

# Projekt MSI 2019L

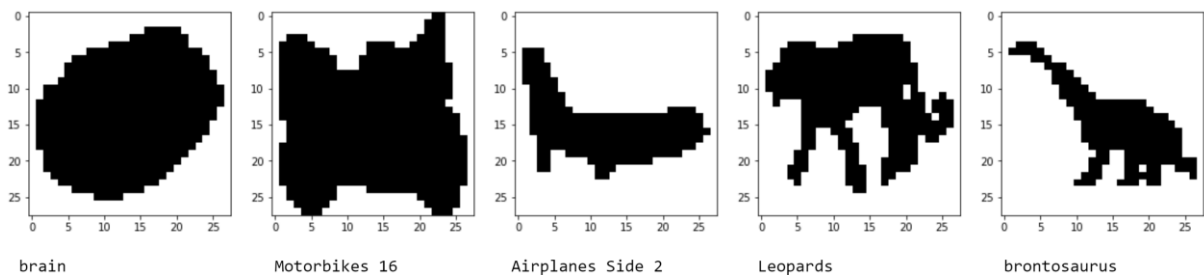
## Klasyfikacji obrazów z wykorzystaniem głębokich sieci przekonań

### Spis treści

1.	Opis projektu .....	2
2.	Wstęp teoretyczny.....	2
2.1.	Restricted Boltzman Machine(RBM) .....	2
2.2.	Deep Belief Network(DBN) .....	3
2.3.	Complementary Priors and Contrastive Divergence (CD) .....	4
3.	Prezentacja wyników.....	5
3.1.	Informacje o danych wejściowych.....	5
3.2.	Informacje o przeprowadzonych testach .....	5
4.	Wnioski.....	6
5.	Bibliografia.....	7

## 1. Opis projektu

Projekt ma na celu praktyczne zaznajomienie się z metodami klasyfikacji wykorzystującymi głębokie sieci przekonań. Zadanie klasyfikacji dotyczy bazy obrazów CALTECH-101 przedstawiająca kontury różnych obiektów. Badaniu zostały poddane obrazy o rozmiarach 16x16 oraz 28x28 pikseli. Dane zostały podzielone na trenujące, walidujące i testujące oraz obejmują klasy 1-33 ze 101 dostępnych. Architektura sieci oraz algorytm uczenia zostały zaproponowane przez Hintona w publikacji 'A fast Learning Algorithm for Deep Belief Nets'. Do rozwiązania zagadnienia wykorzystano między innymi technologie python oraz tensorflow.



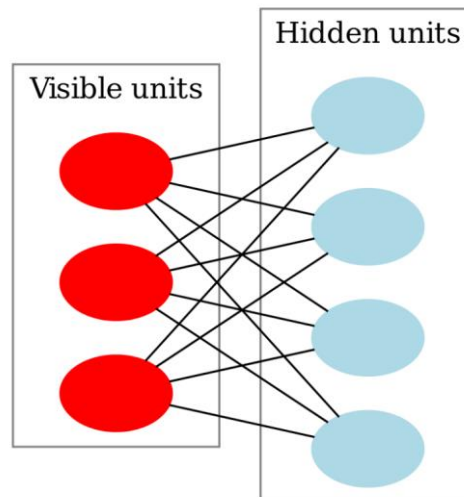
Rys 1. Przykładowe kontury z bazy danych CALTECH-101

## 2. Wstęp teoretyczny

W tym rozdziale opisane zostały najciekawsze zagadnienia teoretyczne zaproponowane przez Hintona dotyczące efektywnego funkcjonowania głębokich sieci przekonań.

