

Dokumentacja Projektu z przedmiotu Metody
Inteligencji Obliczeniowej
*Analiza SHAP istotności poszczególnych elementów
wektora wejściowego SSN*

21 czerwca 2025

Spis treści

1	Wstęp	2
2	Wykorzystane zbiory danych	2
3	Zbiór danych dotyczący sukcesu akademickiego uczniów	2
3.1	Preprocessing	2
3.2	Sieć neuronowa	2
3.3	Analiza SHAP	2
3.4	Otrzymane wyniki	3
3.4.1	Analiza dla pełnego zbioru danych	3
3.4.2	Wyniki na zredukowanym zbiorze cech oraz porównanie wyników modeli	4
3.5	Wnioski	5
3.5.1	Analiza dla pełnego zbioru danych	5
3.5.2	Shap i zredukowany zbiór danych	5
3.6	Podsumowanie	5
4	Zbiór danych dotyczący przewidywania upadłości polskich firm	5
4.1	Preprocessing	5
4.2	Sieć neuronowa	6
4.3	Analiza SHAP	6
4.4	Ponowny trening po redukcji cech	6
4.5	Otrzymane wyniki	7
4.5.1	Analiza dla pełnego zbioru danych	7
4.5.2	Analiza SHAP	9
4.5.3	Ponowny trening po redukcji cech	9
4.6	Wnioski	10
4.6.1	Analiza dla pełnego zbioru danych	10
4.6.2	Analiza SHAP	10
4.6.3	Ponowny trening po redukcji cech	10
4.7	Podsumowanie	10

1 Wstęp

Projekt obejmował tematykę analizy SHAP istotności poszczególnych elementów wektora wejściowego SSN. Postanowiliśmy wykonać cztery modele korzystające z czterech zbiorów danych i na nich przeprowadzić analizę.

2 Wykorzystane zbiory danych

Pierwszy zbiór dotyczył problemu przewidywania sukcesu akademickiego uczniów, natomiast drugi - przewidywania upadłości polskich firm.

3 Zbiór danych dotyczący sukcesu akademickiego uczniów

3.1 Preprocessing

W tym zbiorze należało przekształcić na początku etykiety na liczby, aby działał poprawnie w sieci neuronowej. Następnie podzieliliśmy zbiór na cechy oraz na zbiór uczący i testujący w proporcjach 80/20. Pamiętaliśmy o stratyfikacji, żeby dane były równomiernie podzielone oraz przeskalowaliśmy zbiór za pomocą StandardScaler.

3.2 Sieć neuronowa

Zbudowaliśmy model MLP; pierwsza warstwa ma 16 neuronów oraz funkcję aktywacji "relu". Warstwa wyjściowa ma tyle neuronów, ile mamy klas, czyli 3. Została użyta funkcja aktywacji softmax. Zastosowaliśmy również dropout, żeby nie dopuścić do przeuczenia. Testowane były również inne modele, bardziej złożone: dla modeli trzywarstwowych z ilością neuronów odpowiednio (128, 64, 3), (64, 32, 3) oraz (32, 16, 3) następowało spore zjawisko przeuczenia, dlatego postanowiliśmy uprościć budowę sieci.

