

1 Wstęp

Zaimplementowano model do predykcji wieloklasowej, który na podstawie treści tweeta (w języku angielskim) predykuje dodaną na końcu emoji. Uwzględniono 45 emoji, które zostały przedstawione na rysunku poniżej. Każde emoji będzie reprezentowane w dalszej części pracy przez przypisany mu indeks, będący pozycją na liście 1.



Figure 1: Lista uwzględnionych emoji.

1.1 Dane

Zbiór danych zawiera ponad 300 tysięcy tweetów zscrapowanych z Twittera na przestrzeni kwietnia i listopada 2019 roku. Każdy z tweetów zakończony jest emoji będącym jego etykietą. Rozkład klas został zaprezentowany na rysunku 2.

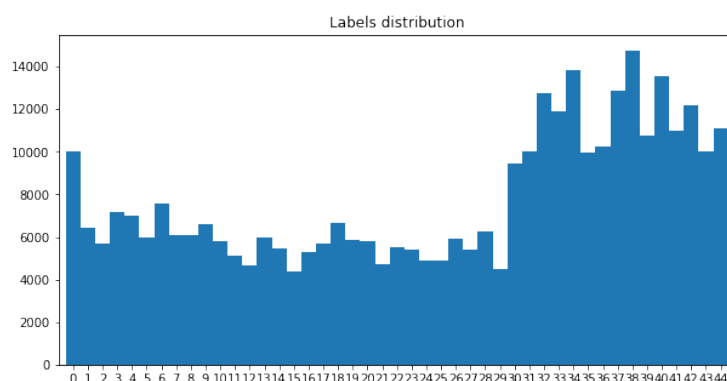
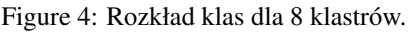
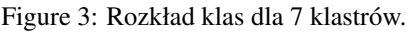


Figure 2: Rozkład klas dla 45 emoji.

1.2 Klasteryzacja

Należy zwrócić uwagę na fakt, że emoji są używane w bardzo niejednoznaczny i subiektywny sposób. Dodatkowo analizując ich kontekst użycia (na podstawie chmur słów i wektorów utworzonych z wykorzystaniem modelu FastText) można stwierdzić, że pewne grupy używane są w podobnych kontekstach. Można więc przypuszczać, że klasteryzacja emoji (na podstawie ich wektorów osadzeń), a następnie predykcja klastra, do którego należą pozwoli uzyskać lepsze wyniki niż w przypadku 45 klas.

W związku z powyższym przeprowadzono klasteryzację z zastosowaniem modelu k-średnich dla k równego 7, 8, 9 i 10. Poniżej zaprezentowano skład grup i rozkład klas dla każdego z badanych k.



W dalszych sekcjach zostaną opisane badania parametrów i hiperparametrów modelu oraz analiza uzyskanych wyników.

2 Wyniki

Sekcja zawiera szczegółowe dane dotyczące przeprowadzonych eksperymentów.

2.1 Klasyfikacja emoji

Dla podstawowego wariantu modelu przebadano różne kombinacje parametrów i hiperparametrów i wybrano ten model, dla którego wartość metryki f-score jest najwyższa (ze względu na niezbalansowanie zbioru accuracy jest jedynie miarą pomocniczą). Podsumowanie badań zamieszczone jest w tabeli 7 znajdującej się poniżej.