### 2.1. Restricted Boltzman Machine(RBM)

Maszyna Boltzmanna to rodzaj stochastycznej, rekurencyjnej sieci neuronowej zbudowanej z binarnych neuronów. Posiada dwie warstwy – neurony widzialne oraz neurony ukryte. Zadaniem neuronów ukrytych jest modelowanie zależności określonych przez neurony widzialne. Budowa maszyny Boltzmana znacząco utrudnia uczenie i sprawia problemy ze skalowalnością. BM wykorzystują do uczenia algorytm rozbieżności kontrastowej. W związku z tym Paul Smolensky zaproponował ograniczone maszyny Boltzmanna, w których połączenia między neuronami zachodzą wyłącznie pomiędzy kolejnymi warstwami (wierzchołki każdej warstwy nie są wzajemnie połączone). Do ich propagacji przyczynił się Hinton, który pracował nad algorytmami ich uczenia.



Rys 2. przykładowa struktura RBM

W RBM(rys 2.) neurony ukryte są niezależne pod warunkiem neuronów widzialnych, zatem struktura niezależności i próbkowanie jest znacznie uproszczone. RBM jest modelem opartym na energii. Rozkłady prawdopodobieństwa w warstwach ukrytych oraz widocznych można zapisać jako

$$P(v, h) = \frac{1}{Z} e^{-E(v, h)} \quad (1)$$

gdzie:

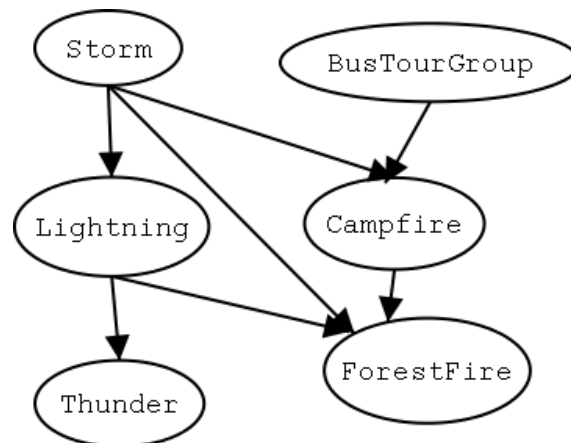
$$Z = \sum_{v, h} e^{-E(v, h)} \quad (2)$$

Połączenie RBM w strukturę gdzie warstwa ukryta jednej maszyny jest warstwą widzialną dla maszyny drugiej tworzy głęboką sieć przekonań DBN.

## 2.2. Deep Belief Network(DBN)

DBN to rodzaj sieci neuronowej, zaproponowany przez Goffrey'a Hintona, który można opisać jako połączenie dwóch innych typów sieci neuronowych – sieci przekonań („Belief Network”) oraz ograniczonej maszyny Boltzmanna („Restricted Boltzmann Machine”). Sieć przekonań to graf acykliczny o probabilistycznych wierzchołkach . Składa się z neuronów widzialnych oraz ukrytych, które przyjmują stan 0 lub 1 z prawdopodobieństwem zależnym od obciążenia oraz ważonych wejść z innych neuronów. Wiąże się z tym problem uczenia takiej sieci – powstawania efektu „explaining away”. Przykładowo, dwa niezależne neurony ukryte mogą stać się zależne jeżeli istnieje zjawisko na które obydwa mają wpływ. Co więcej, dla danego rozkładu prawdopodobieństwa istnieje wiele możliwych konfiguracji wielowarstwowych sieci przekonań. Dlatego Hinton proponuje uczenie sieci warstwa po warstwie, nakładając

pewne ograniczenie w strukturze jej budowy – wykorzystuje strukturę ograniczonej maszyny Boltzmann.