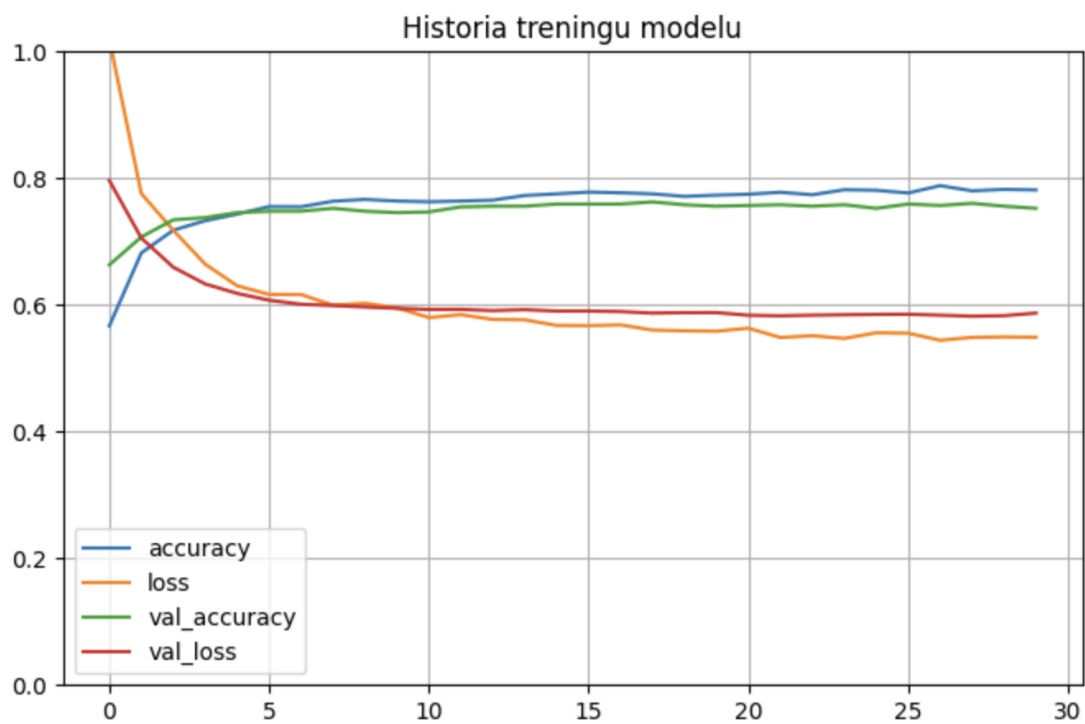
Wykonaliśmy kompilację modelu, a później trening, w którym liczba epok wynosiła 30. Następnie zrobiliśmy wykresy sprawdzające nasz model - zrobiliśmy historię treningu, sprawdziliśmy accuracy i macierz pomyłek.

3.3 Analiza SHAP

Shap umożliwił nam zobaczenie, jak model wygląda w środku. Za pomocą KernelExplainer można było zobaczyć, co wpływa na predykcję konkretnych klas. Ze zbioru testowego wylosowaliśmy 100 studentów i na podstawie tych próbek Explainer mógł utworzyć bazowy profil, z którego mógł wywnioskować istotność konkretnych cech. Explainer badał wpływ każdej z cech na każdą z trzech klas. Stworzyliśmy wizualizację przedstawiającą listę cech według ich uśrednionego wpływu na wszystkie klasy.

