

**UNIWERSYTET KOMISJI EDUKACJI NARODOWEJ  
w Krakowie**



**Uniwersytet Komisji  
Edukacji Narodowej  
w Krakowie**

**INSTYTUT BEZPIECZEŃSTWA  
I INFORMATYKI**

Kierunek: INFORMATYKA

**Wiktor Mendalka**

Nr albumu: 167097

**Rozwój i badania systemu oprogramowania do rozpoznawania  
obrazów graficznych**

Praca magisterska napisana pod kierunkiem  
**dr hab. Serhii Semenov, prof. UP**

**Kraków 2024r.**

## ***Streszczenie***

Niniejsza praca koncentruje się na rozwoju systemu do rozpoznawania obrazów graficznych z wykorzystaniem sztucznych sieci neuronowych, szczególnie sieci Hopfielda i ART-1. Celem było opracowanie i implementacja tych sieci w celu poprawy dokładności rozpoznawania wzorców. W ramach pracy zaimplementowano obie architektury oraz przeprowadzono ich szczegółowe testy, analizując ich zdolność do odtwarzania wzorców graficznych oraz wydajność w zależności od liczby neuronów i pojemności sieci.

Sieć Hopfielda wykazała szybki czas konwergencji, ale ograniczoną dokładność przy bardziej złożonych zbiorach danych, podczas gdy sieć ART-1, dzięki dużej pojemności, wykazała potencjał w tworzeniu licznych kategorii. Mimo to, dokładność ART-1 była niska, co wskazuje na potrzebę dalszej optymalizacji. Praca podkreśla potrzebę dalszych badań i usprawnień w zakresie parametrów, takich jak czujność, aby zwiększyć efektywność i zastosowanie tych sieci w zadaniach analizy graficznej.

## ***Abstract***

This thesis focuses on the development of a system for recognizing graphic images using artificial neural networks, particularly the Hopfield and ART-1 networks. The aim was to design and implement these networks to improve pattern recognition accuracy. Both architectures were implemented and thoroughly tested, with analyses conducted on their ability to restore graphic patterns and their performance depending on the number of neurons and network capacity.

The Hopfield network demonstrated a fast convergence time but limited accuracy with more complex datasets, while the ART-1 network, due to its high capacity, showed potential in creating numerous categories. However, the accuracy of ART-1 was low, indicating the need for further optimization. This study highlights the need for continued research and improvements in parameters such as vigilance to enhance the efficiency and applicability of these networks in graphic analysis tasks.

# Spis treści

Wstęp.....	5
Rozdział 1 Przegląd analityczny.....	6
1.1. Rozpoznawanie wzorców w życiu codziennym.....	6
1.2. Klasyfikacja metod rozpoznawania.....	9
1.2.1. Metody statystyczne.....	9
1.2.2 Metody strukturalne.....	10
1.2.3 Metody syntaktyczne.....	10
1.2.4 Metody oparte na sztucznych sieciach neuronowych.....	11
1.2.5 Metody hybrydowe.....	12
1.2.6 Podsumowanie.....	13
1.3. Historyczny przegląd sieci neuronowych.....	13
1.3.1 Początki: Model McCullocha i Pittsa oraz perceptron.....	13
1.3.2 Okres stagnacji: „Zima sztucznej inteligencji”.....	14
1.3.3 Odrodzenie: Algorytm wstecznej propagacji błędów.....	14
1.3.4 Głębokie uczenie: „Renesans sieci neuronowych”.....	15
1.3.5 Nowe kierunki i zastosowania.....	15
1.3.6 Podsumowanie.....	16
1.4. Rozpoznawanie wzorców z wykorzystaniem sztucznych sieci neuronowych.....	16
1.4.1 Sieci wielowarstwowe perceptrony (MLP).....	16
1.4.2 Konwolucyjne sieci neuronowe (CNN).....	17
1.4.3 Sieci neuronowe adaptacyjnej teorii rezonansu (ART).....	18
1.4.4 Hybrydowe sieci neuronowe.....	18
1.4.5 Podsumowanie.....	19
1.5 Problem uczenia sieci neuronowych.....	19
1.5.1 Problem zanikania i eksplozji gradientu.....	19
1.5.2 Przeuczenie (overfitting).....	20
1.5.3 Trudności w trenowaniu głębokich sieci neuronowych.....	20
Rozdział 2: Modele matematyczne sieci neuronowych.....	22
2.1 Model matematyczny sztucznego neuronu.....	22
2.2 Sieć neuronowa Hopfielda.....	23
2.3 Sieci neuronowe adaptacyjnej teorii rezonansu (ART).....	24
2.4 Wykorzystanie modeli matematycznych w praktyce.....	25
<b>Rozdział 3: Rozwój sieci neuronowych.....</b>	<b>26</b>
3.1 Rozwój sieci neuronowej Hopfielda.....	26
3.1.1 Założenia projektowe.....	27
3.1.2 Architektura i implementacja.....	27
3.1.3 Pomiary i rezultaty dla sieci Hopfielda.....	29
3.1.4 Podsumowanie.....	31
3.2 Rozwój sieci ART-1.....	32
3.2.1 Założenia projektowe.....	32
3.2.2 Architektura i implementacja.....	33
3.2.3 Pomiary i rezultaty dla sieci ART-1.....	34

3.2.4 Podsumowanie.....	37
3.4 Testowanie sieci neuronowych.....	38
3.4.1 Wyniki testowania sieci Hopfielda.....	38
3.4.2 Wyniki testowania sieci ART-1.....	38
3.4.3 Porównanie z innymi typami sieci neuronowych.....	39
3.4.4 Wnioski z porównania.....	40
Zakończenie.....	41
Bibliografia.....	44

## Wstęp

Rozwój technologii sztucznej inteligencji (AI) zrewolucjonizował wiele dziedzin życia, w tym obszar rozpoznawania wzorców, który odgrywa kluczową rolę w analizie i przetwarzaniu danych wizualnych. Systemy rozpoznawania obrazów graficznych, oparte na sztucznych sieciach neuronowych, stały się nieodłącznym elementem nowoczesnych aplikacji, od medycyny, przez przemysł, po sztukę. W niniejszej pracy magisterskiej skupiamy się na badaniu i rozwoju systemu oprogramowania do rozpoznawania obrazów graficznych, uwzględniając przy tym najnowsze osiągnięcia w dziedzinie sztucznych sieci neuronowych. Szczególny nacisk położymy na adaptacyjną teorię rezonansu (ART).

Rozpoznawanie wzorców, które można zdefiniować jako proces identyfikacji struktur i regularności w danych jest fundamentalnym problemem w dziedzinie sztucznej inteligencji. W ciągu ostatnich dekad wraz z rozwojem technik uczenia maszynowego, szczególną uwagę poświęcono sieciom neuronowym, które naśladują procesy zachodzące w ludzkim mózgu. Zastosowanie sieci neuronowych do rozpoznawania obrazów graficznych pozwala na automatyzację i zwiększenie efektywności procesów, które wcześniej były zarezerwowane wyłącznie dla ludzkiego oka.

Praca ta ma na celu przegląd metod rozpoznawania wzorców z uwzględnieniem zarówno tradycyjnych podejść, jak i nowoczesnych technik bazujących na sztucznych sieciach neuronowych. Głównie skupimy się na sieci neuronowej ART-1, której adaptacja do współczesnych systemów oprogramowania stanowi główny aspekt naukowy tej pracy. W ramach badań przeprowadzimy implementację i testowanie sieci ART-1, a także przeanalizujemy możliwości jej zastosowania w różnych dziedzinach, dążąc do poprawy jakości rozpoznawania wzorców.

Dzięki wykorzystaniu nowoczesnych modeli matematycznych oraz zaawansowanych technik uczenia maszynowego, praca ta wniesie wkład w rozwój systemów rozpoznawania obrazów, otwierając nowe możliwości dla ich praktycznych zastosowań. W niniejszej pracy zaprezentowane zostaną zarówno teoretyczne aspekty rozpoznawania wzorców, jak i praktyczne wyniki badań nad rozwojem i adaptacją sieci neuronowej ART-1.

# Rozdział 1 Przegląd analityczny

## 1.1. Rozpoznawanie wzorców w Życiu codziennym

Rozpoznawanie wzorców jest jedną z najważniejszych umiejętności zarówno dla organizmów żywych, jak i dla systemów sztucznej inteligencji. W kontekście biologicznym, zdolność do identyfikacji i interpretacji wzorców w środowisku była i jest kluczowa dla przetrwania. Dla ludzi oznacza to zdolność do rozpoznawania znajomych twarzy, identyfikacji zagrożeń, czy też rozumienia skomplikowanych sygnałów społecznych i językowych. W podobny sposób, sztuczna inteligencja, aby efektywnie funkcjonować, musi być zdolna do identyfikacji wzorców w danych, co umożliwia jej podejmowanie decyzji, przewidywanie oraz wykonywanie określonych zadań.

W życiu codziennym rozpoznawanie wzorców odbywa się na różnych poziomach. Najbardziej oczywistym przykładem jest rozpoznawanie twarzy, które jest możliwe dzięki zdolności mózgu do identyfikacji subtelnych różnic w strukturach twarzy, takich jak odległości między oczami, kształt ust, czy kontur linii szczęki. Proces ten zachodzi w ułamkach sekund i pozwala ludziom nie tylko na identyfikację konkretnych osób, ale również na interpretację emocji i intencji na podstawie wyrazu twarzy.

Innym przykładem jest rozpoznawanie mowy, gdzie mózg przetwarza dźwięki w słowa, zdania i intencje. To umożliwia komunikację i wymianę informacji. Proces ten jest niezwykle złożony, ponieważ wymaga analizy różnych aspektów dźwięku – tonu, rytmu, akcentu oraz kontekstu, co pozwala na rozróżnienie podobnych słów i ich znaczeń w zależności od sytuacji.