	num_labels	init_weights	rnn_type	rnn_units	bidirectional	attention	mask	dense_units	optimizer	lr	num_epochs	batch_size	x_len	acc_Train	acc_Test	f_score_Test	max_acc_Test
0	45	True	LSTM	[128]	True	True	True	64	Adam	0.0010	15	500	42	0.212482	0.166910	0.127161	0.170763
1	45	True	LSTM	[128]	True	True	False	64	Adam	0.0010	15	500	42	0.208362	0.168694	0.124329	0.169634
2	45	True	LSTM	[128]	False	True	True	64	Adam	0.0010	15	500	42	0.194998	0.167299	0.119793	0.167299
3	45	True	LSTM	[128]	False	True	False	64	Adam	0.0010	15	500	42	0.185758	0.167802	0.119443	0.168286
4	45	True	GRU	[128]	True	True	True	64	Adam	0.0010	15	500	42	0.210693	0.168286	0.126479	0.170317
5	45	True	GRU	[128]	True	True	False	64	Adam	0.0010	15	500	42	0.211124	0.166768	0.126110	0.170355
6	45	True	GRU	[128]	False	True	True	64	Adam	0.0010	15	500	42	0.192087	0.167091	0.119226	0.167480
7	45	True	GRU	[128]	False	True	False	64	Adam	0.0010	15	500	42	0.192127	0.167622	0.121619	0.168106
8	45	True	LSTM	[128]	True	True	True	64	Adam	0.0005	15	500	42	0.186193	0.164842	0.117947	0.164842
9	45	True	LSTM	[128]	True	True	False	64	Adam	0.0005	15	500	42	0.187023	0.166161	0.117817	0.166645
10	45	True	GRU	[128]	True	True	True	64	Adam	0.0005	15	500	42	0.184790	0.168742	0.119070	0.168742
11	45	True	GRU	[128]	True	True	False	64	Adam	0.0005	15	500	42	0.185282	0.168058	0.119857	0.168058
12	45	True	LSTM	[128, 64]	True	True	True	64	Adam	0.0010	15	500	42	0.225968	0.165952	0.126575	0.168087
13	45	True	LSTM	[128, 64]	True	True	True	64	Adam	0.0007	15	500	42	0.206036	0.166360	0.125896	0.167128
14	45	True	GRU	[128, 64]	True	True	True	64	Adam	0.0007	15	500	42	0.209139	0.169396	0.125174	0.169700
15	45	True	GRU	[128, 64]	True	True	True	64	Adam	0.0010	15	500	42	0.234814	0.164377	0.127486	0.171608
16	45	True	GRU	[128]	True	False	False	64	Adam	0.0010	15	500	42	0.202189	0.166521	0.121122	0.167907
17	45	True	GRU	[128]	False	False	False	64	Adam	0.0010	15	500	42	0.184717	0.167660	0.119978	0.168476
18	45	True	LSTM	[128]	True	False	False	64	Adam	0.0010	15	500	42	0.196646	0.166493	0.121096	0.168875
19	45	True	LSTM	[128]	False	False	False	64	Adam	0.0010	15	500	42	0.167721	0.158246	0.103026	0.158246
20	45	True	GRU	[128]	True	True	True	128	Adam	0.0010	15	500	42	0.216017	0.168058	0.130770	0.171323
21	45	True	LSTM	[128]	True	True	True	128	Adam	0.0010	15	500	42	0.218196	0.166341	0.127566	0.169937
22	45	True	GRU	[128]	True	True	True	128	Adam	0.0007	15	500	42	0.200017	0.172319	0.124649	0.172424
23	45	True	LSTM	[128]	True	True	True	128	Adam	0.0007	15	500	42	0.202709	0.166341	0.124604	0.168884
24	45	True	GRU	[128]	True	True	True	128	Adam	0.0010	15	200	42	0.242907	0.163219	0.129239	0.171959
25	45	True	LSTM	[128]	True	True	True	128	Adam	0.0010	15	200	42	0.240284	0.164652	0.130540	0.171968

Figure 7: Wyniki badań przeprowadzonych w oparciu o model bazowy (1).

Można zauważyć, że wyniki są do siebie bardzo zbliżone. Dodanie warstwy z mechanizmem uwagi pozwala nieznacznie poprawić jakość modelu. Szczegóły dotyczące Attention layer zostaną zamieszczone w kolejnej sekcji. Można również zaobserwować wzrost wartości f-score dla modeli z warstwą rekurencyjną dwukierunkową i zwiększonym rozmiarem przedostatniej warstwy wępni połączonej.

Dobre wyniki daje zastosowanie stosunkowo wysokiego współczynnika uczenia, bo równego aż 0.001. Na wykresach 9 i 8 przedstawiających wartość accuracy i funkcji kosztu dla zbioru treningowego oraz testowego widać, że wybrany najlepszy model (1.20) zaczyna się przeuczać już około 10 epoki. Zastosowanie mniejszego współczynnika uczenia nie pozwoliło jednak poprawić wyników, ponieważ model utykał w innym optimum lokalnym. Założono, że może to być spowodowane zasumowaniem danych (etykieta była przypisywana w sposób bardzo subiektywny).