3.4 Otrzymane wyniki

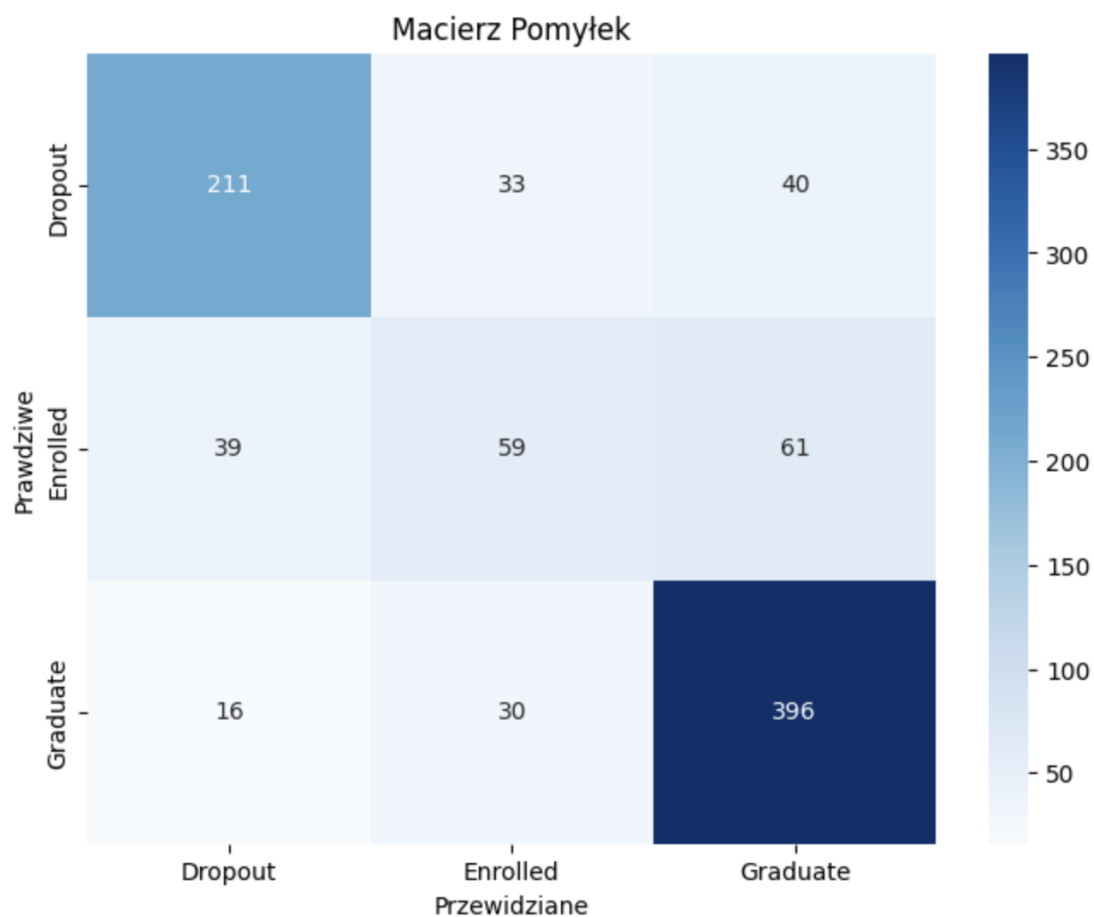
3.4.1 Analiza dla pełnego zbioru danych



Rysunek 1: Wykres nr 1

	precision	recall	f1-score	support
Dropout	0.79	0.74	0.77	284
Enrolled	0.48	0.37	0.42	159
Graduate	0.80	0.90	0.84	442
accuracy			0.75	885
macro avg	0.69	0.67	0.68	885
weighted avg	0.74	0.75	0.74	885

Rysunek 2:



Rysunek 3:

3.4.2 Wyniki na zredukowanym zbiorze cech oraz porównanie wyników modeli

```

Model z pełnym zbiorem cech:      Dokładność = 74.58%, Strata = 0.5971
Model ze zmniejszonym zbiorem cech: Dokładność = 77.06%, Strata = 0.5760

Raport klasyfikacji (zredukowany zbiór):
28/28 ----- 0s 3ms/step

```

	precision	recall	f1-score	support
Dropout	0.81	0.76	0.78	284
Enrolled	0.57	0.37	0.45	159
Graduate	0.79	0.92	0.85	442
accuracy			0.77	885
macro avg	0.72	0.68	0.69	885
weighted avg	0.76	0.77	0.76	885

Rysunek 4:

3.5 Wnioski

3.5.1 Analiza dla pełnego zbioru danych

Wykres dokładności na zbiorze treningowym stopniowo rośnie i stabilizuje się na poziomie około 80 procent, natomiast na wykresie strat widać widoczny spadek wartości. Widać delikatną rozbieżność między wykresami dokładności i straty dla danych treningowych i testowych, natomiast nie jest to zdecydowanie przeuczenie. Krzywe są zbieżne względem siebie.

W przypadku bardziej złożonych sieci, które również były testowane dla tego problemu, wykresy przedstawiały znaczącą rozbieżność dla wykresów z danych treningowych i testowych; szczególnie duże różnice pojawiły się dla treningu z ilością epok wynoszącą 50.

Rysunek drugi przedstawia raport klasyfikacji. Ogólna dokładność modelu wynosi 75 procent; dla klasy Graduate (absolwent) występuje duża skuteczność, poprawnie sklasyfikowanych zostało 90 procent danych. Precision i F1 score wynoszą odpowiednio 0.8 i 0.84. Jeśli chodzi o klasę Dropout (rezygnacja), model również radzi sobie dobrze, ale odrobinę gorzej niż przy poprzedniej klasie. Najsłabsze wyniki są dla klasy Enrolled (w trakcie studiów). Model poradził sobie słabiej niż w poprzednich przypadkach; jest w stanie poprawnie sklasyfikować poniżej 50 procent uczniów z tej grupy.

Na rysunku 3 mamy macierz, która pokazuje nam wizualnie dokładności rozbieżności podczas analizowania i rozpoznawania różnych klas przez model. Model ma problem z rozpoznaniem studentów rezygnujących od studentów kończących studia.