W systemach technologicznych, wzorce są rozpoznawane za pomocą algorytmów, które przetwarzają dane wejściowe i porównują je z wzorcami zapisanymi w pamięci systemu. W kontekście przetwarzania obrazu, systemy te mogą identyfikować obiekty na podstawie ich kształtu, koloru, tekstury, a także innych cech charakterystycznych. Na przykład, w medycynie, technologie rozpoznawania wzorców są wykorzystywane do analizy obrazów diagnostycznych, takich jak zdjęcia rentgenowskie, tomografie komputerowe, czy obrazy MRI, w celu wykrycia anomalii, które mogą wskazywać na obecność chorób.

Rozpoznawanie wzorców znajduje zastosowanie również w automatycznym tłumaczeniu tekstu i mowy. W systemach tych, analiza wzorców gramatycznych, semantycznych oraz kontekstualnych umożliwia przekład treści z jednego języka na inny.

Systemy technologiczne stale się uczą, analizując ogromne ilości danych, zwiększając przy tym precyzję tłumaczeń.

W sektorze bezpieczeństwa, rozpoznawanie wzorców odgrywa kluczową rolę w technologiach biometrycznych. Na przykład, rozpoznawanie odcisków palców, analizowanie układu żył, skanowanie tęczówki oka, czy analiza układu twarzy, to technologie wykorzystywane w celu weryfikacji tożsamości. Systemy te, bazując na unikalnych wzorcach biologicznych każdego człowieka, znajdują zastosowanie w różnych dziedzinach, od zabezpieczania dostępu do urządzeń mobilnych po kontrolę dostępu na lotniskach i w instytucjach finansowych.

W bardziej zaawansowanych zastosowaniach, takich jak autonomiczne pojazdy, rozpoznawanie wzorców pozwala systemom na identyfikację pieszych, pojazdów, sygnalizacji świetlnej oraz innych elementów drogowych, co jest kluczowe dla bezpiecznego poruszania się w ruchu drogowym. Te systemy nie tylko rozpoznają obiekty, ale także przewidują ich ruch i podejmują decyzje w czasie rzeczywistym, co wymaga niezwykle zaawansowanych algorytmów i dużej mocy obliczeniowej.

W handlu elektronicznym, systemy rekomendacji wykorzystujące techniki rozpoznawania wzorców analizują zachowania zakupowe klientów, ich preferencje oraz historię przeglądania, aby proponować produkty, które mogą ich zainteresować. Algorytmy te rozpoznają wzorce w danych dotyczących milionów użytkowników, co pozwala na personalizację oferty i zwiększenie efektywności sprzedaży.

Wszystkie te przykłady pokazują, jak wszechobecne jest rozpoznawanie wzorców w życiu codziennym oraz jak różnorodne są jego zastosowania w nowoczesnych technologiach. W kontekście sztucznej inteligencji, umiejętność ta jest kluczowa dla rozwoju systemów zdolnych do samodzielnego uczenia się, adaptacji i podejmowania decyzji na podstawie analizy dużych zbiorów danych. Niniejsza praca poświęcona jest badaniu tej umiejętności w kontekście sztucznych sieci neuronowych, które naśladując biologiczne procesy rozpoznawania wzorców, otwierają nowe możliwości w dziedzinie automatyzacji i przetwarzania informacji wizualnych. Szczególną uwagę poświęcimy sieciom neuronowym ART-1, które stanowią nowoczesne podejście do problemu rozpoznawania wzorców i mają potencjał do znacznej poprawy jakości analiz w wielu dziedzinach życia.

Rozpoznawanie wzorców jest obszarem intensywnie badanym w wielu dziedzinach nauki, od biologii i neurobiologii po informatykę i sztuczną inteligencję. W ostatnich dekadach pojawiło się wiele prac naukowych, które przyczyniły się do zrozumienia

mechanizmów leżących u podstaw tej kluczowej zdolności oraz do rozwoju technologii umożliwiających jej zastosowanie w systemach sztucznej inteligencji.

W badaniach z zakresu neurobiologii, Grossberg (2013) przedstawił adaptacyjną teorię rezonansu (ART) jako model, który wyjaśnia, jak ludzki mózg przetwarza wzorce w dynamicznym i często niepełnym środowisku percepcyjnym. ART oferuje mechanizm umożliwiający stabilne uczenie się w odpowiedzi na nowe dane, co czyni ją szczególnie atrakcyjną dla zastosowań w sztucznej inteligencji, gdzie stabilność uczenia i adaptacyjność są kluczowe .

W dziedzinie sztucznej inteligencji, prace nad sieciami neuronowymi ART przyczyniły się do znaczącego postępu w rozpoznawaniu wzorców. Dmitrienko i współpracownicy (2014) zaprezentowali nowy algorytm uczenia dla sieci ART, który umożliwia rozwiązywanie problemów z wieloma możliwymi rozwiązaniami. Algorytm ten poprawia zdolność sieci do radzenia sobie z bardziej złożonymi wzorcami, co ma bezpośrednie przełożenie na aplikacje w dziedzinie rozpoznawania obrazów .

W kontekście inżynierii, Silva, Elnabarawy i Wunsch (2019) przeprowadzili szczegółowy przegląd modeli sieci neuronowych ART, skupiając się na ich zastosowaniach inżynierskich. Autorzy wskazali, że sieci ART znalazły zastosowanie w różnych dziedzinach, takich jak diagnostyka medyczna, systemy rekomendacyjne oraz analiza obrazów satelitarnych. Ich badania podkreślają, że sieci ART, dzięki swojej zdolności do stabilnego uczenia i adaptacji, oferują istotne korzyści w kontekście systemów, które muszą przetwarzać i interpretować złożone dane wzorcowe w czasie rzeczywistym .

Podsumowując, dotychczasowe badania wykazały, że rozpoznawanie wzorców jest procesem kluczowym zarówno w kontekście biologicznym, jak i technologicznym. Prace nad sieciami neuronowymi, szczególnie nad sieciami ART, przyczyniły się do znacznego postępu w tej dziedzinie, co otwiera nowe możliwości zastosowań w różnych sektorach przemysłu i nauki. W dalszej części pracy skupimy się na implementacji i testowaniu sieci ART-1 w kontekście rozpoznawania wzorców graficznych, co pozwoli na dalsze rozwinięcie wiedzy zdobytej w dotychczasowych badaniach.



## 1.2. Klasyfikacja metod rozpoznawania

Rozpoznawanie wzorców jest kluczowym zagadnieniem w dziedzinie sztucznej inteligencji, informatyki oraz inżynierii danych wraz z szerokim zakresem zastosowań od analizy obrazów po rozpoznawanie mowy. Aby zrozumieć i skutecznie rozwijać technologię rozpoznawania wzorców, konieczne jest zbadanie i sklasyfikowanie istniejących metod badania, analizy i identyfikacji wzorców w danych. Klasyfikacja metod rozpoznawania wzorców obejmuje różne podejścia, które można podzielić na kilka głównych kategorii: metody oparte na statystyce, metody strukturalne, metody syntaktyczne oraz metody oparte na sztucznych sieciach neuronowych. Każda z tych kategorii ma swoje unikalne cechy, zalety oraz ograniczenia, które determinują ich zastosowanie w różnych kontekstach.

### 1.2.1. Metody statystyczne

Metody statystyczne należą do najstarszych i najczęściej stosowanych technik rozpoznawania wzorców. Ich podstawą jest analiza rozkładu prawdopodobieństwa danych oraz wykorzystanie technik statystycznych do klasyfikacji i prognozowania. W ramach tych metod, wzorce są reprezentowane jako wektory cech, a klasyfikacja polega na przydzieleniu nowych danych do jednej z klas na podstawie prawdopodobieństwa warunkowego. Klasycznym przykładem jest analiza dyskryminacyjna Fishera, która minimalizuje wewnątrzklasową wariancję przy jednoczesnym maksymalizowaniu różnic międzyklasowych, co prowadzi do optymalnej separacji danych (Fisher, 1936).

Inne popularne metody statystyczne obejmują klasyfikację Bayesowską oraz maszyny wektorów nośnych (SVM). Klasyfikacja Bayesowska opiera się na twierdzeniu Bayesa, które pozwala na obliczenie prawdopodobieństwa przynależności wzorca do określonej klasy na podstawie dostępnych danych i wcześniejszych rozkładów prawdopodobieństwa (Duda, Hart & Stork, 2000). Maszyny wektorów nośnych, zaproponowane przez Vapnika (1995), są natomiast techniką, która znajduje optymalną hiperpłaszczyznę oddzielającą klasy danych w przestrzeni cech, co umożliwia efektywną klasyfikację nawet w przypadkach, gdy dane są liniowo nieoddzielne.

Zalety metod statystycznych:

- **Prostota i efektywność:** Łatwe do implementacji i skuteczne przy dobrze zdefiniowanych danych.

- **Teoretyczne podstawy:** Bogata literatura i narzędzia wspierające stosowanie.
- **Interpretowalność:** Modele są łatwe do zrozumienia i wyjaśnienia.

Ograniczenia metod statystycznych:

- **Założenia o danych:** Zakładają określone rozkłady, co nie zawsze jest zgodne z rzeczywistością.
- **Wysokowymiarowość:** Tracą efektywność przy dużej liczbie wymiarów.
- **Wrażliwość na szum:** Problemy przy pracy z danymi zakłóconymi.
- **Ograniczona do liniowości:** Często ograniczone do liniowych separacji, trudności z nieliniowymi problemami.