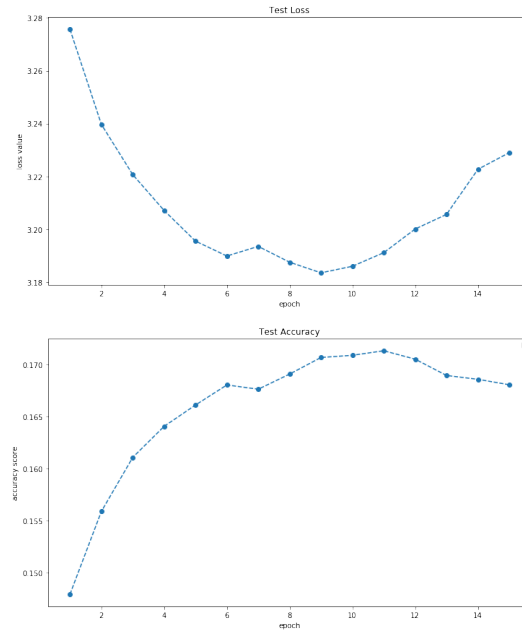


Figure 8: Wartość accuracy i funkcji kosztu dla modelu 1.20 dla zbioru treningowego.

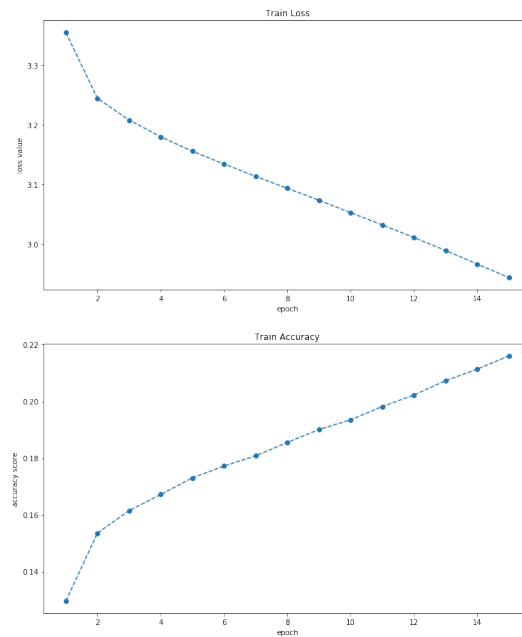


Figure 9: Wartość accuracy i funkcji kosztu dla modelu 1.20 dla zbioru treningowego.

Dla modelu 1.20 zaprezentowano również macierz pomyłek na podstawie której określono jakiego rodzaju błędy popełnia model. Można zauważyć, że podczas gdy dla niektórych emoji model radzi sobie bardzo dobrze (np. 13- urodzinowego rogu, czy 34- ognia), dla innych nie radzi sobie w ogóle (24-29). Należy zwrócić uwagę na fakt, że emoji, dla których wartość f-score jest niska są mało specyficzne, to znaczy mogą być używane w szerokim kontekście i ciężko jednoznacznie zdefiniować ich znaczenie.

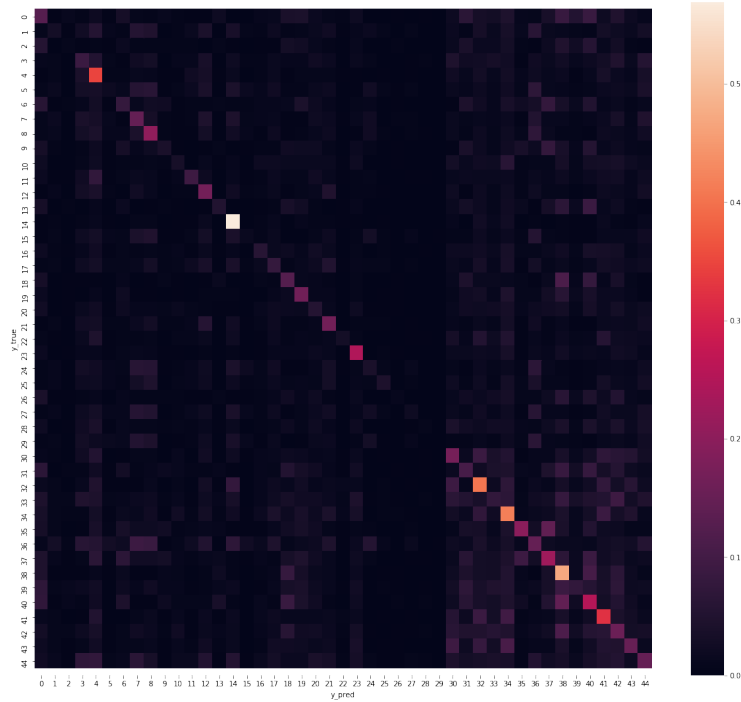


Figure 10: Macierz pomyłek dla modelu 1.20.