3.5.2 Shap i zredukowany zbiór danych

Na podstawie analizy Shap dowiedzieliśmy się, że najmniej istotne cechy w zbiorze to: **'Application order'**, **'Marital status'** oraz **'Application mode'**. Dzięki temu mogliśmy usunąć te trzy cechy i utworzyć nowy model. Sprawilo to, że nasz model usprawnił swoje działanie; dokładność wzrosła o około 3 punkty procentowe, a strata zmalała o około 2 punkty procentowe względem poprzedniej wersji modelu. Nieistotne cechy wprowadzały szum i utrudniały modelowi naukę cech, które realnie były ważne. Poprawiony model lepiej rozpoznawał i przyporządkowywał uczniów do jednej z trzech klas. Przyporządkowywanie do najbardziej problematycznej klasy, czyli "Enrolled" poprawiło się o 10 punktów procentowych, co jest znaczącą różnicą.

3.6 Podsumowanie

Dzięki analizie Shap mogliśmy zobaczyć, że usunięcie niewielkiej liczby cech spowodowało poprawę wyniku działania modelu. Nawet cechy o niskim znaczeniu mogą tworzyć szum zaburzający pracę modelu. Dowiedzieliśmy się więc, że inteligentne wybieranie cech jest bardzo istotne przy budowaniu takich modeli.

4 Zbiór danych dotyczący przewidywania upadłości polskich firm

4.1 Preprocessing

Ten zbiór ma dużo brakujących danych i niebalansowane klasy, dlatego istotne było dobre wykonanie preprocessingu. W procesie preprocessingu zajęliśmy się uzupełnieniem bra-

ków - zamieniliśmy znaki "?" na Nan, a następnie uzupełniliśmy je medianą. Podzieliliśmy zbiór na część testową i treningową i zastosowaliśmy StandardScaler do skalowania. Do zbalansowania zbioru zastosowaliśmy technikę SMOTE na zbiorze treningowym. Dzięki temu model mógł się poprawnie nauczyć wzorców dla obu klas (normalna firma i firma bankrutująca).

4.2 Sieć neuronowa

Stworzyliśmy model zbliżony do poprzedniego. Pierwsza warstwa miała 64 neurony, druga 32 i trzecia - 1, ponieważ jest to klasyfikacja binarna. Zastosowaliśmy funkcje "relu" i "sigmoid". W treningu wybraliśmy liczbę epok równą 50 i zwiększyliśmy batch size. Zastosowaliśmy też funkcję straty `binary_crossentropy`. Do sprawdzenia poprawności działania sieci wykorzystaliśmy metryki "precision" i "recall".

Sprawdziliśmy poprawność działania modelu na zbiorze oryginalnym, niezbalansowanym. Wykonaliśmy wykres z macierzą pomyłek.

4.3 Analiza SHAP

Za pomocą Shap sprawdziliśmy jak cechy wpływają na wartość bankructwa. Analiza została przeprowadzona na losowej próbce 100 firm ze zbioru testowego. Stworzyliśmy obiekt "explainer", który przeprowadził analizę. Explainer pobierał firmy ze zbioru (`X_test_sample`) i tworzył "sztuczne" firmy, dzięki którym mogły obliczać, jak ważna jest konkretna cecha dla danej firmy. Na podstawie wyników wykonaliśmy wykres podsumowujący najsłabsze cechy.