### 1.2.2 Metody strukturalne

Metody strukturalne koncentrują się na rozpoznawaniu złożonych wzorców, które można opisać za pomocą hierarchicznych struktur, takich jak drzewa, grafy czy sieci. W przeciwieństwie do metod statystycznych, które traktują wzorce jako wektory cech, metody strukturalne modelują wzorce jako zestawy elementów połączonych relacjami strukturalnymi. Przykładem może być rozpoznawanie obrazów na podstawie analizy ich elementów składowych, takich jak krawędzie, narożniki czy regiony, oraz relacji między nimi.

W literaturze naukowej, jedną z najbardziej znanych metod strukturalnych jest teoria grafów, która znajduje szerokie zastosowanie w rozpoznawaniu wzorców, gdzie wzorce są reprezentowane jako wierzchołki i krawędzie grafu (Conte et al., 2004). Rozpoznawanie wzorców w oparciu o teorię grafów jest szczególnie efektywne w analizie obrazów i rozpoznawaniu obiektów, gdzie relacje przestrzenne między elementami wzorca odgrywają kluczową rolę. Metody te są jednak zazwyczaj bardziej złożone obliczeniowo niż metody statystyczne, co może ograniczać ich zastosowanie w przypadku dużych zbiorów danych.

### 1.2.3 Metody syntaktyczne

Metody syntaktyczne, nazywane również metodami gramatycznymi, są rozszerzeniem metod strukturalnych, które wykorzystują gramatyki formalne do opisu wzorców. Wzorce są tu traktowane jako wyrażenia w określonym języku formalnym, a proces rozpoznawania polega na zidentyfikowaniu gramatyki generującej dany wzorec. Jednym z głównych atutów

metod syntaktycznych jest ich zdolność do rozpoznawania wzorców o złożonej, hierarchicznej strukturze, co jest trudne do osiągnięcia za pomocą innych metod (Fu, 1982).

W kontekście rozpoznawania obrazów, metody syntaktyczne pozwalają na opisanie złożonych obiektów jako sekwencji prostszych elementów, zgodnie z regułami określonymi przez gramatykę. Na przykład, w rozpoznawaniu znaków drogowych, znak może być opisany jako złożenie kilku podstawowych kształtów, takich jak koło, trójkąt czy prostokąt, zgodnie z określonymi regułami syntaktycznymi. Choć metody syntaktyczne oferują dużą elastyczność, ich praktyczne zastosowanie jest często ograniczone przez trudności związane z automatycznym wyznaczaniem odpowiednich gramatyk oraz wysoką złożonością obliczeniową (Pavlidis, 1977).

#### 1.2.4 Metody oparte na sztucznych sieciach neuronowych

W ostatnich dekadach metody oparte na sztucznych sieciach neuronowych (ANN) zyskały na znaczeniu, stając się jednymi z najbardziej efektywnych narzędzi w rozpoznawaniu wzorców. Sieci neuronowe, inspirowane biologicznymi sieciami nerwowymi, uczą się rozpoznawać wzorce na podstawie przykładów, co pozwala na skuteczną klasyfikację danych nawet w przypadku skomplikowanych i nieuporządkowanych zbiorów danych (LeCun, Bengio & Hinton, 2015).

W ramach sieci neuronowych istnieje wiele różnych architektur, które znajdują zastosowanie w rozpoznawaniu wzorców. Sieci wielowarstwowe perceptrony (MLP) są jedną z najprostszych form ANN, które uczą się poprzez propagację wsteczną błędów, co umożliwia efektywne rozpoznawanie liniowo nieliniowych wzorców (Rumelhart, Hinton & Williams, 1986). Kolejnym kluczowym podejściem są sieci splotowe (CNN), które odniosły ogromny sukces w dziedzinie rozpoznawania obrazów. CNN są specjalnie zaprojektowane do analizy danych o strukturze gridowej, takich jak obrazy, dzięki zastosowaniu warstw splotowych, które automatycznie uczą się odpowiednich filtrów cech (Krizhevsky, Sutskever & Hinton, 2012).

Adaptacyjna teoria rezonansu (ART), zaproponowana przez Grossberga (1980), stanowi kolejne ważne podejście w dziedzinie ANN. ART pozwala na stabilne uczenie się w czasie rzeczywistym, co jest kluczowe w przypadku systemów, które muszą dynamicznie przetwarzać nowe informacje bez zapominania wcześniej nabytych wzorców. Sieci ART, dzięki swojej zdolności do ciągłej adaptacji, znalazły zastosowanie w różnorodnych zadaniach, od analizy obrazów po rozpoznawanie mowy (Carpenter & Grossberg, 1987).

Ograniczenia metod opartych na sztucznych sieciach neuronowych:

- **Wysokie wymagania obliczeniowe:** Trening sieci neuronowych, szczególnie tych głębokich, takich jak CNN, wymaga znacznych zasobów obliczeniowych (GPU) oraz dużych zbiorów danych, co może być kosztowne i czasochłonne.
- **Skłonność do przeuczenia (overfitting):** Sieci neuronowe, zwłaszcza w przypadku złożonych modeli, mogą łatwo dostosować się do danych treningowych i wykazywać niską zdolność do generalizacji na nowe, nieznanne dane, jeśli nie zostaną zastosowane odpowiednie techniki regularyzacji.
- **Brak interpretowalności:** Sieci neuronowe są często traktowane jako "czarne skrzynki". Trudno jest zrozumieć, jakie dokładnie zależności wyłapują, co utrudnia interpretację wyników, zwłaszcza w zadaniach wymagających przejrzystości i wyjaśnialności modelu.
- **Wymóg dużych zbiorów danych:** Sieci neuronowe zazwyczaj potrzebują ogromnych ilości danych treningowych, aby osiągnąć wysoką skuteczność. W przypadku małych zbiorów danych modele mogą nie spełniać oczekiwań i dawać błędne predykcje.
- **Złożoność konfiguracji:** Sieci neuronowe wymagają precyzyjnego dostrojenia hiperparametrów, takich jak liczba warstw, liczba neuronów, funkcje aktywacji, oraz metody optymalizacji, co czyni proces projektowania bardziej skomplikowanym i podatnym na błędy.

### 1.2.5 Metody hybrydowe

Ze względu na złożoność i różnorodność danych wzorcowych, często konieczne jest łączenie różnych podejść w celu uzyskania najlepszych wyników. Metody hybrydowe integrują cechy metod statystycznych, strukturalnych, syntaktycznych oraz neuronowych, aby skutecznie radzić sobie z wyzwaniami związanymi z rozpoznawaniem wzorców w złożonych środowiskach. Na przykład, hybrydowe systemy rozpoznawania obrazów mogą wykorzystywać CNN do ekstrakcji cech, a następnie zastosować metody Bayesowskie lub SVM do końcowej klasyfikacji (Szegedy et al., 2015).

Przykładem takiego podejścia jest zastosowanie sieci neuronowych w połączeniu z metodami grafowymi w analizie obrazów medycznych. Sieci CNN mogą być użyte do ekstrakcji cech z obrazów, natomiast analiza grafowa może pomóc w zrozumieniu relacji

między różnymi regionami obrazów, co prowadzi do bardziej precyzyjnych diagnoz (Zhou i inni., 2019).

### 1.2.6 Podsumowanie

Klasyfikacja metod rozpoznawania wzorców obejmuje szeroki zakres podejść, z których każde ma swoje specyficzne zalety i ograniczenia. Metody statystyczne oferują solidne podstawy teoretyczne i skuteczność w przypadku dobrze zdefiniowanych danych. Metody strukturalne i syntaktyczne pozwalają na analizę bardziej złożonych i hierarchicznych struktur wzorców, choć często kosztem większej złożoności obliczeniowej. Metody oparte na sztucznych sieciach neuronowych, dzięki swojej elastyczności i zdolności do uczenia się na podstawie przykładów, stały się dominującym podejściem w wielu współczesnych zastosowaniach. Wreszcie, metody hybrydowe łączą różne techniki, aby maksymalizować skuteczność w obliczu złożonych problemów rozpoznawania wzorców.

Literatura naukowa dostarcza bogactwa przykładów na to, jak różne podejścia mogą być efektywnie stosowane w praktyce. Każda z metod ma swoje unikalne zastosowania, które są dostosowane do specyficznych wyzwań związanych z typem danych oraz wymaganiami aplikacji. W kolejnych częściach pracy zostaną omówione bardziej specyficzne zastosowania tych metod, ze szczególnym uwzględnieniem sieci neuronowych ART i ich adaptacji do rozpoznawania wzorców graficznych.

## 1.3. Historyczny przegląd sieci neuronowych

Sieci neuronowe, będące jednym z kluczowych narzędzi w dziedzinie sztucznej inteligencji, mają bogatą historię, której początki sięgają lat 40. XX wieku. Rozwój tej technologii można podzielić na kilka etapów, od wczesnych modeli inspirowanych biologią, przez okres stagnacji, aż po współczesny renesans związany z rozwojem głębokiego uczenia. Historia sieci neuronowych odzwierciedla zarówno postępy technologiczne, jak i zmieniające się paradygmaty badawcze.

### 1.3.1 Początki: Model McCullocha i Pittsa oraz perceptron

Historia sieci neuronowych rozpoczyna się w 1943 roku, kiedy to Warren McCulloch i Walter Pitts opublikowali pracę, w której zaproponowali pierwszy formalny model

sztucznego neuronu. Model ten, oparty na podstawowych zasadach logiki, miał na celu symulację sposobu, w jaki biologiczne neurony przetwarzają informacje (McCulloch & Pitts, 1943). Ich praca zainspirowała kolejne pokolenia badaczy do eksploracji możliwości tworzenia maszyn naśladowujących działanie ludzkiego mózgu.

