



**enigma**  
pattern



Deep Learning with pythons and cobras



# Znajdowanie węży na obrazach

---

Piotr Bara  
Patryk Seweryn

# Skąd pomysł?

---

# Gdzie jest python nad Wisłą? Nowe ustalenia!

Python pozował w erotycznej sesji na plaży nad Wisłą. Czy to jego szuka policja?



zwierzęta w Warszawie  
Python się ukrył i czeka na cieplejsze dni. Strażacy przerwali poszukiwania



# Cel projektu

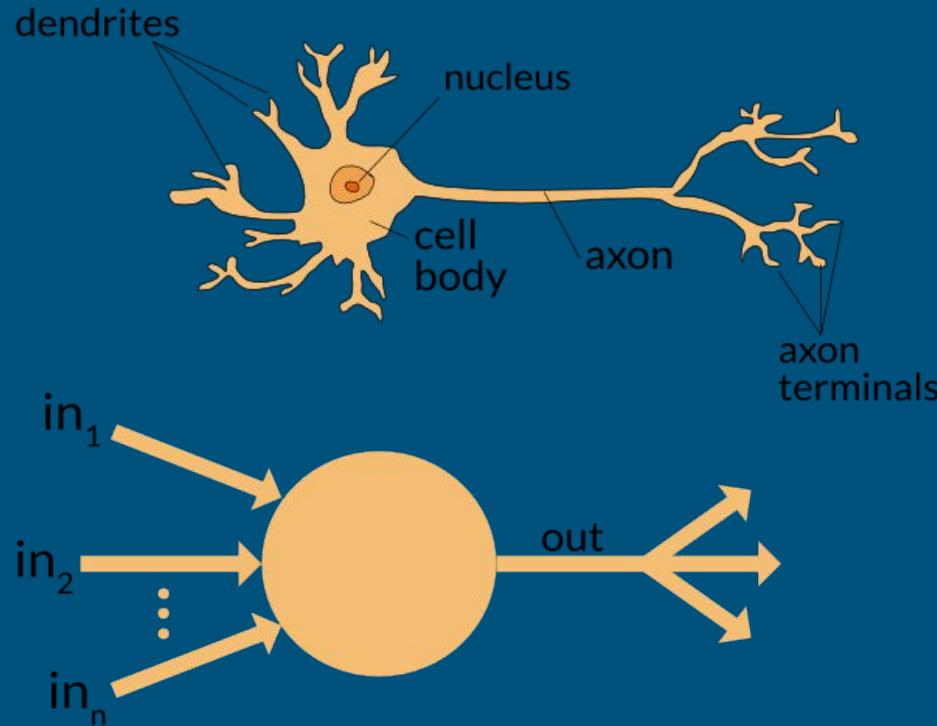
---

Stworzenie inteligentnego modelu, który wskaże gdzie na obrazku znajduje się wąż.

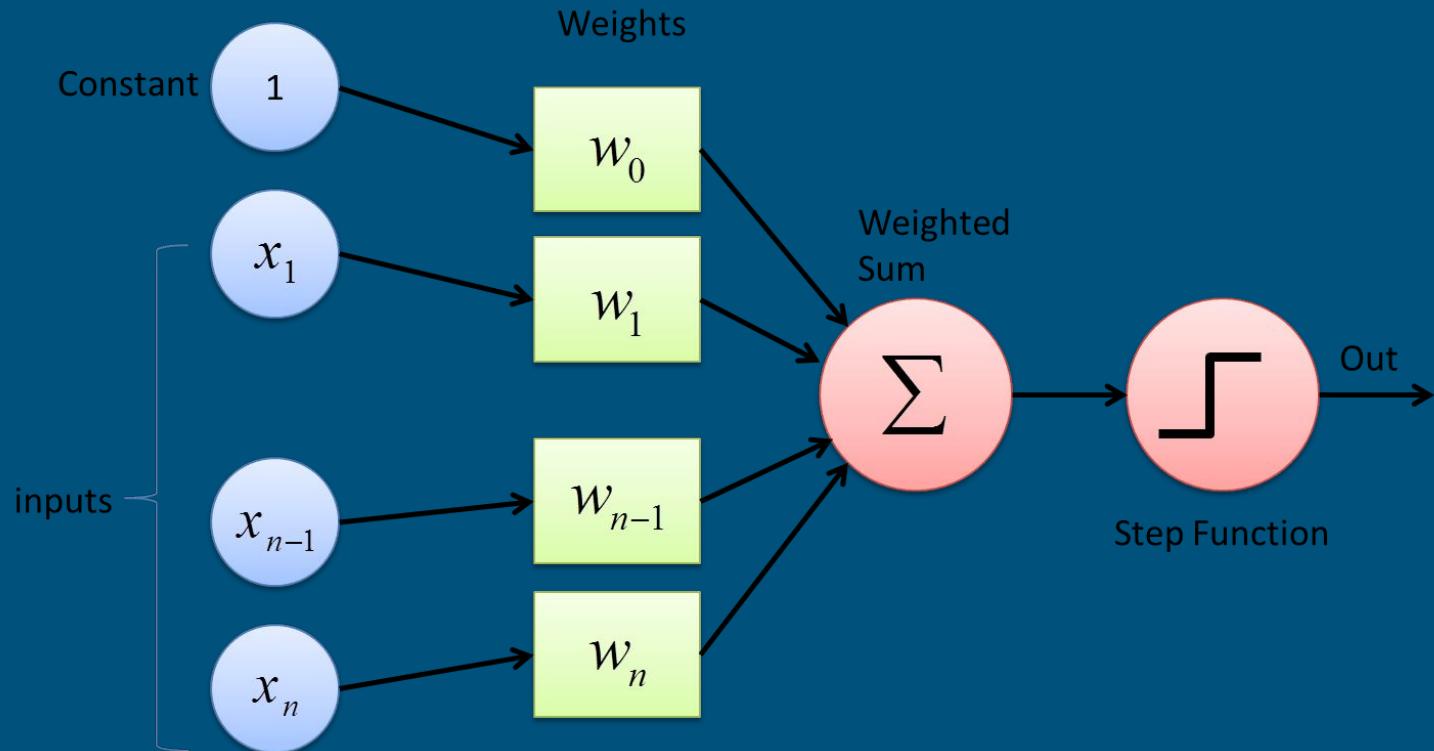
Krótkie wyjaśnienie jak  
działają sztuczne sieci  
neuronowe...

