**Sztuczna Inteligencja i Inżynieria Wiedzy**

**Zadanie 4**

Podstawy maszynowego uczenia na przykładzie klasyfikacji tekstu

## Zbadanie hiperparametrów SVM

W danym eksperymencie zostaną zbadane hiperparemetry SVM z jądrem Gaussa (rbf)

Stałe: Parametry przy ekstrakcji cech:

*max\_features = 3000,*

*min\_df = 10,*

*max\_df = 65%*

Zmienne: ***Gamma***: [0.1, 1, 10]

***C:*** [0.01, 1, 10]

1. Przy C = 1:

* Gamma = 0.1:

precision recall f1-score support

0 0.67 0.26 0.37 54

1 0.82 0.88 0.85 215

2 0.73 0.26 0.38 74

3 0.55 0.90 0.68 197

4 1.00 0.03 0.06 65

macro avg 0.75 0.47 0.47 605

weighted avg 0.73 0.67 0.61 605

*Accurancy 0.6661157024793388*

* Gamma = 1:

precision recall f1-score support

0 0.69 0.58 0.63 66

1 0.86 0.87 0.86 224

2 0.80 0.53 0.63 74

3 0.67 0.90 0.76 192

4 0.71 0.24 0.36 49

macro avg 0.74 0.62 0.65 605

weighted avg 0.76 0.75 0.74 605

***Accurancy 0.7520661157024794***

* Gamma = 10:

precision recall f1-score support

0 0.00 0.00 0.00 54

1 0.36 1.00 0.52 215

2 0.00 0.00 0.00 74

3 0.00 0.00 0.00 197

4 0.00 0.00 0.00 65

macro avg 0.07 0.20 0.10 605

weighted avg 0.13 0.36 0.19 605

*Accurancy 0.35537190082644626*

1. Przy C = 0.01:

• Gamma = 0.1:

precision recall f1-score support

0 0.00 0.00 0.00 54

1 0.36 1.00 0.52 215

2 0.00 0.00 0.00 74

3 0.00 0.00 0.00 197

4 0.00 0.00 0.00 65

macro avg 0.07 0.20 0.10 605

weighted avg 0.13 0.36 0.19 605

*Accurancy 0.35537190082644626*

• Gamma = 1:

precision recall f1-score support

0 0.00 0.00 0.00 54

1 0.36 1.00 0.52 215

2 0.00 0.00 0.00 74

3 0.00 0.00 0.00 197

4 0.00 0.00 0.00 65

macro avg 0.07 0.20 0.10 605

weighted avg 0.13 0.36 0.19 605

*Accurancy 0.35537190082644626*

• Gamma = 10:

precision recall f1-score support

0 0.00 0.00 0.00 54

1 0.36 1.00 0.52 215

2 0.00 0.00 0.00 74

3 0.00 0.00 0.00 197

4 0.00 0.00 0.00 65

macro avg 0.07 0.20 0.10 605

weighted avg 0.13 0.36 0.19 605

*Accurancy 0.35537190082644626*

1. Przy C = 10:

* Gamma = 0.1:

precision recall f1-score support

0 0.58 0.57 0.58 54

1 0.82 0.87 0.85 215

2 0.62 0.47 0.54 74

3 0.72 0.83 0.77 197

4 0.69 0.42 0.52 65

macro avg 0.69 0.63 0.65 605

weighted avg 0.73 0.74 0.73 605

*Accurancy 0.7355371900826446*

* Gamma = 1:

precision recall f1-score support

0 0.63 0.57 0.60 54

1 0.83 0.88 0.85 215

2 0.68 0.49 0.57 74

3 0.69 0.85 0.76 197

4 0.81 0.40 0.54 65

macro avg 0.73 0.64 0.66 605

weighted avg 0.75 0.74 0.73 605

***Accurancy 0.743801652892562***

* Gamma = 10:

precision recall f1-score support

0 0.00 0.00 0.00 54

1 0.36 1.00 0.53 215

2 0.00 0.00 0.00 74

3 0.00 0.00 0.00 197

4 0.00 0.00 0.00 65

macro avg 0.07 0.20 0.11 605

weighted avg 0.13 0.36 0.19 605

*Accurancy 0.35537190082644626*

Wniosek: Po zbadaniu hiperparametrów SVM i przeanalizowaniu otrzymanych wyników, można stwierdzić, że najlepszą wartością parametru **Gamma** jest 1 oraz najlepsza wartość parametru **C** to 10.