Kolejnym ważnym krokiem w rozwoju sieci neuronowych było opracowanie perceptronu przez Franka Rosenblatta w 1958 roku. Perceptron był prostym modelem sieci neuronowej zdolnym do klasyfikacji liniowo separowalnych wzorców, co uczyniło go pierwszym praktycznym zastosowaniem sieci neuronowych w rozpoznawaniu wzorców (Rosenblatt, 1958). Mimo początkowego entuzjazmu, ograniczenia perceptronu, zidentyfikowane przez Minsky'ego i Paperta (1969), szczególnie brak zdolności do rozwiązywania problemów nieliniowych, doprowadziły do okresu stagnacji w badaniach nad sieciami neuronowymi.

### 1.3.2 Okres stagnacji: „Zima sztucznej inteligencji”

W latach 70. XX wieku rozwój sieci neuronowych znacząco zwolnił, co było spowodowane krytyką i ograniczeniami perceptronu. Publikacja Minsky'ego i Paperta (1969), zatytułowana *Perceptrons*, wykazała, że perceptron nie jest w stanie rozwiązać problemów takich jak funkcja XOR, co zniechęciło badaczy do kontynuowania prac w tej dziedzinie. Ten okres, często określany jako „zima sztucznej inteligencji” trwał przez około dekadę. Wówczas badania nad sieciami neuronowymi zostały w dużej mierze zepchnięte na margines na rzecz innych metod, takich jak systemy ekspertowe (Crevier, 1993).

### 1.3.3 Odrodzenie: Algorytm wstecznej propagacji błędów

Przełom nastąpił na początku lat 80. XX wieku, kiedy to niezależnie od siebie Paul Werbos (1974) oraz zespół Davida Rumelhart, Geoffreya Hinton i Ronalda Williams (1986) opracowali algorytm wstecznej propagacji błędów (backpropagation), który umożliwił skuteczne trenowanie wielowarstwowych perceptronów. Algorytm ten pozwalał na obliczanie gradientów i aktualizowanie wag w sieciach wielowarstwowych, co otworzyło drzwi do trenowania bardziej złożonych sieci zdolnych do rozwiązywania problemów nieliniowych (Rumelhart, Hinton & Williams, 1986).

Wprowadzenie algorytmu wstecznej propagacji błędów zrewolucjonizowało dziedzinę sztucznych sieci neuronowych i przyczyniło się do ich odrodzenia. W latach 80. i 90. XX wieku rozwój technologii komputerowej oraz dostęp do większych zbiorów danych umożliwiły rozwój bardziej złożonych modeli sieci neuronowych, co z kolei doprowadziło do ich szerokiego zastosowania w różnych dziedzinach, takich jak rozpoznawanie mowy i obrazów (LeCun et al., 1989).

#### 1.3.4 Głębokie uczenie: "Renesans sieci neuronowych"

Na początku XXI wieku, dzięki wzrostowi mocy obliczeniowej oraz dostępności dużych zbiorów danych, sieci neuronowe weszły w nową erę, zwaną "erą głębokiego uczenia" (deep learning). Głębokie sieci neuronowe, zbudowane z wielu warstw, pozwalają na automatyczne ekstrakcje cech i skuteczne rozpoznawanie wzorców w złożonych danych. Kluczowym momentem było wprowadzenie konwolucyjnych sieci neuronowych (CNN) przez Yann LeCun i jego zespół, które zrewolucjonizowały rozpoznawanie obrazów (LeCun, Bengio & Hinton, 2015; Krizhevsky, Sutskever & Hinton, 2012).

W 2012 roku zespół pod kierownictwem Hinton'a odniósł spektakularny sukces w konkursie ImageNet, wygrywając z ogromną przewagą dzięki zastosowaniu głębokich sieci CNN (Krizhevsky, Sutskever & Hinton, 2012). To wydarzenie zapoczątkowało szerokie zainteresowanie głębokim uczeniem i doprowadziło do eksplozji badań nad tym podejściem, obejmujących nie tylko rozpoznawanie obrazów, ale także przetwarzanie języka naturalnego, generowanie tekstów, a nawet samodzielne uczenie się gier przez komputery (Mnih et al., 2015; Vaswani et al., 2017).

#### 1.3.5 Nowe kierunki i zastosowania

Współcześnie rozwój sieci neuronowych kontynuuje się w wielu kierunkach. Jednym z nich jest uczenie przez wzmacnianie (reinforcement learning), które w połączeniu z głębokimi sieciami neuronowymi doprowadziło do stworzenia systemów, takich jak AlphaGo, które przewyższają ludzkie możliwości w złożonych grach strategicznych (Silver et al., 2016). Innym obszarem badań jest adaptacyjna teoria rezonansu (ART), która znajduje zastosowanie w systemach wymagających stabilnego uczenia się w zmiennym środowisku (Grossberg, 2013).

Sieci neuronowe są obecnie integralną częścią wielu systemów sztucznej inteligencji stosowanych w przemyśle, medycynie, finansach oraz innych dziedzinach. Ich zdolność do przetwarzania i analizowania dużych zbiorów danych oraz uczenia się skomplikowanych wzorców sprawia, że są one kluczowym narzędziem w rozwoju nowoczesnych technologii (Goodfellow, Bengio & Courville, 2016).

### 1.3.6 Podsumowanie

Historia sieci neuronowych to opowieść o przełomach, stagnacjach i dynamicznym rozwoju. Od skromnych początków inspirowanych neurobiologią, przez okres krytyki i zaniedbania, aż po współczesny renesans, sieci neuronowe ewoluowały w potężne narzędzie wykorzystywane w niemal każdej dziedzinie nauki i technologii. Ich rozwój pokazuje, jak istotna jest interdyscyplinarność badań oraz zdolność do adaptacji do nowych wyzwań technologicznych.

## 1.4. Rozpoznawanie wzorców z wykorzystaniem sztucznych sieci neuronowych

Rozpoznawanie wzorców przy użyciu sztucznych sieci neuronowych (ANN) to jeden z najbardziej dynamicznie rozwijających się obszarów badań w dziedzinie sztucznej inteligencji. Sieci neuronowe, inspirowane biologicznymi procesami przetwarzania informacji w mózgu, oferują potężne narzędzia do analizy, klasyfikacji i rozpoznawania skomplikowanych wzorców w danych. W niniejszym rozdziale omówimy, w jaki sposób sztuczne sieci neuronowe są wykorzystywane do rozpoznawania wzorców, ze szczególnym uwzględnieniem różnych typów sieci, takich jak sieci wielowarstwowe perceptrony (MLP), konwolucyjne sieci neuronowe (CNN) oraz adaptacyjne sieci neuronowe, takie jak sieci ART.

### 1.4.1 Sieci wielowarstwowe perceptrony (MLP)

Sieci wielowarstwowe perceptrony (MLP) to jedna z podstawowych architektur ANN wykorzystywanych do rozpoznawania wzorców. Składają się one z warstwy wejściowej, jednej lub więcej warstw ukrytych oraz warstwy wyjściowej. Każdy neuron w danej warstwie jest połączony z neuronami w warstwie poprzedzającej oraz następnej za pomocą wag, które są optymalizowane w procesie uczenia sieci.



MLP są szeroko stosowane w zadaniach klasyfikacji i regresji, gdzie uczą się mapowania wejściowych danych na wyjściowe klasy poprzez proces optymalizacji funkcji błędu. Przykładem może być funkcja strat oparta na różnicy między przewidywanymi i rzeczywistymi wynikami (Rumelhart, Hinton & Williams, 1986). MLP są szczególnie skuteczne w rozpoznawaniu wzorców, gdy dane są liniowo separowalne lub gdy dodanie nieliniowych funkcji aktywacji, takich jak ReLU czy sigmoidalna, pozwala na rozdzielenie bardziej złożonych wzorców (Goodfellow, Bengio & Courville, 2016).

#### 1.4.2 Konwolucyjne sieci neuronowe (CNN)

Konwolucyjne sieci neuronowe (CNN) to architektura ANN zaprojektowana specjalnie do przetwarzania danych o strukturze gridowej, takich jak obrazy. CNN zyskały ogromną popularność dzięki swojej zdolności do automatycznej ekstrakcji cech z danych wejściowych, co jest kluczowe w zadaniach rozpoznawania obrazów, segmentacji oraz detekcji obiektów.

Architektura CNN składa się z kilku warstw konwolucyjnych, w których filtr (lub kernel) przesuwa się po danych wejściowych, wykrywając lokalne wzorce, takie jak krawędzie, tekstury czy inne istotne cechy wizualne (LeCun et al., 1998). Filtry te uczą się automatycznie w procesie trenowania sieci, co umożliwia CNN adaptację do różnorodnych zadań rozpoznawania wzorców. Wraz z warstwami konwolucyjnymi, CNN zawierają również warstwy poolingowe, które redukują rozmiar danych, zachowując najważniejsze cechy (Krizhevsky, Sutskever & Hinton, 2012).

CNN odniosły szczególny sukces w konkursach takich jak ImageNet, gdzie osiągnęły znacznie lepsze wyniki w porównaniu z tradycyjnymi metodami rozpoznawania wzorców (He et al., 2016). Ich zastosowanie obejmuje szeroki zakres dziedzin, od analizy obrazów medycznych po systemy rozpoznawania twarzy i autonomiczne pojazdy.

#### 1.4.3 Sieci neuronowe adaptacyjnej teorii rezonansu (ART)

Model ART składa się z dwóch głównych komponentów: pola rozpoznawania, które przetwarza sygnały wejściowe, oraz pola porównywania, które dopasowuje te sygnały do wcześniej nauczonych kategorii (Grossberg, 2013).