---

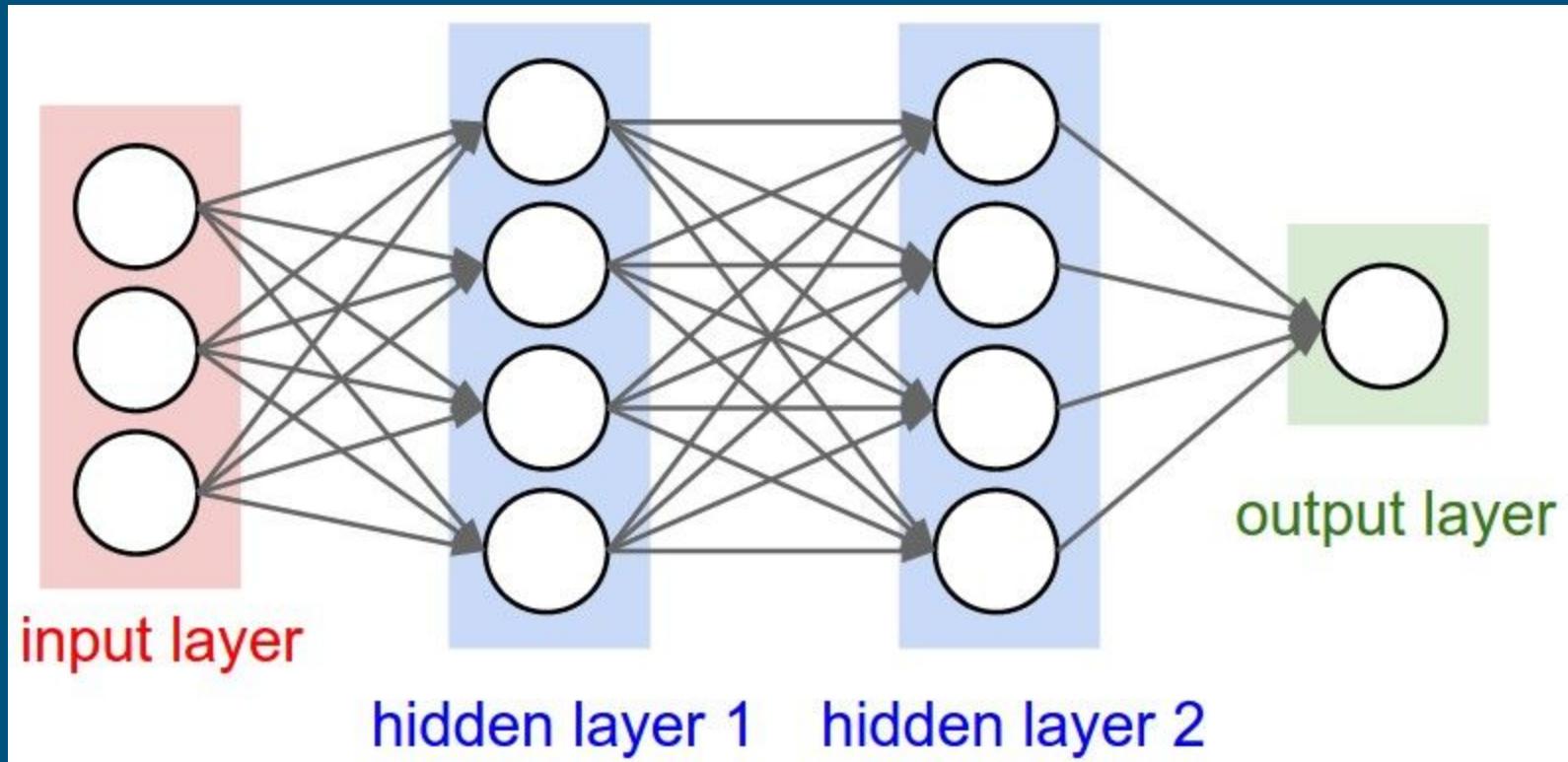
# Sposób funkcjonowania neuronów



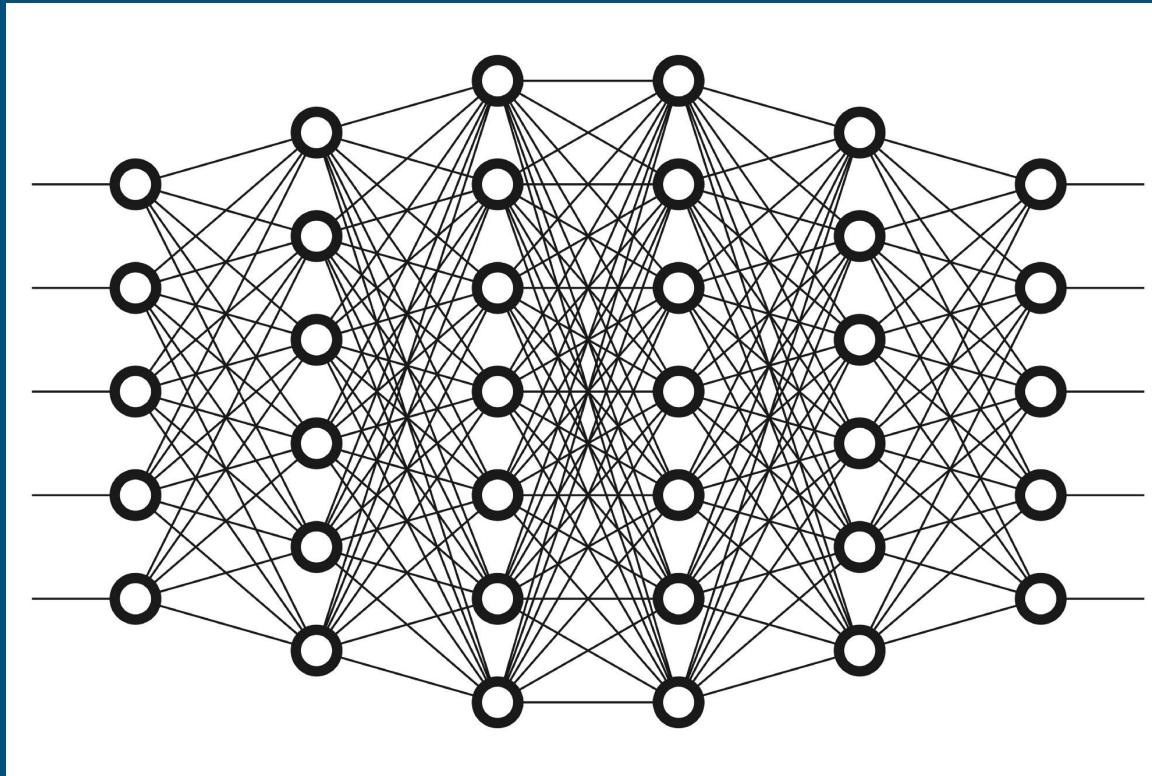
# Perceptron - najprostsza sieć neuronowa