2.2 Mechanizm uwagi

Dla przykładowych tweetów wyznaczono wagi mechanizmu uwagi, które reprezentowane są poprzez natężenie koloru (czerwonego lub zielonego). Można zauważyć, że w przypadku poprawnie zaklasyfikowanych tekstów (rysunek 11) mechanizm uwagi bardzo mocno skupia się na słowach charakterystycznych dla danego emoji. Podobnie dla przykładów źle zaklasyfikowanych (rysunek 12) mechanizm skupia się na popularnych słowach specyficznych dla jednego z emoji (niekoniecznie porownego) i nie ujmuje prawdziwego sensu zdania. W przykładach kontrowersyjnych zauważono dokładnie tę samą prawidłowość.

😭
i know it is and i am about to burst into tears just watching it again
🙏
i hope i win your giveaway wish me luck
❤️
missing our little bunny so much my heart hurts

Figure 11: Wagi przypisane do poszczególnych słów dla przykładów poprawnie zaklasyfikowanych.

where did yall come from never had this much likes lmao real talk tho
i had an amazing day
y_true: ❤️, y_pred: 😭

Figure 12: Wagi przypisane do poszczególnych słów dla przykładów niepoprawnie zaklasyfikowanych.

they shudve ask u to give them a lil gym workout
y_true: 😊, y_pred: 🤔
be strong enough to walk away from what s hurting you and be patient
enough to wait for the blessings you deserve
y_true: 😊, y_pred: 🙌

Figure 13: Wagi przypisane do poszczególnych słów dla przykładów kontrowersyjnych.

2.3 Usprawnienie modelu

W ramach usprawnienia modelu zastosowano dodatkową gałąź na wejściu, która przyjmuje sekwencję znaków z tweeta. Taka architektura powinna pozwolić na lepsze radzenie sobie z zanieczyszczonym tekstem z mediów społecznościowych. Schemat modelu został zaprezentowany na rysunku 14.

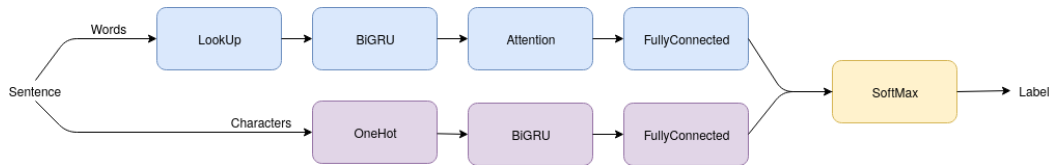


Figure 14: Schemat modelu po wprowadzeniu usprawnień.

Można zauważyć, że takie rozszerzenie pomogło na minimalną poprawę wyników (wartość miary f-score wzrosła dla modelu 2.4), a wykresy wartości funkcji kosztu i accuracy wyglądają analogicznie ak w poprzednim przykładzie.

	optimizer	lr	num_epochs	batch_size	x_len	acc_Train	acc_Test	f_score_Test
0	Adam	0.0010	15	500	42	0.203997	0.170392	0.131404
1	Adam	0.0007	15	500	42	0.211340	0.169936	0.122406
2	Adam	0.0005	15	500	42	0.192921	0.163712	0.118258
3	Adam	0.0007	15	200	42	0.208690	0.170830	0.120909
4	Adam	0.0010	15	200	42	0.226439	0.169690	0.132734

Figure 15: Wyniki badań przeprowadzonych w oparciu o model rozszerzony (2).