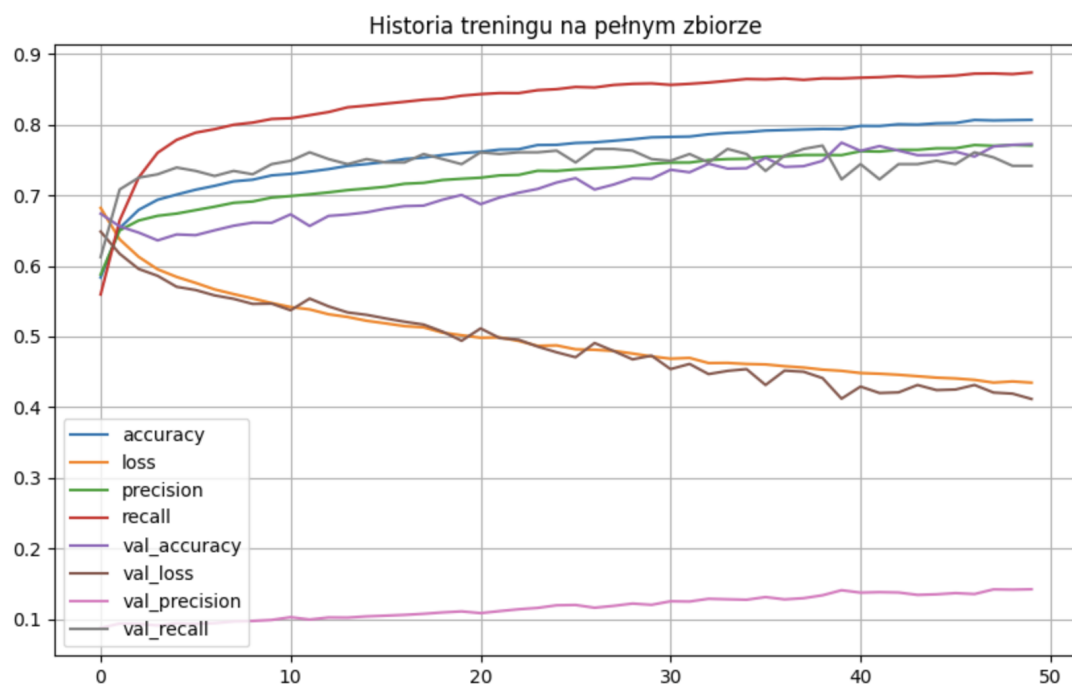
4.4 Ponowny trening po redukcji cech

Zrobiliśmy powtórkę z treningu poprzez powtórzenie go z usunięciem najmniej ważnych cech. Przez to, że cech jest dużo, bo aż 64, można było sporo z nich usunąć. Dla nowych danych powtórzyliśmy preprocessing, skalowanie, wykorzystanie SMOTE, a następnie wytrenowaliśmy nową wersję modelu.

Na koniec porównaliśmy obie wersje modelu.