# Sieci neuronowe z warstwami ukrytymi



# Głębokie sieci neuronowe



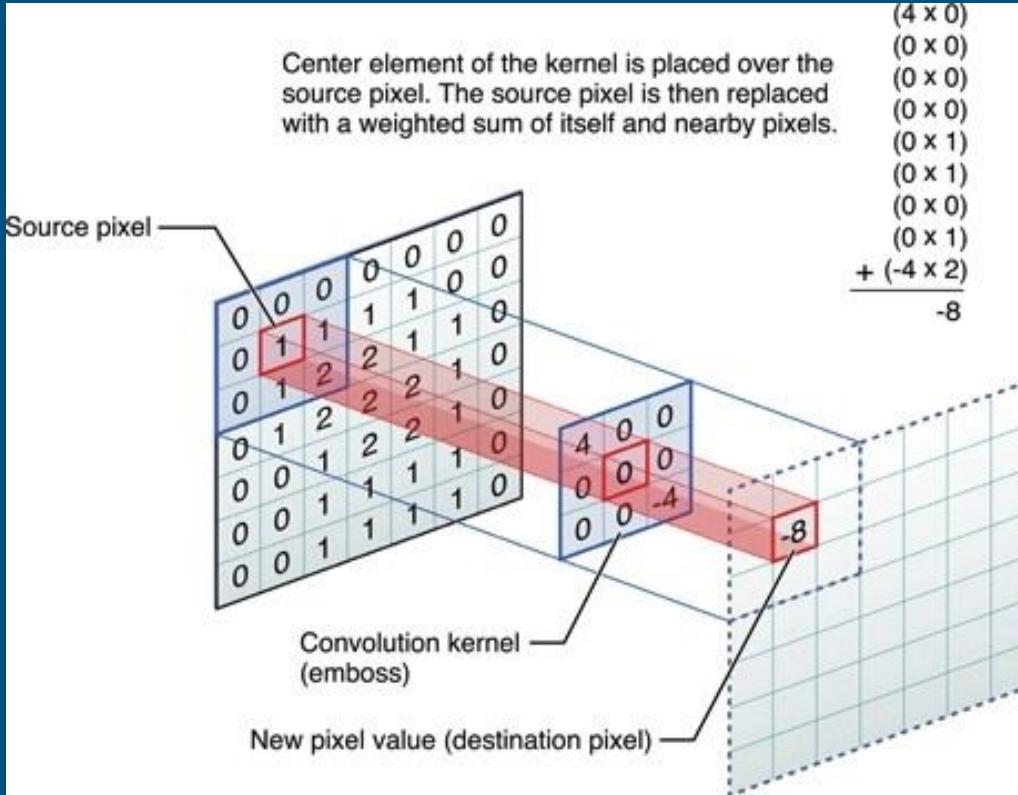


# KONWOLUCJE?

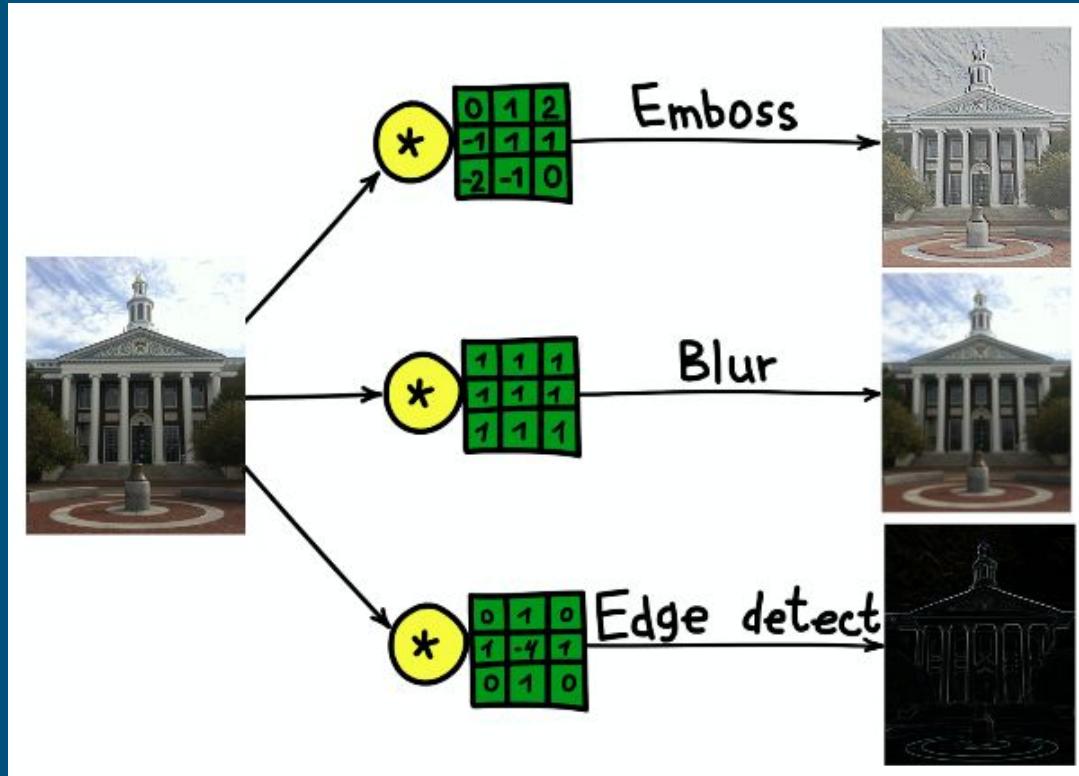


# A KOMU TO POTRZEBNE?

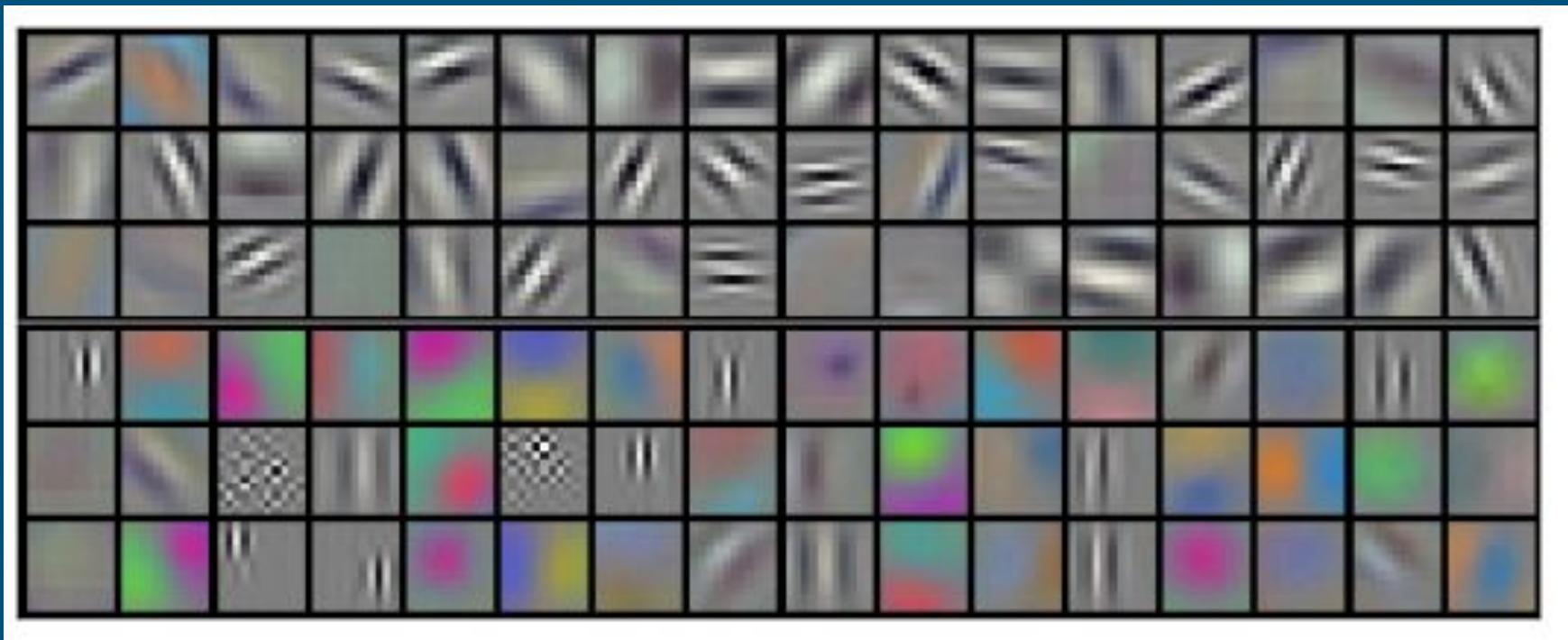
# Konwolucje



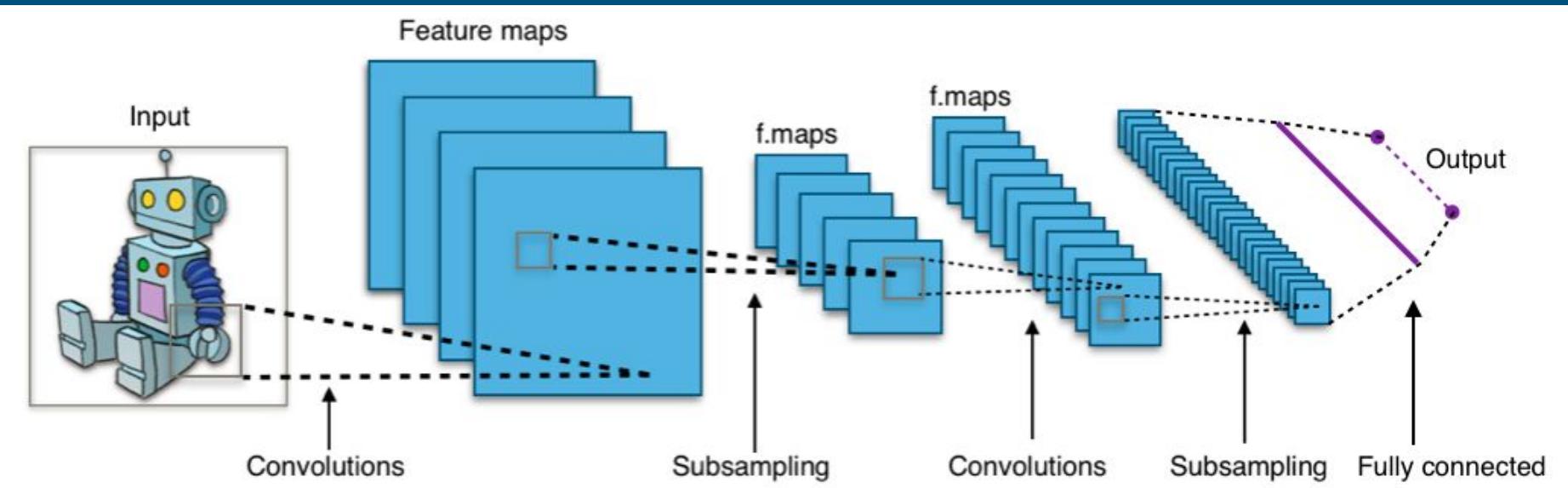
# Konwolucje - praktyczne wyjaśnienie



Tak wyglądają nauczone filtry konwolucyjne:



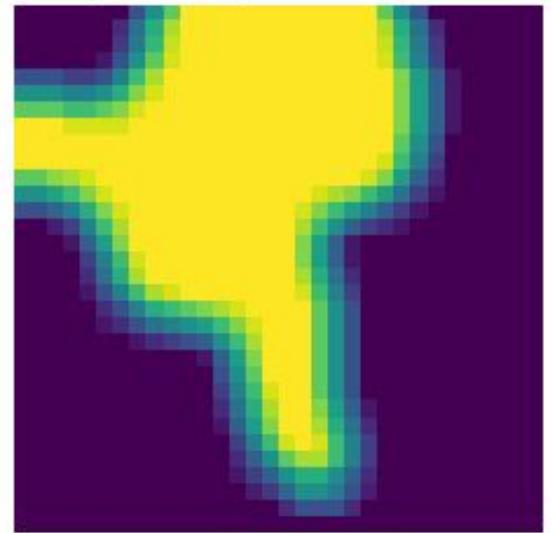
# Sieci konwolucyjne



Wejście

---

Wyjście



# Przygotowanie zbioru danych

---

# Przygotowanie zbioru danych

---

Skąd wziąć zdjęcia węży?

# Zbiór, który wybraliśmy:

---

<https://github.com/arjun921/Indian-Snakes-Dataset>