Sieci neuronowe adaptacyjnej teorii rezonansu (ART) zostały opracowane w celu rozwiązania problemów związanych z stabilnością i plastycznością uczenia się, co jest kluczowe w zadaniach rozpoznawania wzorców w zmiennych środowiskach. W odróżnieniu od klasycznych sieci neuronowych, które mogą cierpieć na problem "katastrofalnego zapominania" (ang. catastrophic forgetting), sieci ART są zdolne do stabilnego uczenia się nowych wzorców bez utraty wcześniej nauczonych informacji (Carpenter & Grossberg, 1987).

Sieci ART znalazły zastosowanie w zadaniach wymagających wysokiej odporności na zmiany w danych wejściowych, takich jak rozpoznawanie mowy, gdzie sygnały mogą ulegać zakłóceniom, czy w systemach rozpoznawania pisma odręcznego, gdzie występuje duża zmienność w stylach pisma (Carpenter, Grossberg & Reynolds, 1991).

#### 1.4.4 Hybrydowe sieci neuronowe

Współczesne badania często łączą różne podejścia do rozpoznawania wzorców, tworząc hybrydowe sieci neuronowe, które wykorzystują zalety różnych architektur. Przykładem może być już wcześniej wspomniane połączenie CNN z sieciami ART w celu stworzenia systemu zdolnego do stabilnego uczenia się nowych wzorców w oparciu o dane obrazowe (Zhou et al., 2019). Inne podejścia hybrydowe obejmują integrację sieci neuronowych z metodami grafowymi, co pozwala na rozpoznawanie złożonych relacji między obiektami w danych (Scarselli et al., 2009).

Hybrydowe sieci neuronowe oferują elastyczność i adaptacyjność, co jest szczególnie istotne w kontekście nowoczesnych aplikacji, które muszą przetwarzać złożone, wielowymiarowe dane w czasie rzeczywistym. Tego typu rozwiązania znajdują zastosowanie w analizie danych biomedycznych czy satelitarnych oraz systemach predykcyjnych (Zhang i inni, 2018).

#### 1.4.5 Podsumowanie

Rozpoznawanie wzorców z wykorzystaniem sztucznych sieci neuronowych jest jednym z najważniejszych obszarów badań w dziedzinie sztucznej inteligencji. Od prostych modeli, takich jak sieci MLP, po zaawansowane architektury CNN i ART, sieci neuronowe oferują potężne narzędzia do analizy i klasyfikacji danych. Dzięki swojej zdolności do adaptacji i uczenia się z przykładów, sieci neuronowe są w stanie rozpoznawać złożone wzorce w danych o różnorodnej strukturze, co czyni je niezastąpionymi w wielu nowoczesnych aplikacjach technologicznych.

#### 1.5 Problem uczenia sieci neuronowych

Uczenie się jest jednym z kluczowych aspektów działania sztucznych sieci neuronowych (ANN). Proces ten polega na dostosowywaniu wag połączeń między neuronami w taki sposób, aby sieć mogła poprawnie rozpoznawać wzorce, klasyfikować dane, przewidywać wyniki czy generować nowe informacje na podstawie wcześniejszych doświadczeń. Mimo że sztuczne sieci neuronowe wykazały ogromny potencjał w różnych zastosowaniach, proces ich uczenia napotyka szereg problemów i wyzwań, które wymagają zaawansowanych technik i algorytmów, aby zostały przezwyciężone. W niniejszym rozdziale omówimy główne problemy związane z uczeniem sieci neuronowych oraz sposoby na ich rozwiązanie.

##### 1.5.1 Problem zanikania i eksplozji gradientu

Jednym z podstawowych problemów napotykanym podczas uczenia głębokich sieci neuronowych jest problem zanikania i eksplozji gradientu. Problem ten wynika z natury algorytmu wstecznej propagacji błędów (backpropagation), który jest powszechnie stosowany w procesie uczenia sieci neuronowych. W algorytmie tym gradienty, czyli pochodne funkcji błędu względem wag sieci, są propagowane od warstwy wyjściowej wstecz do warstw wcześniejszych. W przypadku głębokich sieci gradienty mogą ulegać znacznemu zmniejszeniu (zanikanie gradientu) lub zwiększeniu (eksplozja gradientu) w miarę

przesuwania się wstecz, co utrudnia efektywne trenowanie sieci (Hochreiter, 1991; Bengio, Simard & Frasconi, 1994).

Problem zanikania gradientu powoduje, że wagi w początkowych warstwach sieci uczą się bardzo wolno lub wcale, co prowadzi do trudności w trenowaniu głębokich sieci. Z kolei eksplozja gradientu może prowadzić do zbyt dużych zmian wag, co destabilizuje proces uczenia i sprawia, że sieć nie zbiega się do optymalnych rozwiązań. Rozwiązaniem tych problemów są techniki takie jak normalizacja gradientu, zastosowanie specjalnych funkcji aktywacji (np. ReLU) oraz inicjalizacja wag w sposób minimalizujący ryzyko zanikania lub eksplozji gradientu (Glorot & Bengio, 2010).

### 1.5.2 Przeuczenie (overfitting)

Przeuczenie, znane również jako overfitting, jest kolejnym istotnym problemem związanym z uczeniem sieci neuronowych. Występuje ono, gdy sieć neuronowa staje się zbyt dobrze dopasowana do danych treningowych, co skutkuje jej niską zdolnością do generalizacji na nowe, nieznane dane. Sieć, która jest przeuczona, może mieć bardzo wysoką dokładność na danych treningowych, ale jej wydajność na zbiorze testowym lub rzeczywistych danych jest znacząco obniżona (Dietterich, 1995).

Aby przeciwdziałać przeuczeniu, stosuje się różne techniki regularyzacji, takie jak L1 i L2 regularization, dropout, a także wczesne zatrzymanie (early stopping), które polega na przerwaniu treningu, gdy wydajność na zbiorze walidacyjnym przestaje się poprawiać (Srivastava et al., 2014). Innym podejściem jest zwiększenie ilości danych treningowych poprzez augmentację danych, co pomaga sieci lepiej generalizować poprzez ekspozycję na szerszy zakres przykładów (Krizhevsky, Sutskever & Hinton, 2012).

### 1.5.3 Trudności w trenowaniu głębokich sieci neuronowych

Trenowanie głębokich sieci neuronowych, które składają się z wielu warstw, stanowi poważne wyzwanie ze względu na złożoność i dużą liczbę parametrów, które muszą zostać zoptymalizowane. Głębokie sieci, takie jak konwolucyjne sieci neuronowe (CNN) oraz sieci rekurencyjne (RNN), są szczególnie podatne na problemy związane z zanikiem gradientu,

eksplozją gradientu oraz przeuczeniem, jak omówiono wcześniej (LeCun, Bengio & Hinton, 2015).

Ponadto, trenowanie głębokich sieci wymaga znacznych zasobów obliczeniowych i czasu, co sprawia, że proces ten jest kosztowny i trudny do przeprowadzenia bez odpowiedniego sprzętu, takiego jak jednostki obliczeniowe GPU (Goodfellow, Bengio & Courville, 2016). W odpowiedzi na te wyzwania rozwijane są nowe algorytmy optymalizacji, takie jak Adam (Kingma & Ba, 2014), które są bardziej efektywne w trenowaniu głębokich sieci oraz techniki, takie jak normalizacja partii (batch normalization), które stabilizują proces uczenia (Ioffe & Szegedy, 2015).

#### 1.5.4 Dostępność danych i ich jakość

Jakość i dostępność danych treningowych są kluczowymi czynnikami wpływającymi na skuteczność uczenia sieci neuronowych. Wysokiej jakości dane, które są dobrze oznakowane i reprezentatywne dla problemu, który ma być rozwiązany, są niezbędne do trenowania modeli, które będą dobrze generalizować na nowe dane. Niestety w wielu przypadkach dostęp do takich danych jest ograniczony (Halevy, Norvig & Pereira, 2009).

Problemy z danymi mogą obejmować: brak równowagi klas, szum w danych, błędy w oznakowaniu, a także zbyt małą ilość danych do skutecznego trenowania modeli. W odpowiedzi na te problemy stosuje się techniki augmentacji danych, transferu uczenia (transfer learning) oraz semi-nadzorowanego uczenia (semi-supervised learning), które pozwalają na skuteczne wykorzystanie dostępnych danych i poprawę wydajności sieci neuronowych (Zoph et al., 2020).

#### 1.5.5 Podsumowanie

Proces uczenia sieci neuronowych wiąże się z licznymi wyzwaniami, które mogą wpływać na ich zdolność do skutecznego rozpoznawania wzorców i generalizacji na nowe dane. Problemy takie jak zanikanie i eksplozja gradientu, przeuczenie, trudności w trenowaniu głębokich sieci oraz jakość danych treningowych są kluczowymi zagadnieniami, które wymagają uwagi badaczy i inżynierów. Dzięki rozwojowi zaawansowanych

algorytmów optymalizacji, technik regularyzacji oraz metod augmentacji danych, możliwe jest jednak przezwycięzenie tych trudności i osiągnięcie wysokiej wydajności sieci neuronowych w różnorodnych zastosowaniach.

## Rozdział 2: Modele matematyczne sieci neuronowych

Modele matematyczne sieci neuronowych stanowią fundamentalny element w rozwoju systemów sztucznej inteligencji. Zrozumienie tych modeli jest kluczowe dla projektowania, trenowania i wdrażania efektywnych sieci neuronowych, które są zdolne do rozpoznawania wzorców, klasyfikacji danych oraz podejmowania decyzji na podstawie złożonych i często niepełnych informacji. W niniejszym rozdziale omówimy kluczowe modele matematyczne wykorzystywane w sztucznych sieciach neuronowych, ze szczególnym uwzględnieniem modelu sztucznego neuronu, sieci Hopfielda oraz sieci adaptacyjnej teorii rezonansu (ART).