4.5 Otrzymane wyniki

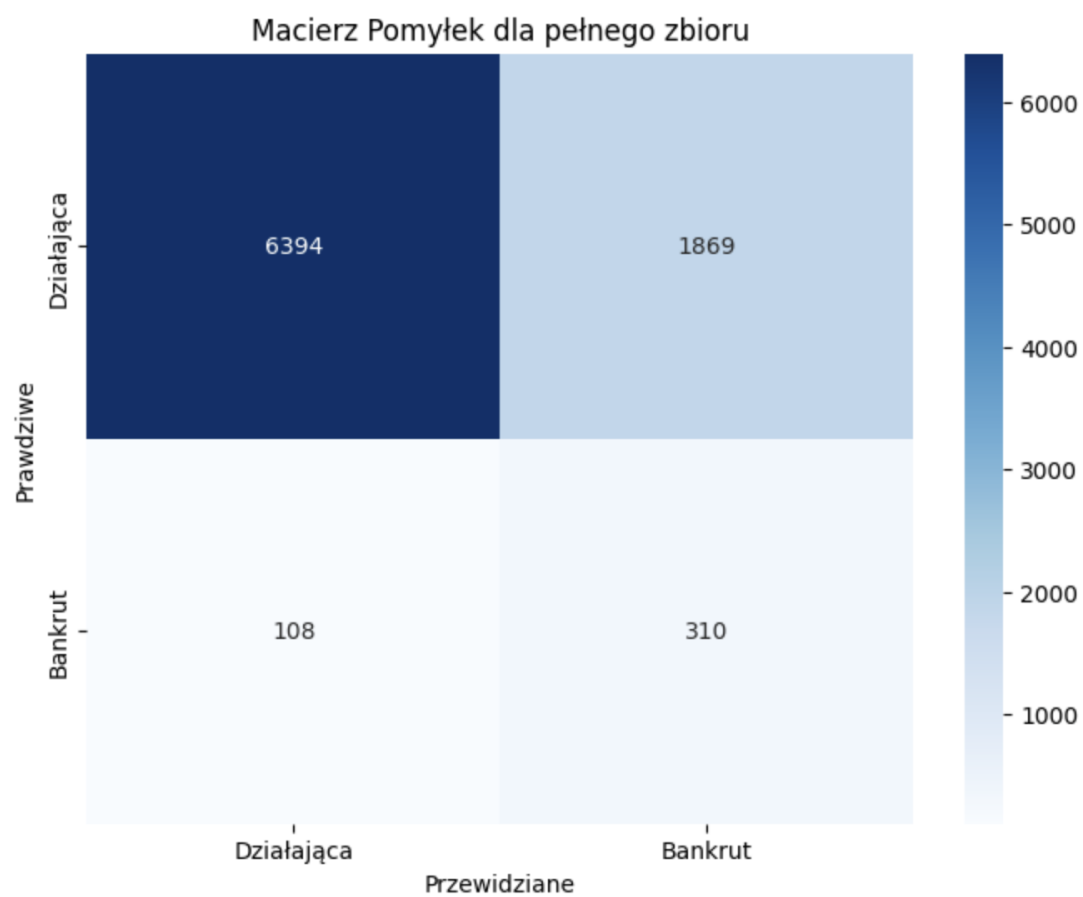
4.5.1 Analiza dla pełnego zbioru danych



Rysunek 5:

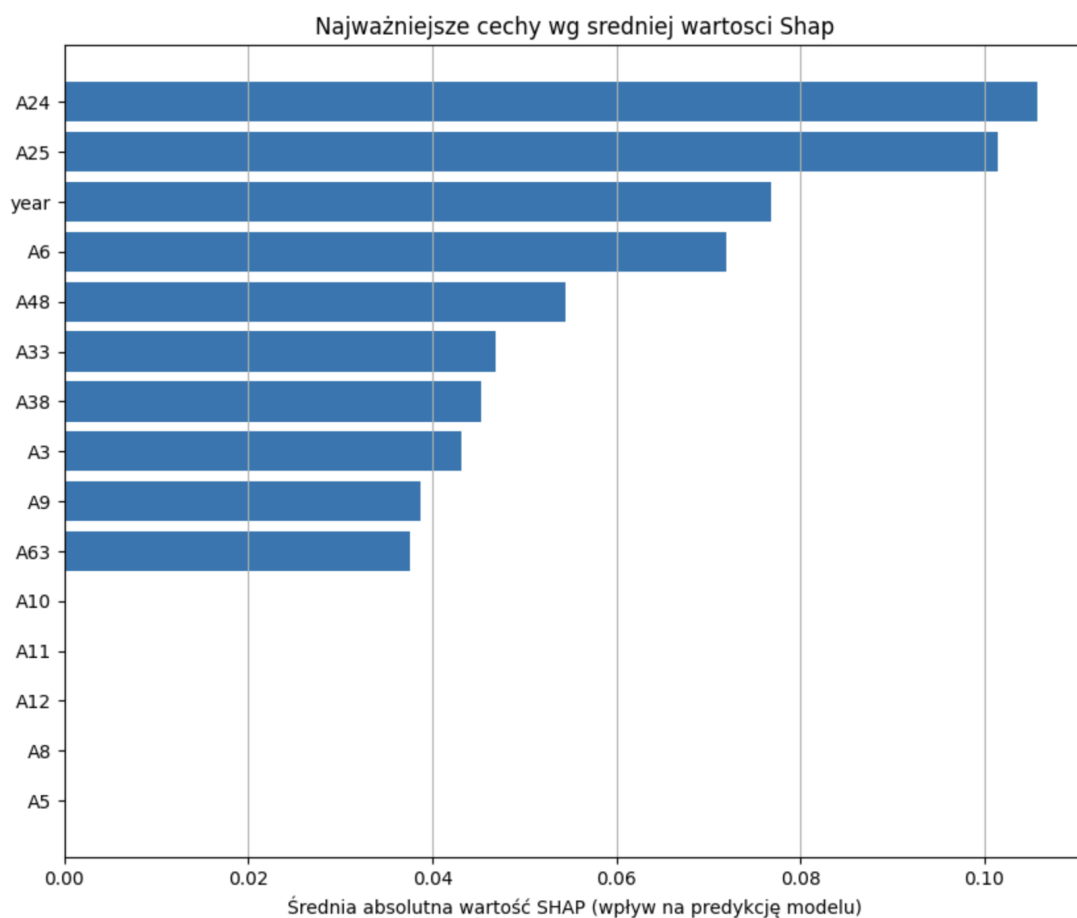
	precision	recall	f1-score	support
Działająca	0.98	0.77	0.87	8263
Bankrut	0.14	0.74	0.24	418
accuracy			0.77	8681
macro avg	0.56	0.76	0.55	8681
weighted avg	0.94	0.77	0.84	8681

Rysunek 6:



Rysunek 7:

4.5.2 Analiza SHAP



Rysunek 8:

4.5.3 Ponowny trening po redukcji cech

```
Model z pełnym zbiorem cech:  
loss: 0.6636  
compile_metrics: 0.7001  
  
Model ze zmniejszonym zbiorem cech:  
loss: 0.4291  
compile_metrics: 0.7537
```

Rysunek 9:

4.6 Wnioski

4.6.1 Analiza dla pełnego zbioru danych

Krzywe dokładności dla danych testowych i treningowych są stosunkowo blisko siebie, więc nie następuje przeuczenie. Krzywe straty tak samo, natomiast widać różnicę między recall dla zbioru testowego i treningowego.