# Dostajecie taki obrazek:

---



# Kolejne pytanie

---

Jak powiedzieć sieci neuronowej gdzie znajduje się wąż na obrazku?

# Szparowanie





No dobra, ale ile takich  
obrazków trzeba  
przygotować?

---

# Możliwe opcje:

---

- Siedzieć i szparować setki obrazków własnoręcznie
- Wynająć kogoś, kto to zrobi za nas
- Albo... zmienić model tak, żeby zwiększyć nasze dane (liczbę przykładów), pozostając przy tych zdjęciach, które mamy

# Pytanie za 100 punktów:

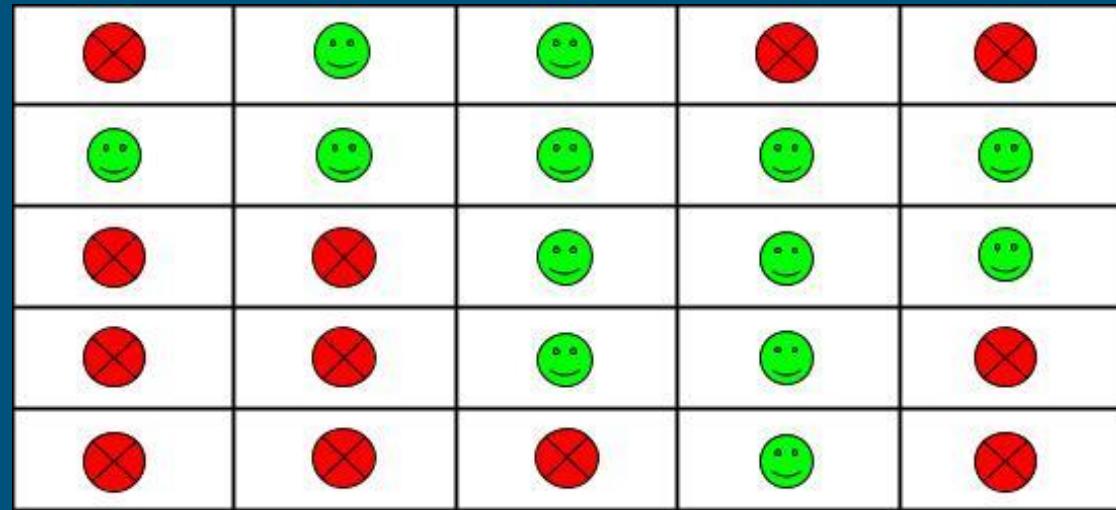
---

Co zrobić żeby mieć więcej danych przy tej samej ilości obrazków?

Podpowiedź: Możemy zmienić wyjście naszego modelu

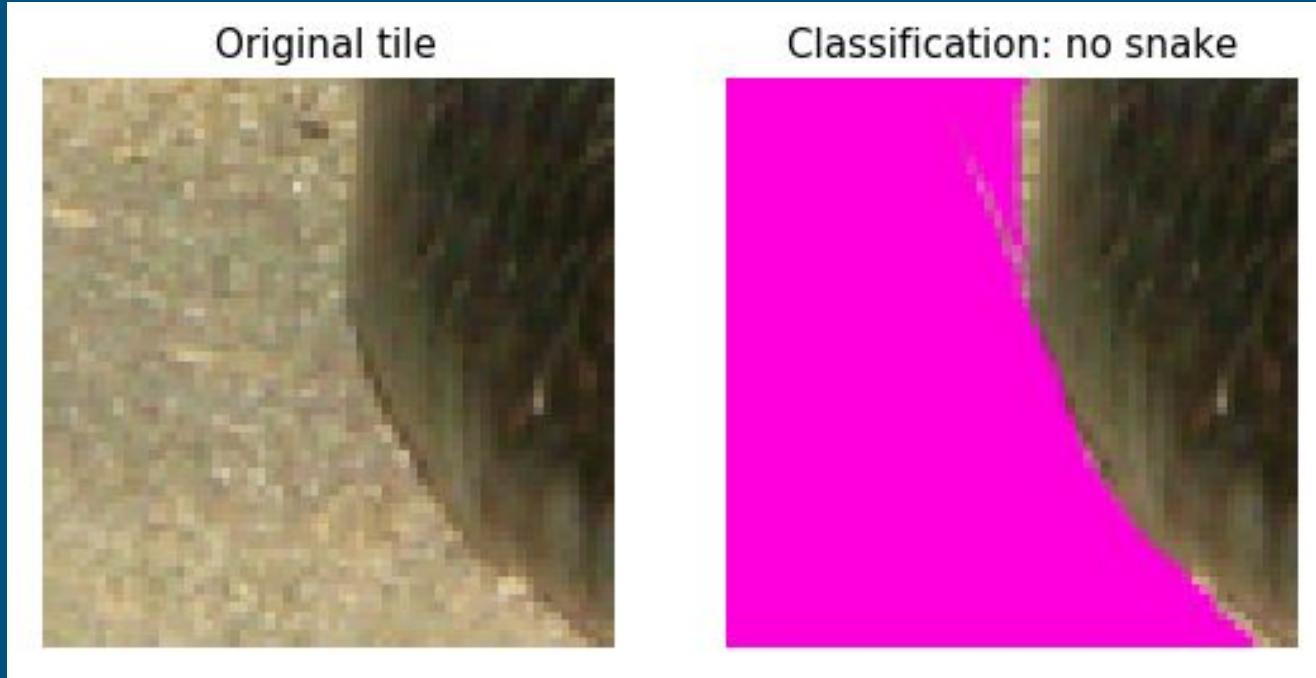
# Odpowiedź:

---



# Jak to wygląda w praktyce:

---



# Jak to wygląda w praktyce:

---

Original tile



Classification: cobra



# Wejście

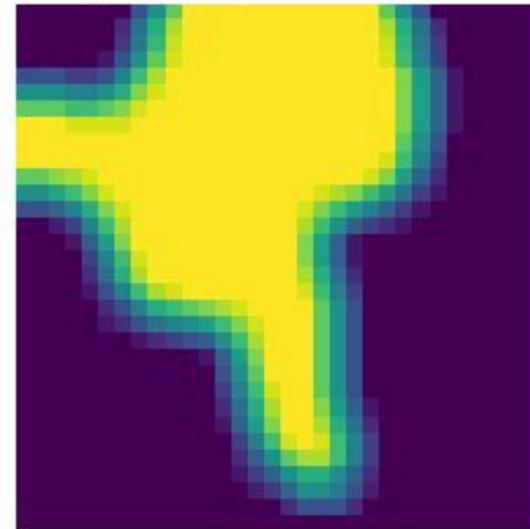
---

# Wyjście

Original image



Snake position prediction with tiles

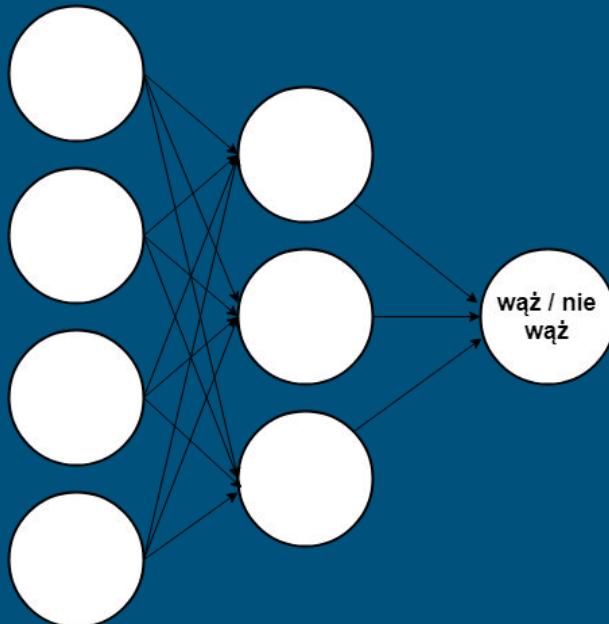


# Budowanie modelu sieci neuronowej

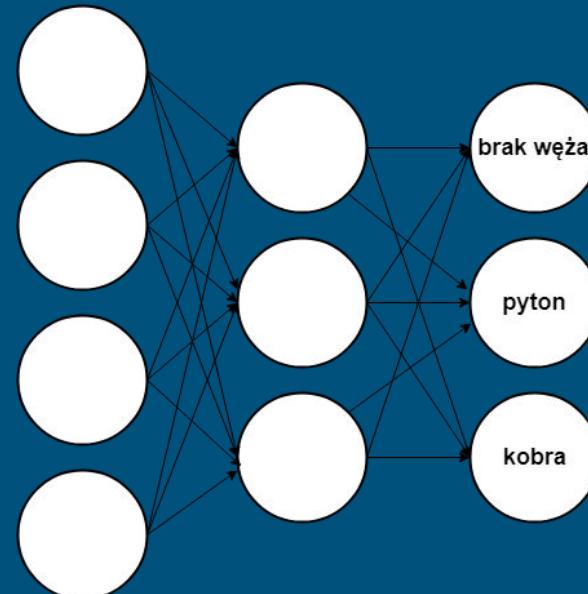
---

## 1. Model do detekcji węży

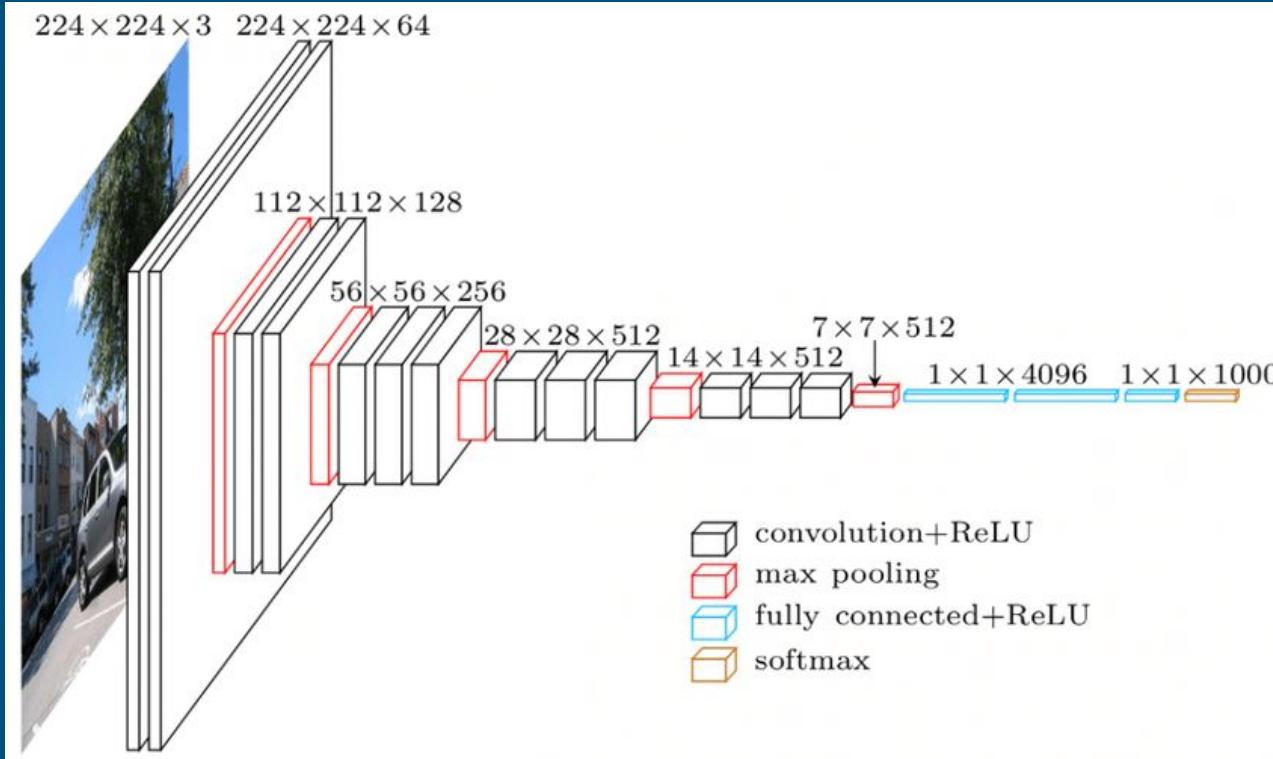
---



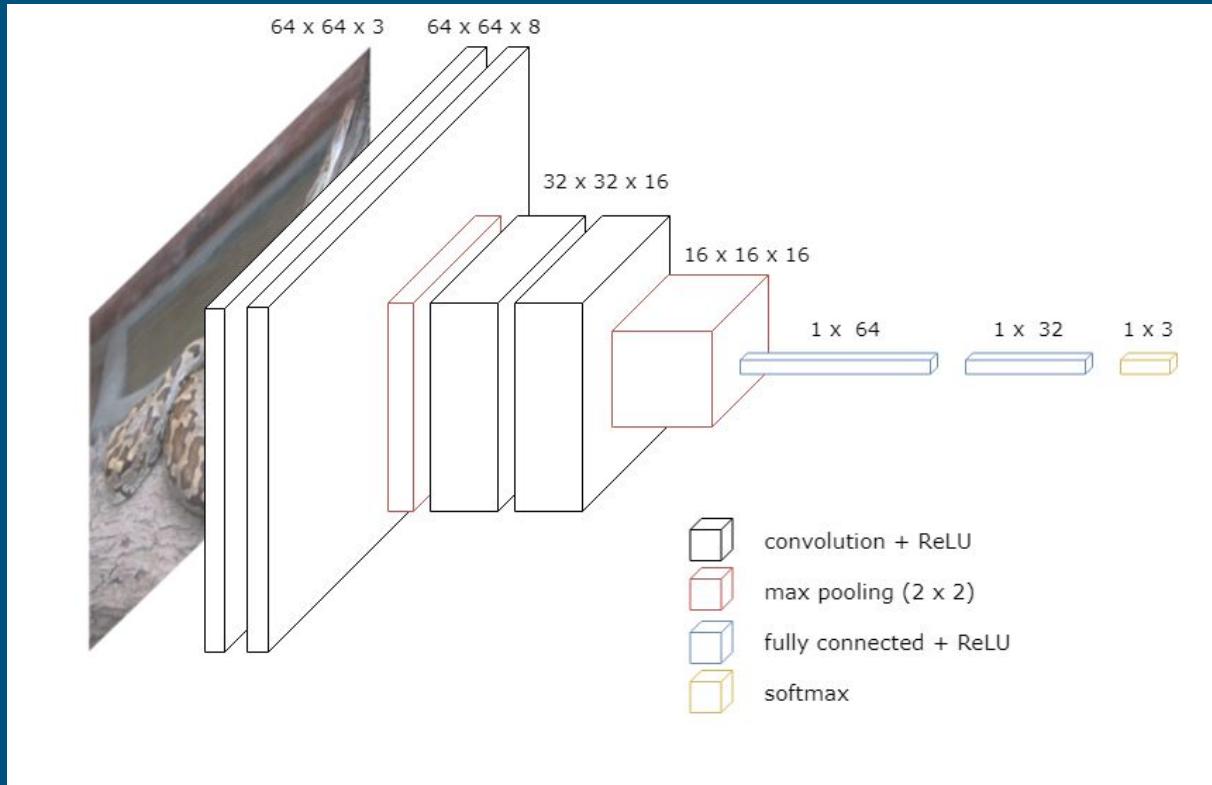
## 2. Model do detekcji i klasyfikacji węży



# Z czego mogliśmy skorzystać - sieć VGG16:

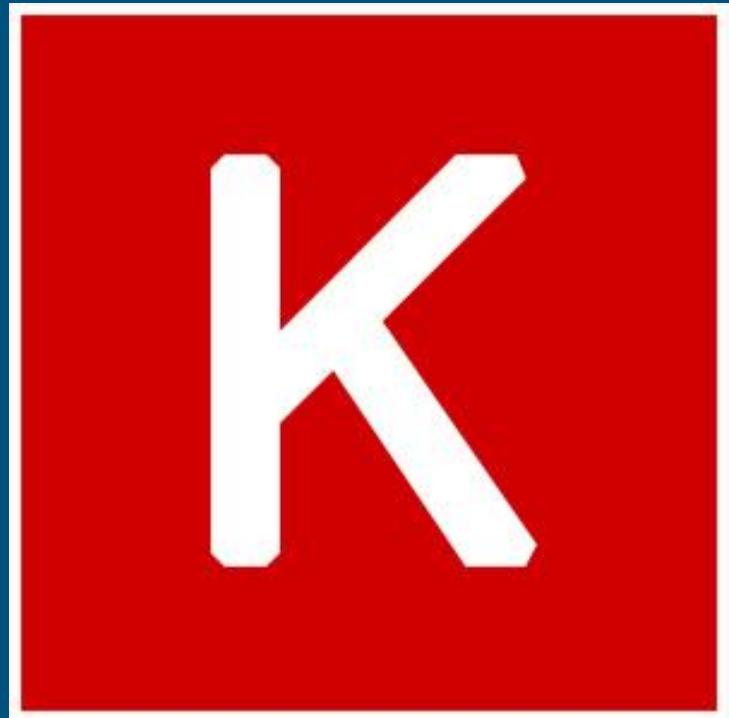


# Nasz model



# Użyte technologie - Python + Keras

---



# Przygotowanie do treningu

---

- Przygotowaliśmy po 22 zdjęcia na każdy gatunek węża (kobra i python)
- Zdjęcia podzieliliśmy na dwa zbiory - treningowy i walidacyjny
- 80%/20%: 17 zdjęć poszło do zbioru treningowego, 5 do zbioru walidacyjnego

# Jak wygląda trening modelu?

---

# Po co podział na zbiory treningowe i walidacyjne?

---



**SPOKO, POCZEKAM**

# Pierwsze wyniki:

---

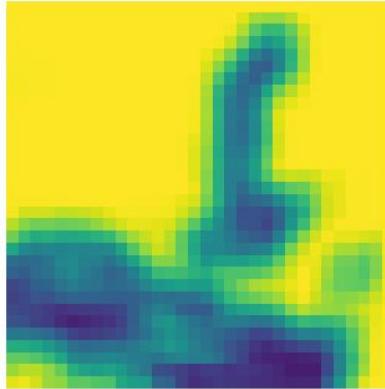
# Pierwotny model:

---

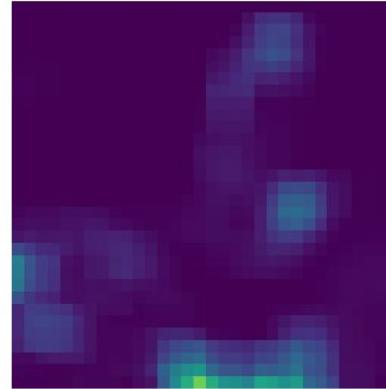
king\_cobra



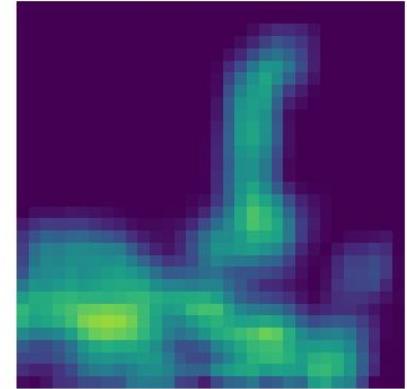
no snake



cobra



python



# Co zmienić?

---

# USUWAC ZDjecia z DATASETU



## BO MODEL GORZEJ DZIAŁA

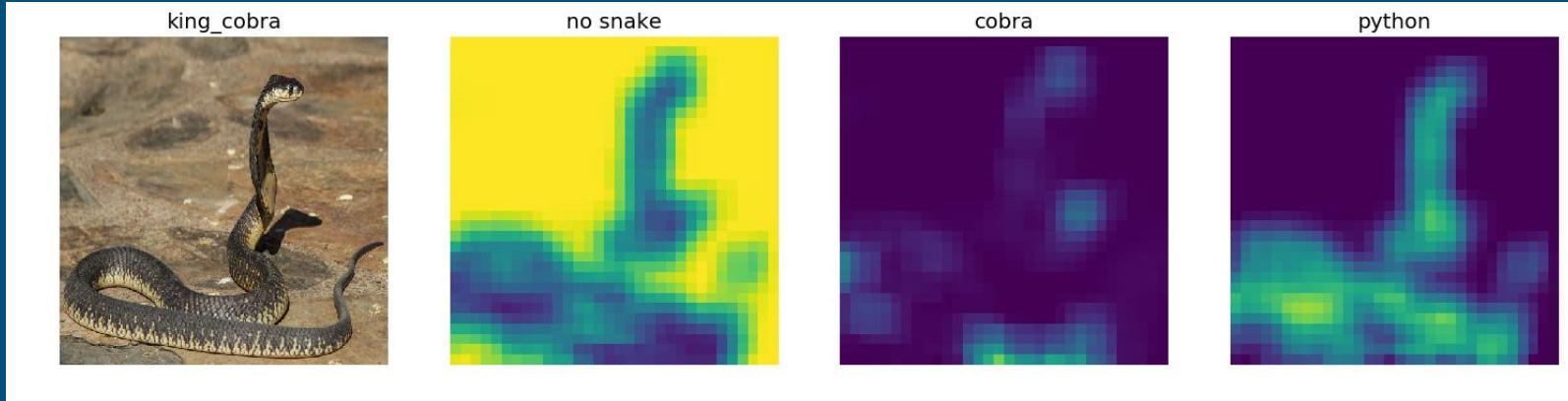
# Co zmieniliśmy?

---

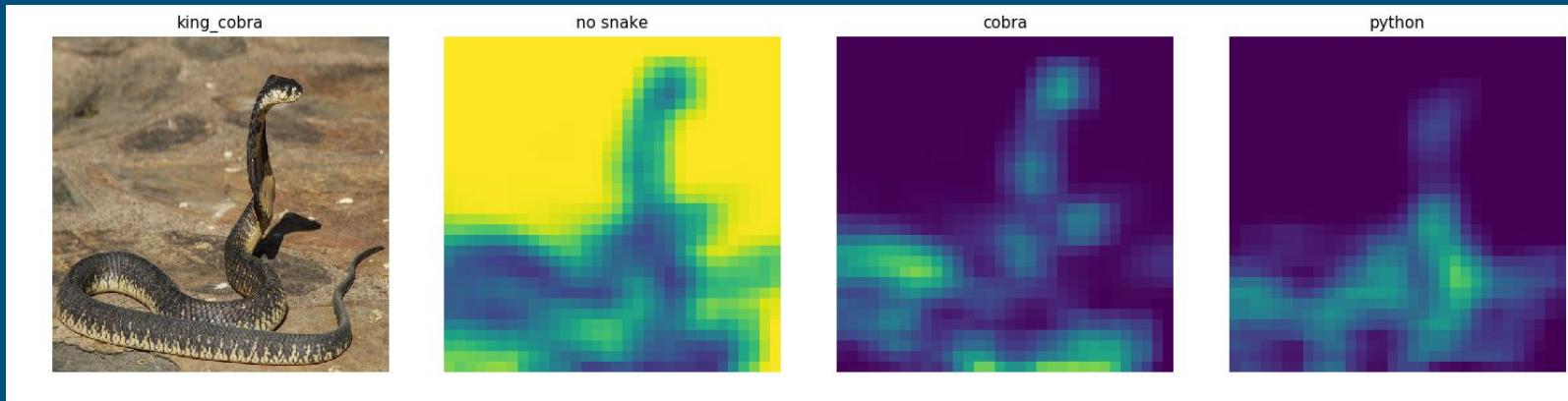
- Zmiana współczynnika “wypełnienia węża” dla kafelków
- Zmiana parametrów modelu
- Augmentacja danych