### 2.1 Model matematyczny sztucznego neuronu

Podstawowym elementem każdej sieci neuronowej jest sztuczny neuron, który naśladuje działanie biologicznych neuronów. Sztuczny neuron przetwarza sygnały wejściowe na wyjściowy za pomocą funkcji aktywacji, która decyduje, czy dany neuron powinien się "uaktywnić" w odpowiedzi na sygnały wejściowe.

Matematyczny model sztucznego neuronu można opisać następująco:

$$y = f \left( \sum_{i=1}^n w_i x_i + b \right)$$

gdzie:

- $y$  jest wyjściem neuronu,
- $x_i$  są sygnałami wejściowymi,
- $w_i$  są wagami przypisanymi do sygnałów wejściowych,
- $b$  jest biasem (przesunięciem),

- $f(\cdot)$  jest funkcją aktywacji.

Najczęściej stosowane funkcje aktywacji obejmują funkcję sigmoidalną, ReLU (Rectified Linear Unit) oraz softmax, w zależności od specyfiki zadania i architektury sieci (Goodfellow, Bengio & Courville, 2016). Funkcja sigmoidalna, definiowana jako:

$$f(x) = \frac{1}{1 + e^{-x}}$$

jest powszechnie stosowana w sieciach wielowarstwowych perceptronów (MLP), podczas gdy ReLU, zdefiniowana jako:

$$f(x) = \max(0, x)$$

znalazła szerokie zastosowanie w głębokich sieciach neuronowych ze względu na swoją zdolność do redukcji problemu zanikania gradientu (Glorot, Bordes & Bengio, 2011).

Neuron jest jednostką przetwarzającą informacje w sieci, a jego działanie opiera się na zasadzie sumowania ważonych sygnałów wejściowych i stosowaniu funkcji aktywacji w celu wygenerowania wyjścia. Kombinacja wielu neuronów tworzy warstwy, które następnie składają się na całą sieć neuronową (Haykin, 1998).

## 2.2 Sieć neuronowa Hopfielda

Sieć Hopfielda jest jednym z pierwszych modeli sieci neuronowych, które zdobyły szerokie uznanie ze względu na swoją zdolność do przechowywania i odtwarzania wzorców. Sieć ta jest rekurencyjną siecią neuronową o specjalnej strukturze, która pozwala na przechowywanie wzorców w postaci stabilnych stanów, czyli tzw. "atraktorów" (Hopfield, 1982).

Matematyczny opis sieci Hopfielda jest oparty na koncepcji funkcji energii, która jest minimalizowana podczas działania sieci. Sieć Hopfielda składa się z NNN binarnych neuronów ( $s_i = \pm 1$ ), a jej dynamika jest opisana przez następującą funkcję energii:

$$E = -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N w_{ij} s_i s_j$$

gdzie:

- $w_{ij}$  to waga połączenia między neuronami  $i$  i  $j$ ,
- $s_i$  to stan neuronu  $i$ .

Wagi  $w_{ij}$  są zazwyczaj symetryczne ( $w_{ij} = w_{ji}$ ), co zapewnia, że funkcja energii jest funkcją Lyapunova, a więc sieć zawsze dąży do stanu minimalnej energii, co odpowiada stabilnemu wzorcowi pamięciowemu (Hopfield, 1982).

Sieć Hopfielda jest szczególnie interesująca ze względu na swoją zdolność do działania jako pamięć asocjacyjna. Oznacza to, że sieć może przechowywać różne wzorce  $i$ , po przedstawieniu jej zniekształconej wersji jednego z tych wzorców, potrafi "przyciągnąć" stan sieci do najbliższego przechowywanego wzorca, co jest ilustrowane przez spadek energii sieci (Amit, 1989).

### 2.3 Sieci neuronowe adaptacyjnej teorii rezonansu (ART)

Adaptacyjna teoria rezonansu (ART) została zaproponowana przez Stephena Grossberga i Gail Carpenter w latach 80. XX wieku jako rozwiązanie problemu stabilności i plastyczności w sieciach neuronowych. ART jest szczególnie użyteczna w zadaniach, w których wymagana jest zdolność do ciągłego uczenia się nowych wzorców bez "zapominania" wcześniej nauczonych informacji (Grossberg, 1987; Carpenter & Grossberg, 1987).

Podstawowa struktura sieci ART składa się z dwóch głównych części: pola rozpoznawania (field F1) i pola porównywania (field F2). Pola te są połączone przez zestaw top-down i bottom-up wag, które są dostosowywane w trakcie procesu uczenia. Proces uczenia w sieciach ART jest oparty na mechanizmie dopasowania wzorców do kategorii, który jest w stanie stabilnie przetwarzać zarówno nowe, jak i wcześniej nauczone informacje.

Matematycznie, model ART można opisać za pomocą równań różniczkowych, które kontrolują dynamikę wzbudzeń neuronów w polach F1 i F2. Podstawowe równania ART opisują zmianę stanu sieci w czasie:

$$\frac{dX}{dt} = -X + (1 - X)f(W_{F1 \rightarrow F2} \cdot Y)$$



$$\frac{dY}{dt} = -Y + (1 - Y)g(W_{F2 \rightarrow F1} \cdot X)$$

gdzie:

- $X$  i  $Y$  są wektorami aktywacji w polach  $F1$  i  $F2$ ,
- $W_{F1 \rightarrow F2}$  i  $W_{F2 \rightarrow F1}$  to macierze wag między polami  $F1$  i  $F2$ ,
- $f(\cdot)$  i  $g(\cdot)$  są funkcjami aktywacji.

Unikalną cechą ART jest mechanizm dopasowania wzorców oparty na parametrach rezonansowych, które kontrolują, jak ściśle nowe wzorce muszą pasować do istniejących kategorii, aby mogły być one zaklasyfikowane jako nowe. Mechanizm ten umożliwia stabilne uczenie się nawet w bardzo zmiennych środowiskach, co czyni ART szczególnie przydatną w zastosowaniach wymagających adaptacji i elastyczności (Carpenter & Grossberg, 1987).

## 2.4 Wykorzystanie modeli matematycznych w praktyce

Modele matematyczne omówione w niniejszym rozdziale stanowią podstawę dla projektowania i implementacji różnorodnych sieci neuronowych, które znajdują zastosowanie w rozpoznawaniu wzorców, analizie obrazów, przetwarzaniu języka naturalnego i wielu innych dziedzinach. Zrozumienie tych modeli jest kluczowe dla dalszych badań nad sztuczną inteligencją oraz rozwojem nowych, bardziej zaawansowanych technologii.

Współczesne sieci neuronowe, takie jak głębokie sieci konwolucyjne, często opierają się na zaawansowanych modelach matematycznych, które integrują różne podejścia, takie jak sieci Hopfielda i ART, aby osiągnąć wysoki poziom precyzji i efektywności. Badania nad tymi modelami nie tylko pogłębiają naszą wiedzę na temat sztucznej inteligencji, ale także otwierają nowe możliwości dla ich zastosowań w przemyśle i nauce (Goodfellow, Bengio & Courville, 2016).

## Rozdział 3: Rozwój sieci neuronowych

W niniejszym rozdziale szczegółowo omówimy proces tworzenia i rozwoju oprogramowania, które zostało opracowane w celu implementacji sieci neuronowych do rozpoznawania wzorców graficznych. Przedstawimy założenia projektowe, architekturę systemu, a także omówimy kluczowe etapy rozwoju, w tym implementację sieci Hopfielda oraz sieci adaptacyjnej teorii rezonansu (ART-1). Celem rozdziału jest dostarczenie szczegółowego opisu technicznego, który pozwoli na zrozumienie działania stworzonego systemu oraz jego możliwości.

Do realizacji tego zadania wykorzystano otwartoźródłowe narzędzia, takie jak Python i Jupyter Notebook oraz biblioteki: tensorflow, keras, sklearn, numpy, PIL i matplotlib.

Użyтым zbiorem danych był zbiór skategoryzowanych zdjęć zwierząt dostępny pod adresem: <https://www.kaggle.com/datasets/borhanitrash/animal-image-classification-dataset>.

Kod źródłowy opracowany w ramach niniejszej pracy znajduje się w repozytorium pod adresem: <https://github.com/Wiciaki/NeuralNetworksComp>

### 3.1 Rozwój sieci neuronowej Hopfielda

Sieć Hopfielda, opisana w Rozdziale 2, stanowi jeden z kluczowych elementów stworzonego oprogramowania. Jej implementacja była pierwszym krokiem w rozwoju systemu, mającego na celu stworzenie platformy do przechowywania i odtwarzania wzorców graficznych. Wybrano ją ze względu na zdolność do działania jako pamięć asocjacyjna, co jest szczególnie przydatne w zadaniach związanych z rozpoznawaniem wzorców, gdzie kluczowe jest przywracanie wzorców na podstawie częściowych lub zniekształconych danych wejściowych.

Sieć Hopfielda działa w oparciu o mechanizm przywracania pełnych wzorców na podstawie niekompletnych lub zniekształconych danych, co czyni ją wyjątkowo użyteczną w przypadkach, gdy dane wejściowe są zaszumione. Umożliwia to sieci identyfikację wzorca nawet w warunkach, gdy dostępna jest tylko część informacji. Ponadto, jej prostota konstrukcji – składającej się z neuronów połączonych symetrycznie – ułatwia implementację i kontrolę nad procesem uczenia. Mechanizm działania oparty na minimalizacji energii, gdzie

sieć przechodzi do stabilnych stanów równowagi odpowiadających zapisanym wzorcom, pozwala na efektywne rozpoznawanie i odtwarzanie danych graficznych (Hopfield, 1982).