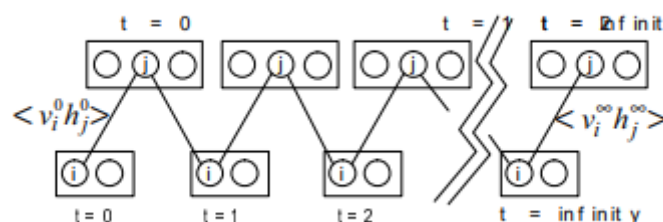
Rys 3. przykład prostej sieci przekonań

### 2.3. Complementary Priors and Contrastive Divergence (CD)

W celu umożliwienia efektywnego trenowania sieci przekonań, w swojej pracy Geoffrey Hinton przedstawia wykorzystanie metody rozbieżności kontrastowej („complementary priors”) do eliminacji efektu nazwanego „explaining away phenomenon”, który polega na trudności w obliczaniu kolejnych rozkładów prawdopodobieństwa dla warstw ukrytych w sieciach przekonań. Metoda ta upraszcza obliczenia tych rozkładów, co pozwala na stworzenie szybkiego, zachłannego algorytmu uczącego wagi w głębokich sieciach przekonań, warstwa po warstwie. Polega na tym, że połączenie pomiędzy dwiema najwyższymi warstwami nie jest kierunkowe, co symuluje efekt nieskończonej ilości warstw ukrytych.

Aby przyspieszyć proces uczenia RBM używany jest algorytm rozbieżności kontrastowej, którego idea polega na równoczesnym aktualizowaniu wszystkich ukrytych neuronów zaczynając od tych widocznych, rekonstrukcji widocznych neuronów przy pomocy ukrytych. Na końcu aktualizowane są ponownie neurony ukryte.

$$\frac{\partial \ln P}{\partial w_{ij}} = \langle v_i h_j \rangle_0 - \langle v_i h_j \rangle_\infty \quad (3)$$



Rys 4. Przedstawienie algorytmu rozbieżności kontrastowej dla RB

### 3. Przedstawienie wyników

#### 3.1. Informacje o danych wejściowych

Do odczytania danych obrazów zapisanych w formacie danych oprogramowania matlab .mat wykorzystano `loadmat` z `scikit.io`. Klasa każdego z wczytywanych obrazów jest znana. Dokonana klasyfikacja jest typu 'supervised' czyli sieć sama nie wyodrębnia klas co jest możliwe przy głębokich sieciach przekonań.

Całkowita ilość poddanych analizie obrazów: 4456  
Całkowita ilość danych trenujących: 1587  
Całkowita ilość danych walidujących: 1428  
Całkowita ilość danych testujących: 1441

CLASS NAME	ID	TRAIN	VAL	TEST
Airplanes Side 2	1	100	349	349
Faces 2	2	100	167	168
Faces 3	3	100	167	168
Leopards	4	100	50	50
Motorbikes 16	5	100	349	349
accordion	6	33	11	11
anchor	7	26	8	8
ant	8	26	8	8
barrel	9	29	9	9
bass	10	33	10	11
beaver	11	28	9	9

Rys 5. Przedstawienie niektórych klas oraz ich liczebność w każdym z podzbiorów

#### 3.2. Informacje o przeprowadzonych testach

W przeprowadzonej analizie główne rozpatrywane parametry to liczba powłok ukrytych 'hidden structure layer', 'batch size' czyli ilość próbek podawana propagacji oraz liczba epok wpływająca na ilość powtórzeń trenowania na danych testujących. Zaletą przeprowadzenia podobnych testów na obrazach o rozmiarze 16x16 oraz 28x28 pikseli jest możliwość oszacowania wpływu rozmiaru obrazu na skuteczność klasyfikacji za pomocą głębokiej sieci przekonań.

28x28

Testy badające wpływ parametru `batch_size` na dokładność klasyfikacji:

Hidden structure	Num of epochs	Batch size	Accuracy
256 x 256	10	16	0.6884108258
256 x 256	10	32	0.7265787647
256 x 256	10	48	0.6863289382
256 x 256	10	64	0.2421929216

Rys 6. Tabela wyświetlona w programie PyCharm, testy `batch_size` 28x28

Testy badające wpływ rozmiaru struktury ukrytej na dokładność klasyfikacji:

Hidden structure			Num of epochs	Batch size	Accuracy
28	x	28	10	32	0.5850104094
128	x	128	10	32	0.5634975711
256	x	256	10	32	0.7265787647
512	x	512	10	32	0.6835530881
784	x	784	10	32	0.7328244275

Rys 7. Przedstawienie wyników testów dla różnych wielkości struktury ukrytej, 28x28