## Zbadanie parametrów przy ekstrakcji cech

Ekstrakcja cech za pomocą funkcji *TfidVectorizer,* która przekształca dokumenty na macierz funkcji TF-IDF:

*tfidfconverter = TfidfVectorizer (max\_features, min\_df, max\_df), gdzie:*

* ***Max\_features****:* najczęściej występujących słów jako funkcji do trenowania naszego klasyfikatora
* ***Min\_df*:** ignoruj terminy, których częstotliwość dokumentów jest znacznie niższa niż podana liczba
* ***Max\_df:*** uwzględnij tylko te słowa, które pojawiają się w maksymalnie n% wszystkich dokumentów

Stałe: Parametry SVM:

*Gamma = 1*

*C = 10*

Zmienne: Parametry przy ekstrakcji cech:

***max\_features*** *= 1500, 3000, 4000*

1. Max\_features = 1500, min\_df = 10, max\_df = 65%:

* NB z 10-krotną walidacją krzyżową:

precision recall f1-score support

0 0.74 0.49 0.58 103

1 0.74 0.83 0.78 307

2 0.70 0.41 0.52 102

3 0.61 0.83 0.70 305

4 0.75 0.17 0.27 90

macro avg 0.71 0.55 0.57 907

weighted avg 0.69 0.68 0.65 907

*Accurancy: 0.679162072767365*

* SVM (Gamma = 1, C = 10) z 10-krotną walidacją krzyżową:

precision recall f1-score support

0 0.70 0.58 0.63 103

1 0.77 0.84 0.80 307

2 0.61 0.48 0.54 102

3 0.67 0.81 0.73 305

4 0.74 0.32 0.45 90

macro avg 0.70 0.61 0.63 907

weighted avg 0.71 0.71 0.69 907

*Accurancy 0.7067254685777288*

1. Max\_features = 3000, min\_df = 10, max\_df = 65%:

* NB z 10-krotną walidacją krzyżową:

precision recall f1-score support

0 0.64 0.39 0.48 54

1 0.80 0.92 0.85 215

2 0.67 0.27 0.38 74

3 0.61 0.87 0.72 197

4 0.93 0.20 0.33 65

macro avg 0.73 0.53 0.55 605

weighted avg 0.72 0.70 0.66 605

*Accurancy 0.6991735537190082*

* SVM (Gamma = 1, C = 10) z 10-krotną walidacją krzyżową:

precision recall f1-score support

0 0.69 0.58 0.63 66

1 0.86 0.87 0.86 224

2 0.80 0.53 0.63 74

3 0.67 0.90 0.76 192

4 0.71 0.24 0.36 49

macro avg 0.74 0.62 0.65 605

weighted avg 0.76 0.75 0.74 605

*Accurancy 0.7520661157024794*

1. Max\_features = 4000, min\_df = 10, max\_df = 65%:

* NB z 10-krotną walidacją krzyżową:

precision recall f1-score support

0 0.89 0.48 0.63 66

1 0.84 0.91 0.87 228

2 0.86 0.38 0.53 81

3 0.59 0.86 0.70 187

4 0.82 0.21 0.33 43

macro avg 0.80 0.57 0.61 605

weighted avg 0.77 0.73 0.71 605

***Accurancy 0.7289256198347107***

* SVM (Gamma = 1, C = 10) z 10-krotną walidacją krzyżową:

precision recall f1-score support

0 0.83 0.67 0.74 66

1 0.84 0.90 0.87 228

2 0.76 0.46 0.57 81

3 0.66 0.80 0.72 187

4 0.73 0.51 0.60 43

macro avg 0.76 0.67 0.70 605

weighted avg 0.76 0.76 0.75 605

***Accurancy 0.7570247933884298***

Wniosek: Po wykonaniu danego eksperymentu można dojść do wniosków, że im większa jest wartość wskaźnika najczęściej występujących słów, tym lepiej maszyna się uczy.

## Zbadanie parametru wybrania K najlepszych cech na podstawie statystyki Chi-kwadrat

Selekcja cech polega na wybraniu cech według k najwyższych wyników. Celem tego badania jest obserwacja wpływu wybrania wartości dla parametru k.

Stałe: Parametry przy ekstrakcji cech oraz uczeniu modelu:

*max\_features = 6000,*

*min\_df = 10,*

*max\_df = 65%*

*Gamma* = 10

*C = 1*

Zmienne: k: [100, 1000, 1500, 3000, 4000]

1. Wyniki predykcji na danych testowych przy **k = 100:**

* NB z 10-krotną walidacją krzyżową:

Time: 0.8s

precision recall f1-score support

macro avg 0.81 0.33 0.31 605

weighted avg 0.66 0.56 0.48 605

*Accurancy 0.5570247933884298*

* SVM (Gamma = 10, C = 1) z 10-krotną walidacją krzyżową:

Time: 9.5s

precision recall f1-score support

macro avg 0.76 0.60 0.63 605

weighted avg 0.76 0.72 0.71 605

*Accurancy 0.7239669421487603*

1. Wyniki predykcji na danych testowych przy **k = 1000:**

* NB z 10-krotną walidacją krzyżową:

Time: 0.1s

precision recall f1-score support

macro avg 0.83 0.57 0.61 605

weighted avg 0.78 0.73 0.70 605

*Accurancy 0.7256198347107438*

* SVM (Gamma = 10, C = 1) z 10-krotną walidacją krzyżową:

Time: 1m 27s

precision recall f1-score support

macro avg 0.69 0.66 0.67 605

weighted avg 0.74 0.75 0.75 605

*Accurancy 0.7504132231404959*

1. Wyniki predykcji na danych testowych przy **k = 1500:**

* NB z 10-krotną walidacją krzyżową:

Time: 0.1s

precision recall f1-score support

macro avg 0.84 0.53 0.57 605

weighted avg 0.77 0.70 0.67 605

*Accurancy 0.6975206611570248*

* SVM (Gamma = 10, C = 1) z 10-krotną walidacją krzyżową:

Time: 2m 21s

precision recall f1-score support

macro avg 0.69 0.64 0.66 605

weighted avg 0.71 0.72 0.71 605

*Accurancy 0.715702479338843*

1. Wyniki predykcji na danych testowych przy **k = 3000:**

* NB z 10-krotną walidacją krzyżową:

Time: 0.2s

precision recall f1-score support

macro avg 0.78 0.56 0.59 605

weighted avg 0.75 0.71 0.68 605

*Accurancy 0.7057851239669422*

* SVM (Gamma = 10, C = 1) z 10-krotną walidacją krzyżową:

Time: 6m 23s

precision recall f1-score support

macro avg 0.72 0.69 0.70 605

weighted avg 0.75 0.75 0.75 605

*Accurancy 0.7487603305785124*

1. Wyniki predykcji na danych testowych przy **k = 4000:**

* NB z 10-krotną walidacją krzyżową:

Time: 0.2s

recision recall f1-score support

macro avg 0.75 0.50 0.52 605

weighted avg 0.73 0.69 0.65 605

*Accurancy 0.6859504132231405*

* SVM (Gamma = 10, C = 1) z 10-krotną walidacją krzyżową:

Time: 7m 59s

precision recall f1-score support

macro avg 0.74 0.63 0.66 605

weighted avg 0.75 0.73 0.72 605

*Accurancy 0.7338842975206612*

Wniosek: Posiadanie nieistotnych cech w danych może zmniejszyć dokładność modeli i sprawić, że model będzie uczył się na podstawie nieistotnych cech. Dany eksperyment pozwala stwierdzić, że zmniejszenie liczby zmiennych wejściowych (cech) zarówno zmniejsza koszt obliczeniowy, i w niektórych przypadkach poprawia wydajności modelu. Optymalną wartością parametru k jest 1500.

## Eksperyment podziału danych na treningowe i testowe

Dany eksperyment polega na sprawdzeniu wybrania proporcji podziału danych na treningowe i testowe przy uwzględnieniu 10-krotnej walidacji krzyżowej.

Stałe: Parametry przy ekstrakcji cech oraz uczeniu modelu:

*max\_features = 6000, min\_df = 10, max\_df = 65%*

*Gamma* = 10

*C = 1,*

*K = 1500*

Zmienne: *podział danych: 90/10, 80/20, 70/30, 60/40*

1. 90/10

* NB z 10-krotną walidacją krzyżową: *Accurancy 0.6975206611570248*
* SVM (Gamma = 10, C = 1) z 10-krotną walidacją krzyżową: *Accurancy 0.715702479338843*

1. 80/20

* NB z 10-krotną walidacją krzyżową: *Accurancy 0.7057851239669422*
* SVM (Gamma = 10, C = 1) z 10-krotną walidacją krzyżową: *Accurancy 0.7206611570247934*

1. 70/30

* NB z 10-krotną walidacją krzyżową: *Accurancy 0.7045203969128997*
* SVM (Gamma = 10, C = 1) z 10-krotną walidacją krzyżową: *Accurancy 0.7342888643880926*

1. 60/40

* NB z 10-krotną walidacją krzyżową: *Accurancy 0.6895411326994626*
* SVM (Gamma = 10, C = 1) z 10-krotną walidacją krzyżową: *Accurancy 0.7209590739975197*

Wniosek: Po przeprowadzeniu eksperymentu, można dojść do wniosku, że nie ma optymalnego podziału procentowego. Przy wybraniu podziału procentowego należy uwzględni takie kwestie, jak koszt obliczeniowy w uczeniu modelu, koszt obliczeniowy w ocenie modelu oraz jak dużo jest danych. Wielkość danych jest istotna, ponieważ przy niedobrym wybraniu podziału może dojść do tego, że do oceny modelu trafi za mało danych.

## Badanie hiperparametru alpha dla wielomianowego klasyfikatora NB

Dany eksperyment polega na zbadaniu hiper-parametru alpha w klasyfikatorze wielomianowym NB, który jest parametrem wygładzania addytywnego (Laplace/Lidstone) (0 oznacza brak wygładzania).

Stałe: Parametry przy ekstrakcji cech oraz uczeniu modelu:

*max\_features = 6000, min\_df = 10, max\_df = 65%*

*k = 1500*

Zmienne: *alpha: [0.0001, 0.001, 0.01, 0.1, 0.5, 1]*

Za pomocą GridSearch, który pozwala na wyszukiwanie określonych wartości parametrów dla estymatora, została znaleziona wartość parametru alpha, dla której wynik predykcji jest najlepszy:

* Best alpha: **{'alpha': 0.1}**
* Best score: 0.7516945493793848
* Accurancy 0.7487603305785124