Ogólna dokładność wynosi 77 procent, ale jest to niezbyt istotna informacja, ponieważ ponad 95 procent firm w zbiorze nie bankrutuje. Ważne jest natomiast, że dla klasy "Bankrut" model poprawnie identyfikuje 74 procent wszystkich firm, które faktycznie zbankrutowały. Wynika z tego więc, że technika SMOTE umożliwiła modelowi dobre nauczanie się wzorców. Niestety precision ma niską wartość, co oznacza, że model wykonał bardzo dużo "fałszywych alarmów", co nie zawsze może być dobrym rozwiązaniem. Z tego też powodu wartość F1 score jest dość niska.

4.6.2 Analiza SHAP

Na wykresie widać 15 najważniejszych cech, które mają wpływ na decyzję modelu. Na osi x mamy średnią absolutną wartość Shap, która pokazuje, jak bardzo dana cecha ma wpływ na predykcję. Najważniejsze w rankingu są **(przychody brutto + amortyzacja) / zobowiązania ogółem** oraz **(przychody brutto + odsetki) / aktywa ogółem**. Te cechy opisują, jak dana firma dobrze radzi sobie z generowaniem przychodów w porównaniu do zadłużenia i posiadanych aktywów. Model wie więc, że niska zdolność firmy do generowania przychodów jest jednym z głównych czynników wpływających na jej przyszły sukces lub upadek. Ważną cechą jest również 'year' określający warunki ekonomiczne w danym roku w kraju. Wynika z tego, że ryzyko bankructwa dla danej firmy nie jest stałe w czasie. Ważną cechą jest również wskaźnik płynności bieżącej, czyli zdolność firmy do regulowania bieżących zobowiązań.

Na wykresie widać, że kluczowe jest 10 cech, natomiast wszystkie pozostałe mogą stanowić szum informacyjny. Można stąd wywnioskować, że przy rozwijaniu modelu można usunąć wiele cech i przeprowadzić kolejne treningi.

4.6.3 Ponowny trening po redukcji cech

Dzięki Shap nasz model zanalizował i znalazł 15 najmniej istotnych cech: ['A49', 'A51', 'A50', 'A53', 'A54', 'A55', 'A52', 'A56', 'A57', 'A59', 'A58', 'A60', 'A61', 'A62', 'A64']. Dzięki temu można było stworzyć nowy model po redukcji cech, który jest prostszy i bardziej wydajny.

Po wykonaniu kolejnego treningu porównaliśmy modele.

Usunięcie najmniej istotnych cech poprawiło bardzo wydajność modelu. Na rysunku widać, że strata znacząco spadła a metryka jakości działania modelu wzrosła. Potwierdza to więc efektywność Shap, który może pomagać w analizowaniu modelu i poprawianiu jego jakości.

4.7 Podsumowanie

Zbiór danych dotyczących bankructwa firm okazał się być dość wymagający i skomplikowany w przygotowaniu. Gdyby nie preprocessing oraz próbkowanie metodą SMOTE, model mógłby nie działać efektywnie. Ten model działa efektywnie, jednak miał sporo przypadków "fałszywych alarmów". Może to być przydatne, jeśli chcielibyśmy budować

narzędzie do wczesnego ostrzegania. Metoda Shap pomogła przy korzystaniu modelu, ponieważ można było zobaczyć, że model podejmuje decyzje na podstawie logicznych przesłanek finansowych (np. rentowność, płynność). Shap pomógł również zidentyfikować cechy o małym znaczeniu, dzięki czemu można było zredukować wymiarowość. To doprowadziło do powstania skureczniejszego, prostszego i bardziej wydajnego obliczeniowo modelu. Usunięcie szumu spowodowało, że model lepiej się uczył i generalizował.