Testy badające wpływ ilości epok na dokładność klasyfikacji:

Hidden structure			Num of epochs	Batch size	Accuracy
256	x	256	5	32	0.6904927134
256	x	256	10	32	0.7265787647
256	x	256	15	32	0.6592643997
256	x	256	20	32	0.6294240111
256	x	256	30	32	0.7009021513
256	x	256	40	32	0.6557945871

Rysunek 8. Przedstawienie wyników testów dla różnych ilości epok, 28x28

## 4. Wnioski

Największa uzyskana skuteczność klasyfikowania obrazów rozdzielczości 28x28 wynosi ~73% , co dla zbioru 1441 obrazów testujących oznacza 1052 poprawnie sklasyfikowane obrazy. Parametry najlepszego wyniku:

Rozmiar struktury ukrytej: 784x784,  
Liczba epok: 10  
Batch size: 32

Przeprowadzając testy badające parametry w sposób odizolowany, można wywnioskować, że najlepsze wyniki dla rozpatrywanych danych można otrzymać gdy ilość warstw ukrytych jest w okolicy 256 bądź 784, liczba epok jest w okolicy 10 oraz ilość próbek waha się w granicy 32. Należy jednak przeprowadzić dodatkowe testy rozpatrując większą ilość kombinacji wartości parametrów.

16x16

Hidden structure			Num of epochs	Batch size	Accuracy
10	x	10	10	32	0.4112734864
16	x	16	10	32	0.3437717467
24	x	24	10	32	0.6165622825
32	x	32	10	32	0.7070285317
40	x	40	10	32	0.7139874739
48	x	48	10	32	0.6979819068
64	x	64	10	32	0.6931106472

Rys 9. Przedstawienie wyników dla różnej ilości warstw ukrytych, 16x16

Hidden structure			Num of epochs	Batch size	Accuracy
32	x	32	1	32	0.7292971468
32	x	32	2	32	0.6576200418
32	x	32	5	32	0.6833681280
32	x	32	8	32	0.6228253305
32	x	32	10	32	0.7070285317
32	x	32	12	32	0.6708420320

Rys 10. Przedstawienie wyników dla różnej ilości epok, 16x16

Obrazy o rozdzielczości 16x16 pikseli zawierają 256 pikseli, jest to niemalże 3-krotnie mniej niż obrazy o rozdzielczości 28x28, które mają ich 784. W rezultacie, zgodnie z oczekiwaniami dla danych o mniejszej ilości nodów widzialnych(pikseli) można uzyskać wysoką skuteczność wykorzystując mniejszą strukturę warstw ukrytych. Zaskakujące jest to, że dla liczby epok w ilości 1 uzyskano najwyższą efektywność na poziomie ~73%. Uzyskana efektywność klasyfikacji wskazuje na istnienie wykładniczej zależności pomiędzy rozmiarem danych wejściowych a optymalnym rozmiarem struktury warstw ukrytych.

Wyniki testów sugerują słuszność dalszych prowadzenia prac, należałoby przeprowadzić więcej testów uwzględniając większą ilość kombinacji różnych wartości parametrów. Należałoby również porównać uzyskane wyniki z wyniki z pozostałych grup klas oraz przeprowadzić całościowe testy na zbiorze wszystkich 101 klas.

## 5. Bibliografia

1. <https://people.cs.umass.edu/~marlin/data.shtml> - źródło danych testowych
2. Hinton, 'A fast Learning Algorithm for Deep Belief Nets'  
<https://www.cs.toronto.edu/~hinton/absps/fastnc.pdf>
3. HussamHebbo, Jae Won Kim, 'Classification with deep belief network' -  
[https://www.ki.tu-berlin.de/fileadmin/fg135/publikationen/Hebbo\\_2013\\_CDB.pdf](https://www.ki.tu-berlin.de/fileadmin/fg135/publikationen/Hebbo_2013_CDB.pdf)
4. <https://github.com/albertbup/deep-belief-network> - Implementacja DBN, RBM  
Tensorflow