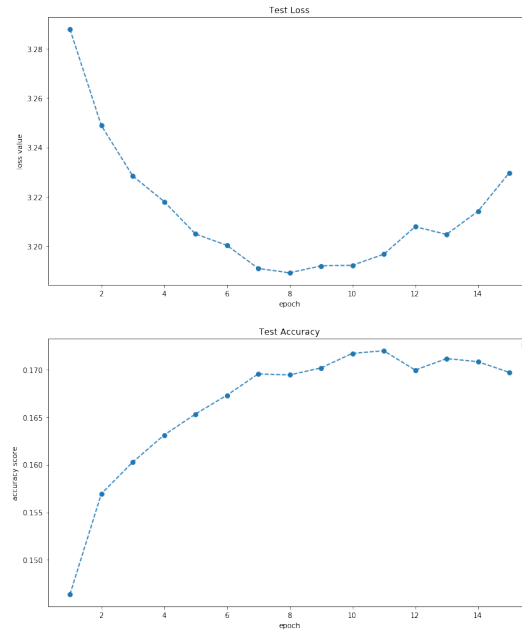


Figure 16: Wartość accuracy i funkcji kosztu dla modelu 1.20 dla zbioru treningowego.

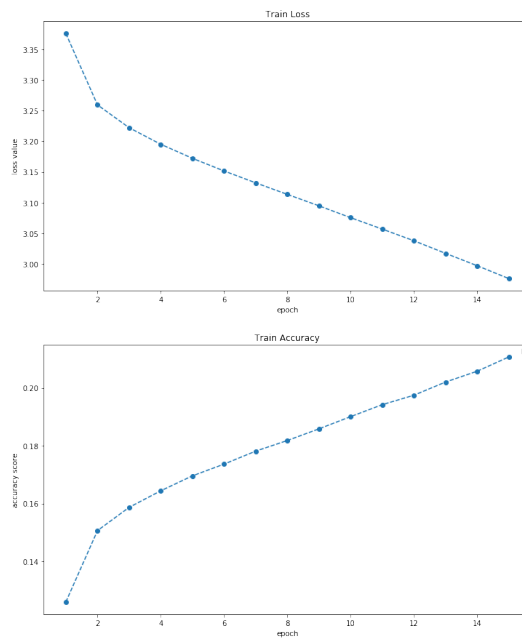


Figure 17: Wartość accuracy i funkcji kosztu dla modelu 1.20 dla zbioru treningowego.

2.4 Klasyfikacja klastrów

Z wykorzystaniem modelu bazowego przeprowadzono także klasyfikację dla grup emoji. Należy jednak wspomnieć o 2 wadach tego podejścia. Po pierwsze przez zastosowanie klasteryzacji zbiór staje się bardzo niezbalansowany. Na przedstawionych macierzach pomyłe widać, że ma to bardzo negatywny wpływ na jakość klasyfikacji, ponieważ grupy większościowe są po prostu częściej predykowane. Dodatkowo należy zwrócić uwagę na fakt, że nie w każdym klastrze emoji są silnie powiązane semantycznie.

Analizując uzyskane wyniki można stwierdzić, że wraz ze wzrostem liczby klastrów jakość klasyfikatora się pogarsza. Oczywiście wynika to również z prostego faktu, że wzrasta liczba klas. Można było jednak przypuszczać, że dla większej liczby klastrów emoji w grupie będą silniej powiązane semantycznie. Niestety silne niebalansowanie zbioru wyraźnie utrudnia zadanie.

	num_labels	init_weights	rnn_type	rnns_units	bidirectional	attention	mask	dense_units	optimizer	lr	num_epochs	batch_size	x_len	clustering_id	acc_Train	acc_Test	f_score_Test	max_acc_Test
0	7	True	LSTM	[128]	True	True	True	128	Adam	0.001	15	500	42	0	0.548771	0.487388	0.399312	0.491820
1	8	True	LSTM	[128]	True	True	True	128	Adam	0.001	15	500	42	1	0.508240	0.433991	0.332877	0.444278
2	9	True	LSTM	[128]	True	True	True	128	Adam	0.001	15	500	42	2	0.467143	0.398472	0.309019	0.410448
3	10	True	LSTM	[128]	True	True	True	128	Adam	0.001	15	500	42	3	0.471812	0.408322	0.319187	0.414462

Figure 18: Wyniki badań przeprowadzonych w oparciu o model podstawowy (1.20) dla grup emoji.

