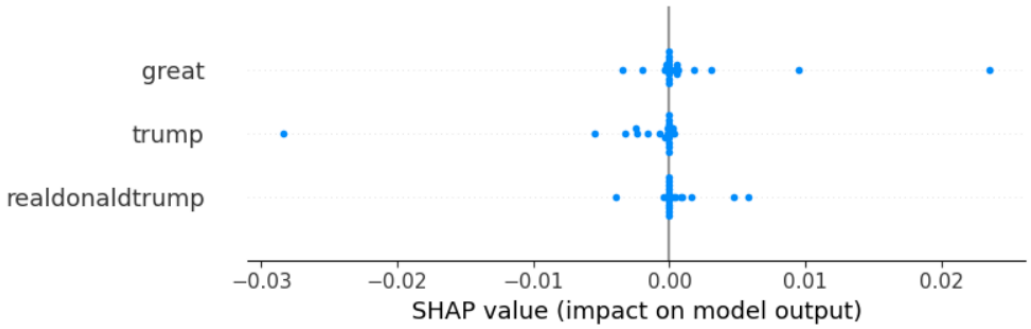
3.4 Użycie analizy SHAP do wyjaśnienia wyników

Analiza SHAP jest techniką wyjaśniania przewidywań modeli maszynowego uczenia się. SHAP wartości przypisują znaczenie każdej cechy w modelu, co pozwala na zrozumienie, jak poszczególne cechy wpływają na wyniki modelu.

1. **Przygotowanie Danych do SHAP**

- Podzieliliśmy dane na zbiory treningowe i testowe.
- Przetworzyliśmy dane testowe w taki sam sposób jak dane treningowe, aby były zgodne z modelem.

2. **Tworzenie i Wyjaśnianie za pomocą SHAP**
   - Wybraliśmy pierwsze 100 przykładów z przetworzonych danych testowych jako dane tła.
   - Stworzyliśmy obiekt wyjaśniający SHAP za pomocą `shap.KernelExplainer`, który wykorzystuje funkcję predykcji i dane tła.
   - Obliczyliśmy wartości SHAP dla pierwszych 100 przykładów z zestawu testowego, co pozwoliło na określenie wpływu poszczególnych cech na wyniki modelu.
3. **Wizualizacja Wyników**
   - Wykorzystaliśmy SHAP do generowania wykresów (*force plots*), które pokazują wpływ poszczególnych cech na przewidywania modelu dla konkretnych przykładów.
   - Stworzyliśmy wykres podsumowujący (*summary plot*), który przedstawia ogólny wpływ cech na przewidywania modelu, co pomaga zidentyfikować najważniejsze cechy.



Rysunek 2: Analiza shap

Jak widać na wykresie uzyskaliśmy tylko 3 cechy. Z pewnością nie jest to prawidłowa wartość i powinno być więcej cech jednak po wielu próbach to najlepszy rezultat jaki udało nam się uzyskać z tej analizy.