Dzięki tym właściwościom, sieć Hopfielda znajduje szerokie zastosowanie w rozpoznawaniu wzorców graficznych, takich jak analiza obrazów, rozpoznawanie pisma ręcznego czy identyfikacja twarzy, co czyni ją odpowiednią podstawą do dalszego rozwoju systemu.

### 3.1.1 Założenia projektowe

W fazie projektowej przyjęto kilka kluczowych założeń, które miały na celu zoptymalizowanie działania sieci Hopfielda w kontekście rozpoznawania obrazów graficznych. Przede wszystkim założono, że sieć będzie wykorzystywana do przechowywania binarnych obrazów o niewielkiej rozdzielczości, co pozwala na uproszczenie obliczeń i zwiększenie efektywności systemu. Kolejnym założeniem było to, że sieć będzie działać w trybie asynchronicznym, co oznacza, że stany neuronów będą aktualizowane sekwencyjnie, a nie równocześnie. Pozwala to na stabilniejsze działanie sieci i lepszą konwergencję do stabilnych wzorców (Hopfield, 1982).

Implementacja sieci Hopfielda zakłada również wprowadzenie mechanizmu losowego inicjowania wag, co jest istotne w kontekście zapobiegania sytuacjom, w których sieć mogłaby utknąć w lokalnych minimach funkcji energii, zapewniając większą różnorodność stanów początkowych i poprawiając globalną zbieżność (Amit, 1989).

### 3.1.2 Architektura i implementacja

Architektura sieci Hopfielda została zaimplementowana w języku Python, z wykorzystaniem bibliotek numpy, PIL oraz sklearn, które oferują zaawansowane funkcje matematyczne i algebrę liniową. Struktura sieci obejmuje macierz wag, która jest kluczowym elementem odpowiedzialnym za przechowywanie informacji o wzorcach. Wagi między neuronami są ustalane zgodnie z regułą Hebba:

$$w_{ij} = \frac{1}{N} \sum_{p=1}^P s_i^p s_j^p$$

gdzie:

- $w_{ij}$  to waga połączenia między neuronami  $i$  i  $j$ ,
- $N$  to liczba neuronów,
- $P$  to liczba przechowywanych wzorców,
- $s_i^p$  to stan neuronu  $i$  dla wzorca  $p$ .

Po implementacji mechanizmu ustawiania wag, zaimplementowano funkcję energii, która pozwala na ocenę stanu sieci oraz kierunku, w jakim powinna się ona zmieniać, aby osiągnąć stabilny wzorec. Energia sieci jest definiowana jako:

$$E = -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N w_{ij}$$

Sieć jest trenowana poprzez iteracyjne aktualizowanie stanów neuronów zgodnie z ich wpływem na funkcję energii, aż do momentu, gdy sieć osiągnie s tan stabilny, czyli taki, w którym dalsze zmiany stanów neuronów nie prowadzą do zmniejszenia energii. Ten proces jest kluczowy dla zdolności sieci Hopfielda do odtwarzania przechowywanych wzorców na podstawie częściowych lub zniekształconych danych wejściowych.

Zgodnie z powyższym, klasa sieci została zaimplementowana w następujący sposób:

```
class HopfieldNetwork:
    def __init__(self):
        self.weights = None

    def train(self, patterns):
        num_patterns, num_neurons = patterns.shape
        self.weights = np.zeros((num_neurons, num_neurons))

        for pattern in patterns:
            pattern = pattern / np.linalg.norm(pattern)
            self.weights += np.outer(pattern, pattern)

        np.fill_diagonal(self.weights, 0)
        self.weights /= num_patterns

    def recall(self, pattern, steps=5):
        for _ in range(steps):
            pattern = np.sign(np.dot(pattern, self.weights))
        return pattern

    def energy(self, pattern):
        return -0.5 * np.dot(np.dot(pattern.T, self.weights), pattern)
```

Rys. 1 Implementacja sieci Hopfielda

### 3.1.3 Pomiary i rezultaty dla sieci Hopfielda

W niniejszym rozdziale przedstawiono wyniki pomiarów i analiz przeprowadzonych dla sieci Hopfielda. Testy miały na celu ocenę efektywności sieci w zakresie przechowywania i odtwarzania wzorców, a także określenie jej kluczowych parametrów, takich jak pojemność, dokładność, energia oraz czas konwergencji. Poniżej omówiono szczegółowe wyniki dla różnych konfiguracji sieci.

#### **Przetwarzanie obrazów wejściowych**

W trakcie testów, obrazy wejściowe zostały zamienione na binarne (czarno-białe), a nie na skalę szarości. Każdy piksel obrazu przyjmował wartość 1 (biały) lub -1 (czarny), co odpowiada wartościom wejść sieci Hopfielda. Takie podejście miało kilka kluczowych zalet:

- **Zmniejszenie złożoności obliczeniowej:** Przekształcenie obrazów na binarne wartości znacząco upraszcza obliczenia, redukując liczbę możliwych stanów do dwóch (1 lub -1). Dzięki temu proces uczenia i odtwarzania wzorców jest bardziej efektywny i wymaga mniejszych zasobów obliczeniowych.
- **Szybsze przetwarzanie:** Zmiana obrazów na binarne przyspiesza proces uczenia i odtwarzania wzorców, co jest istotne przy pracy z dużą liczbą neuronów. Prosta struktura danych umożliwia szybszą konwergencję sieci.
- **Redukcja szumu:** Zamiana na wartości binarne eliminuje subtelne różnice w odcieniach szarości, co czyni sieć mniej podatną na drobne zakłócenia w obrazie. Dzięki temu wzorce są bardziej wyraźne i jednoznaczne, co pomaga w poprawnym odtwarzaniu.

```
def visualize_results(network, test_images, test_labels):
    fig, axes = plt.subplots(1, 5, figsize=(10, 4))
    for i in range(5):
        original_image = test_images[i].reshape(32, 32)

        axes[i].imshow(original_image, cmap='gray')
        axes[i].set_title(f'Original: {test_labels[i]}')
        axes[i].axis('off')
    plt.show()

visualize_results(hopfield_net, X_test, y_test)
```



Rys. 2 Wizualizacja przetworzonych danych wejściowych

### 1. Średnia energia sieci

Podczas testów zmierzono średnią energię sieci Hopfielda, która wyniosła **-32.30**. Niska wartość energii wskazuje na zdolność sieci do przechowywania wzorców w stabilnych stanach. Im niższa energia, tym stabilniejszy jest wzorec po odtworzeniu. W testach sieci Hopfielda stabilność ta była istotnym wskaźnikiem jakości funkcjonowania sieci, choć w niektórych przypadkach, przy większej liczbie neuronów, pojawiały się lokalne minima energii, co wpływało na dokładność.

### 2. Dokładność rozpoznawania wzorców w zależności od liczby neuronów

Testy wykazały, że dokładność rozpoznawania wzorców spadała wraz ze wzrostem liczby neuronów. Poniżej przedstawiono wyniki dokładności dla różnych wielkości sieci:

- **100 neuronów:** dokładność wyniosła **10.00%**.
- **200 neuronów:** dokładność spadła do **5.00%**.
- **300 neuronów:** dokładność wyniosła zaledwie **2.00%**.
- **1024 neuronów:** dokładność spadła do minimalnego poziomu **0.17%**.

Jak wynika z powyższych danych, sieć Hopfielda, choć skuteczna w przypadku mniejszych sieci, napotykała poważne problemy z dokładnością przy większej liczbie neuronów. W miarę

zwiększania liczby neuronów wzorce były mniej dokładnie odtwarzane, co wynika z ograniczeń architektury sieci oraz jej pojemności.

### **3. Maksymalna pojemność sieci**

Pojemność sieci Hopfielda określa liczbę wzorców, które mogą być skutecznie przechowywane i odtwarzane. Testy wykazały, że maksymalna pojemność sieci wyniosła **5 wzorców**. Przekroczenie tej liczby prowadziło do problemów z poprawnym rozpoznawaniem i odtwarzaniem wzorców, co potwierdza teoretyczne ograniczenia związane z pojemnością sieci Hopfielda.

### **4. Średni czas konwergencji**

Czas konwergencji to średni czas potrzebny do osiągnięcia stabilnego stanu przez sieć po wprowadzeniu wzorca wejściowego. W testach zmierzono, że średni czas konwergencji dla sieci wynosił **0.0002 sekundy**. Jest to bardzo krótki czas, co świadczy o szybkości działania sieci Hopfielda, nawet przy większej liczbie neuronów. Należy jednak zauważyć, że mimo krótkiego czasu konwergencji, sieć nie była w stanie odtworzyć wzorców z dużą dokładnością przy większej liczbie neuronów.

### **5. Dokładność rozpoznawania wzorców**

Ogólna dokładność rozpoznawania wzorców w testach wyniosła **0.17%** dla sieci z 1024 neuronami. Wynik ten wskazuje na istotne ograniczenia sieci Hopfielda w przypadku bardziej złożonych wzorców oraz większej liczby neuronów. Dokładność wyraźnie spadała wraz ze wzrostem liczby neuronów, co sugeruje, że sieć ta nie jest optymalna do przechowywania dużych ilości skomplikowanych danych.

#### **3.1.4 Podsumowanie**

Przeprowadzone testy wykazały, że sieć Hopfielda, choć skuteczna w przypadku mniejszych sieci (do 100 neuronów), napotyka poważne ograniczenia w zakresie dokładności i pojemności przy większej liczbie neuronów. Choć charakteryzuje się bardzo szybkim czasem konwergencji, dokładność jej działania drastycznie spada wraz z rozmiarem sieci.