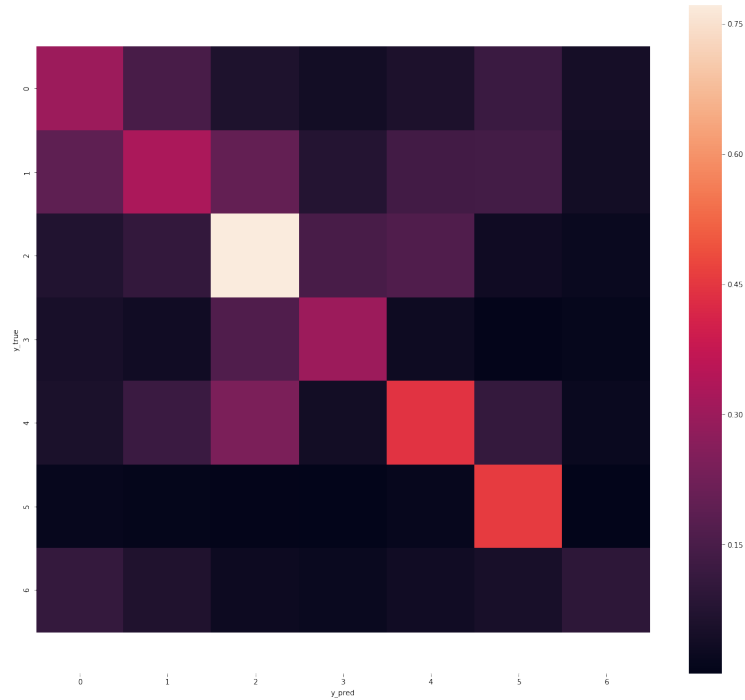


Figure 19: Macierz pomyłek dla modelu 1.20 dla liczby grup równej 7.

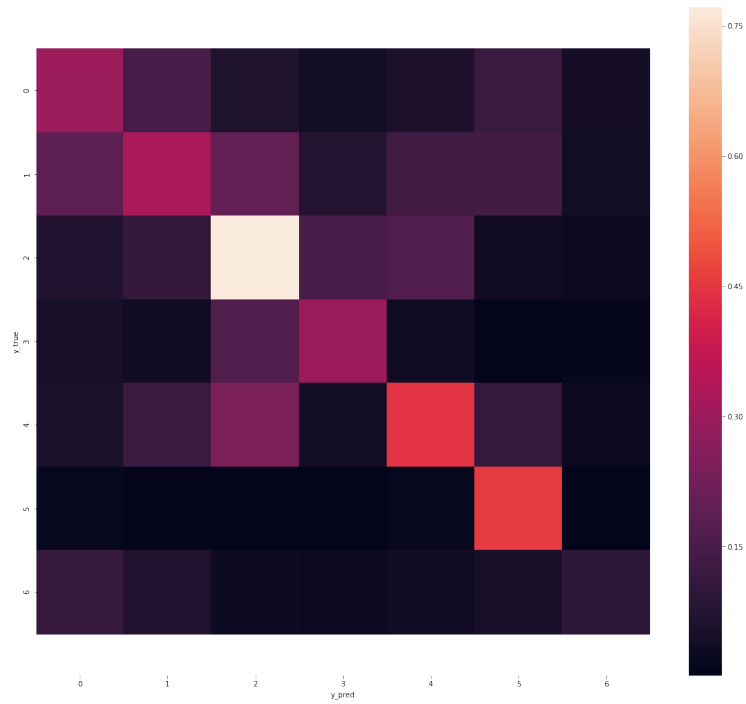


Figure 20: Macierz pomyłek dla modelu 1.20 dla liczby grup równej 7.

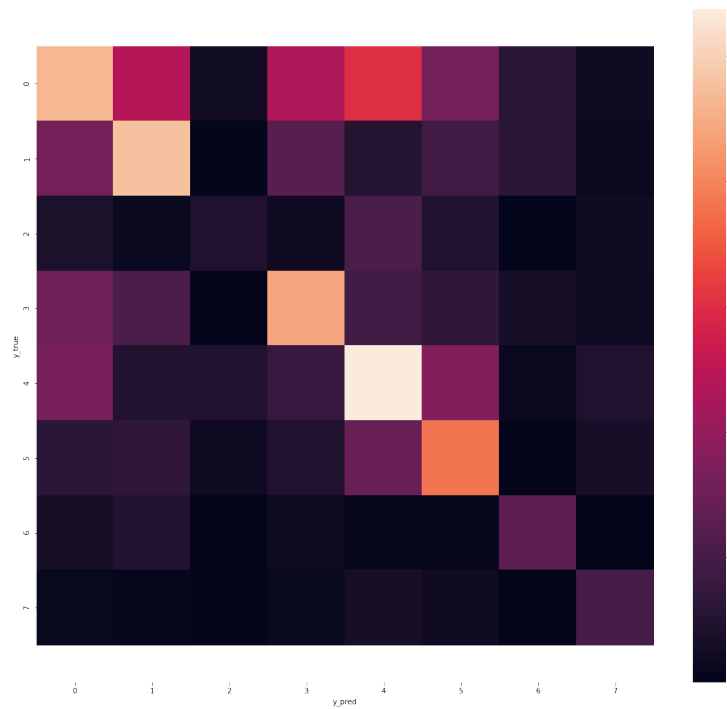


Figure 21: Macierz pomyłek dla modelu 1.20 dla liczby grup równej 8.