# Pierwotny model



# Po zmianach



# Lokalizacja bez rozróżniania gatunków

---

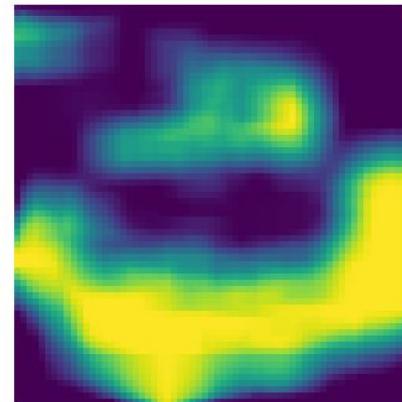
original



annotated



prediction



result



# Przykłady:

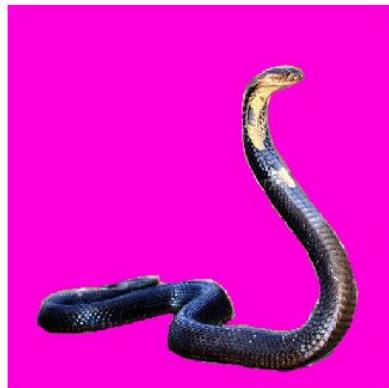
---



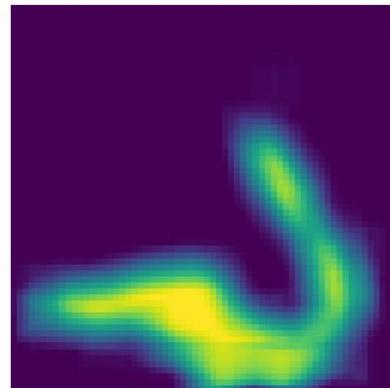
original



annotated



prediction



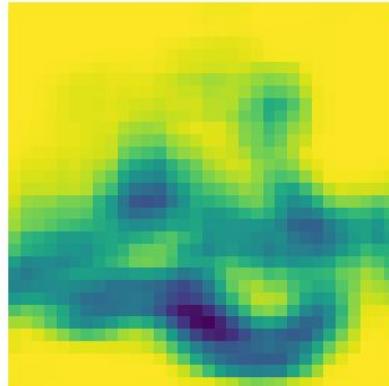
result



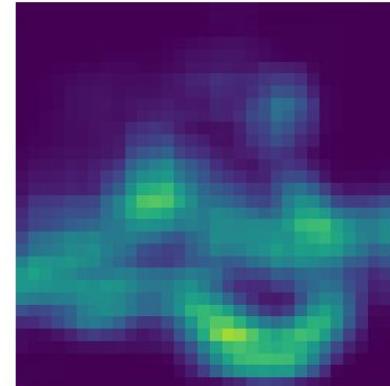
king\_cobra



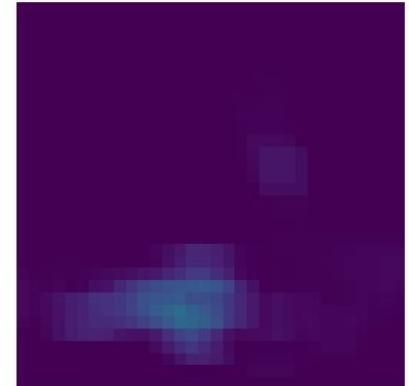
no snake



cobra



python





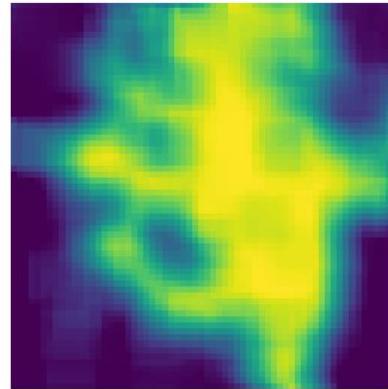
original



annotated



prediction



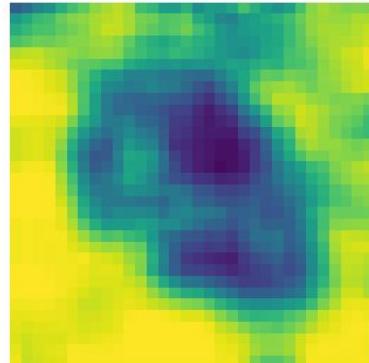
result



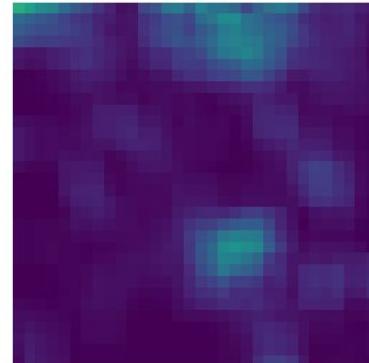
indian\_rock\_python



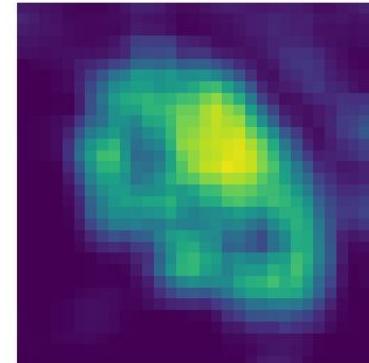
no snake



cobra



python





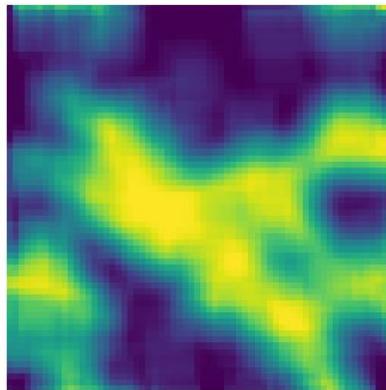
original



annotated



prediction



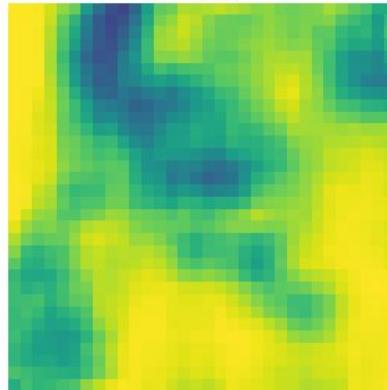
result



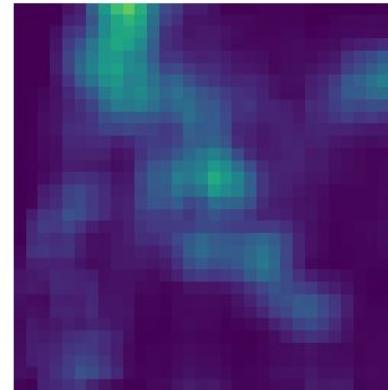
indian\_rock\_python



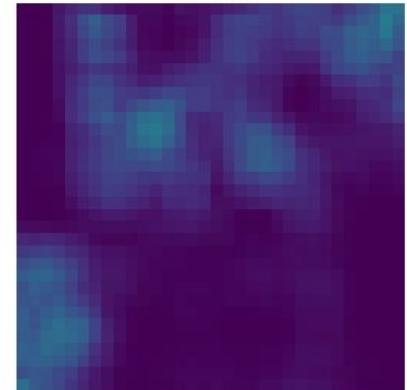
no snake



cobra



python



# Crash test

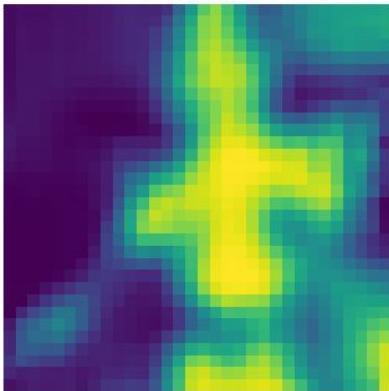
---



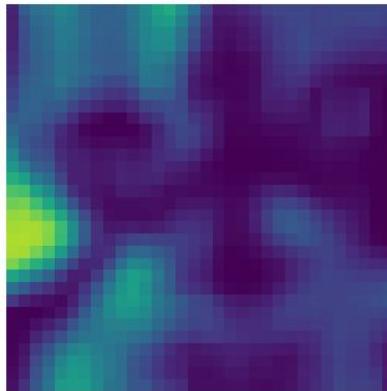
# Predykcje

---

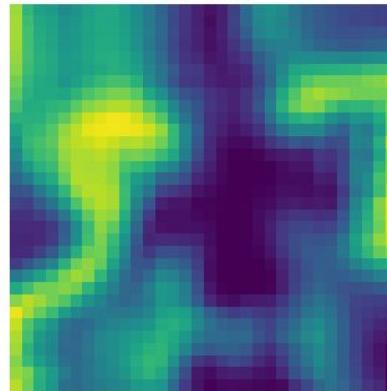
no snake



cobra



python



original



# Podsumowanie

---

- Wykrywanie obiektów na zdjęciach można zaimplementować za pomocą (dość) prostych sieci neuronowych
- Warstwy konwolucyjne uczą się cech charakterystycznych dla wykrywanych obiektów
- Odpowiednio bogaty i różnorodny zbiór danych to podstawa
- Dla osób zainteresowanych tworzeniem własnych modeli - stwórzcie jak najszybciej działający model i pracujcie nad nim iteracyjnie

# Kursy i źródła wiedzy

---

## Praktyczne wprowadzenie do dziedziny

- *Machine Learning* - Andrew Ng, Coursera  
[www.coursera.org/learn/machine-learning](http://www.coursera.org/learn/machine-learning)
- *Deep Learning* - Andrew Ng, Coursera  
[www.coursera.org/specializations/deep-learning](http://www.coursera.org/specializations/deep-learning)
- Intel AI Academy  
<https://software.intel.com/en-us/ai-academy/basics>

## Źródła akademickie

- *Convolutional Neural Networks for Visual Recognition* - Stanford University  
<http://cs231n.stanford.edu>
- *Deep Learning* (książka) - Goodfellow, Bengio, Courville

# Kontakt

---

- [piotr.bara@enigmapattern.com](mailto:piotr.bara@enigmapattern.com)
- [patryk.seweryn@enigmapattern.com](mailto:patryk.seweryn@enigmapattern.com)

Artykuł na LinkedIn:

<https://www.linkedin.com/pulse/deep-learning-pythons-cobras-patryk-seweryn/>

Dziękujemy za uwagę!

---