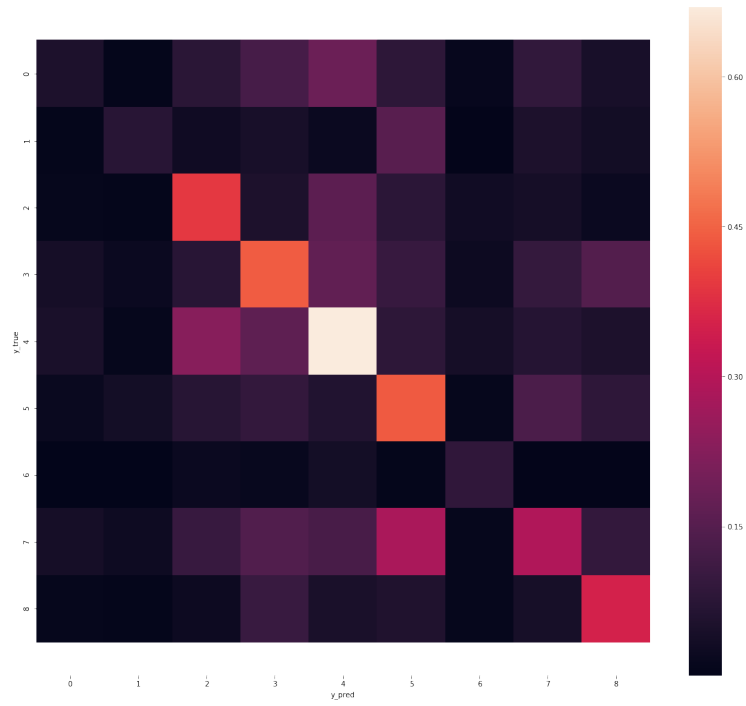


Figure 22: Macierz pomyłek dla modelu 1.20 dla liczby grup równej 9.

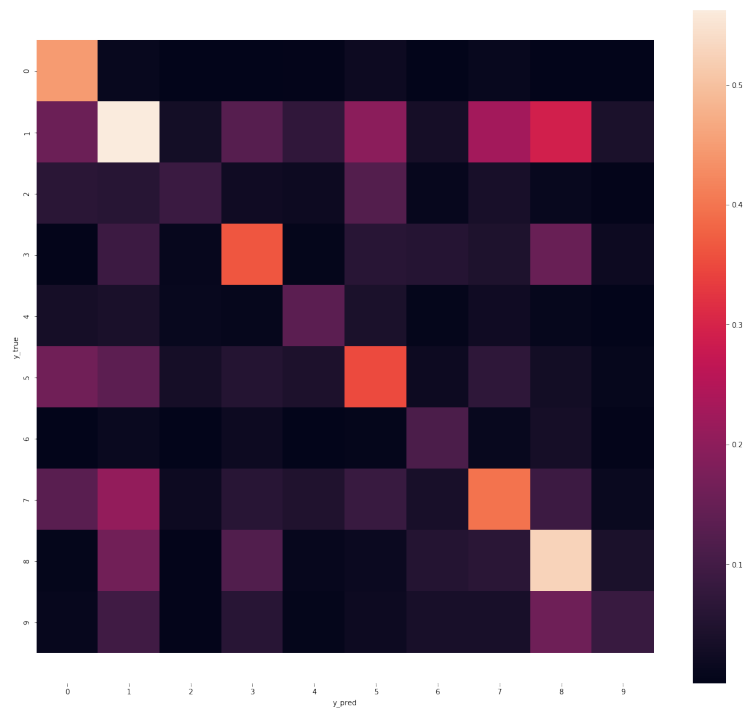


Figure 23: Macierz pomyłek dla modelu 1.20 dla liczby grup równej 10.