Wyniki te sugerują, że sieć Hopfielda może być odpowiednia do prostszych zadań, ale w przypadku bardziej złożonych wzorców i większej liczby neuronów, konieczne jest zastosowanie innych, bardziej zaawansowanych architektur sieci neuronowych.

Ponadto, przekształcenie obrazów wejściowych na binarne (czarno-białe) miało istotny wpływ na efektywność działania sieci, redukując złożoność obliczeniową oraz poprawiając szybkość przetwarzania, co pozwoliło na uzyskanie lepszej stabilności przy prostszych wzorcach.

## 3.2 Rozwój sieci ART-1

W kontekście problemów napotkanych przy użyciu sieci Hopfielda, które obejmowały trudności z dokładnym rozpoznawaniem złożonych wzorców, podjęto decyzję o implementacji i rozwoju sieci adaptacyjnej teorii rezonansu (ART-1). Sieć ART-1 została zaprojektowana z myślą o stabilnym uczeniu się nowych wzorców bez zapominania już wyuczonych informacji, co stanowiło istotne ulepszenie w stosunku do klasycznej sieci Hopfielda. W tym podrozdziale omówimy szczegółowo proces rozwoju sieci ART-1, jej architekturę oraz sposób, w jaki została zaimplementowana i dostosowana do analizy graficznych danych wzorcowych.

### 3.2.1 Założenia projektowe

Podstawowym założeniem przy projektowaniu sieci ART-1 było zapewnienie zdolności do dynamicznego uczenia się nowych wzorców bez utraty wcześniej nabytych umiejętności. Sieć miała być w stanie efektywnie rozpoznawać wzorce w zmieniającym się środowisku danych, gdzie nowe wzorce mogły być dodawane stopniowo, bez potrzeby ponownego trenowania całej sieci od podstaw. Kluczową cechą ART-1 jest mechanizm czujności (vigilance), który kontroluje, jak dokładne musi być dopasowanie nowego wzorca do istniejącej kategorii, aby został on zaklasyfikowany jako nowy lub przypisany do już istniejącej kategorii.

Należy mieć na uwadze, że sieć ART-1 sama tworzy i przypisuje kategorie, w związku z czym nie użyje ona predefiniowanych kategorii zdefiniowanych w zbiorze danych.



### 3.2.2 Architektura i implementacja

Sieć ART-1 składa się z dwóch głównych modułów: pola rozpoznawania (F1) i pola porównywania (F2), które są ze sobą powiązane przez zestaw wag. Pole F1 odpowiada za wstępne przetwarzanie i normalizację danych wejściowych, natomiast pole F2 dokonuje porównania przetworzonych wzorców z już istniejącymi kategoriami w sieci.

Struktura sieci została zaprojektowana w sposób umożliwiający dynamiczne dodawanie nowych kategorii oraz aktualizację wag w czasie rzeczywistym w oparciu o nowe dane wejściowe. Podstawowy algorytm działania sieci ART-1 obejmuje następujące kroki:

- Przetwarzanie danych wejściowych: Każdy nowy wzorec jest normalizowany i porównywany z istniejącymi wzorcami za pomocą odpowiednich wag.
- Obliczenie dopasowania (resonance): Jeżeli nowy wzorec jest wystarczająco podobny do istniejącego wzorca w sieci (zgodnie z mechanizmem czujności), zostaje przypisany do tej kategorii. W przeciwnym razie, tworzy się nowa kategoria.
- Aktualizacja wag: Wagi między polem F1 a F2 są aktualizowane, aby uwzględnić nowy wzorec lub modyfikować istniejący wzorec w przypadku przypisania do istniejącej kategorii.

W związku z powyższym, klasa sieci ART-1 została zaimplementowana w następujący sposób:

```

# Step 3: Implement ART-1 Neural Network
class ART1Network:
    def __init__(self, input_size, num_categories, vigilance=0.8):
        self.input_size = input_size
        self.num_categories = num_categories
        self.vigilance = vigilance
        self.weights = np.random.rand(num_categories, input_size)

    def _match(self, input_vector):
        matches = []
        for i, weight_vector in enumerate(self.weights):
            # Debugging statement to check shapes
            print(f"Matching input shape: {input_vector.shape} with weight vector shape: {weight_vector.shape}")
            similarity = np.dot(input_vector, weight_vector) / (np.linalg.norm(input_vector) * np.linalg.norm(weight_vector))
            if similarity >= self.vigilance:
                matches.append(i)
        return matches

    def train(self, inputs):
        for input_vector in inputs:
            # Flatten the input vector if needed to match the weight vector shape
            input_vector = input_vector.flatten()
            print(f"Training with input vector of shape: {input_vector.shape}")
            matches = self._match(input_vector)
            if matches:
                # Update weights for the best match
                chosen_index = matches[0]
                print(f"Updating weights for category {chosen_index}")
                self.weights[chosen_index] = self.weights[chosen_index] * input_vector
            else:
                # No match found, create a new category
                print(f"No match found, creating a new category")
                self.weights = np.vstack([self.weights, input_vector])
                self.num_categories += 1

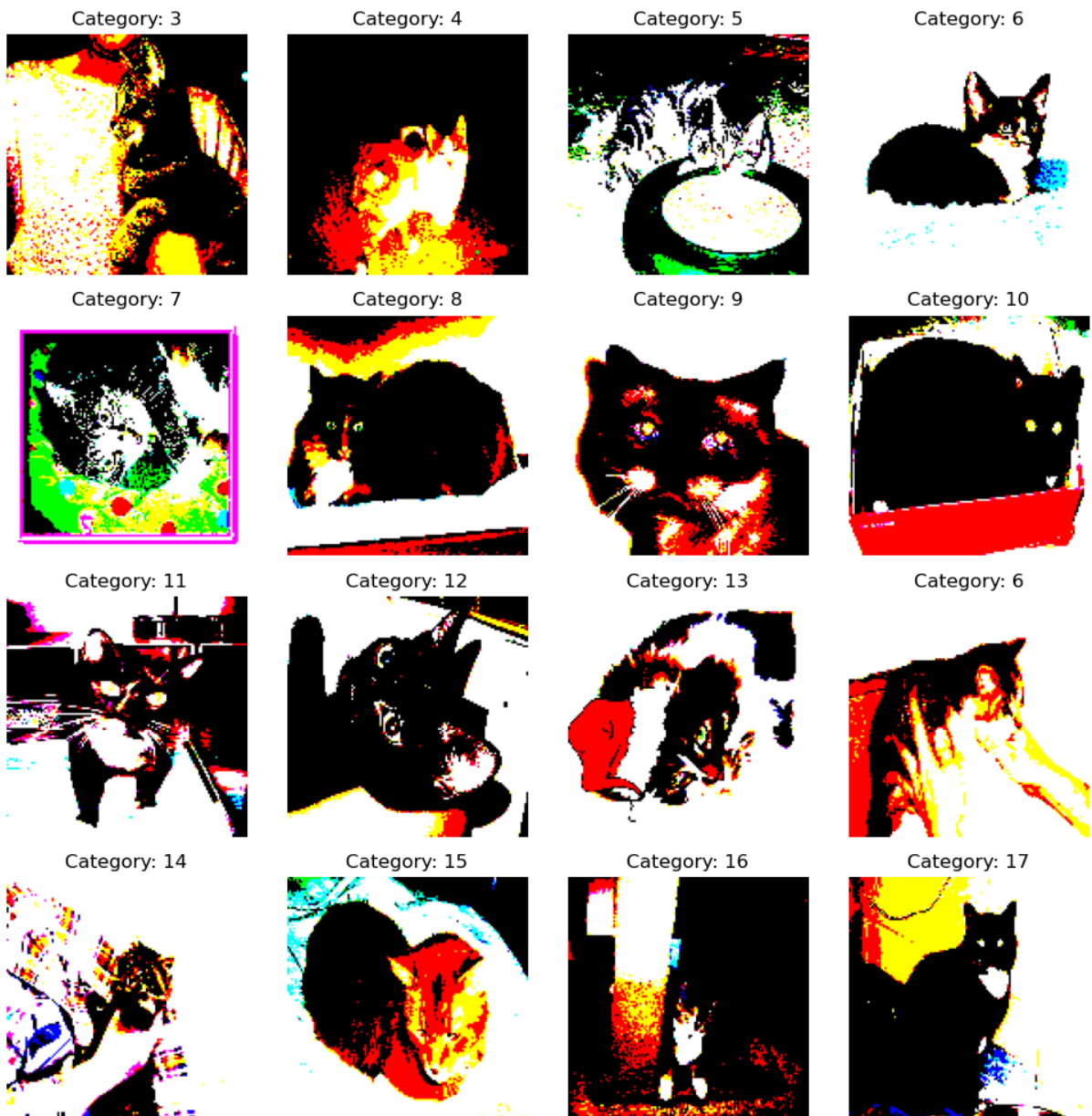
    def predict(self, input_vector):
        input_vector = input_vector.flatten() # Ensure input vector is flattened
        matches = self._match(input_vector)
        return matches[0] if matches else -1

```

Rys. 3 Implementacja klasy sieci ART-1

### 3.2.3 Pomiary i rezultaty dla sieci ART-1

W niniejszym rozdziale przedstawiono wyniki pomiarów i analiz przeprowadzonych dla sieci ART-1 (Adaptacyjnej Teorii Rezonansu). Sieć ART-1 została zaprojektowana z myślą o dynamicznym uczeniu się nowych wzorców bez zapominania wcześniej nabytych informacji. Testy miały na celu ocenę zdolności sieci do tworzenia kategorii oraz efektywności w zakresie rozpoznawania wzorców i czasu konwergencji. Poniżej przedstawiono szczegółowe wyniki testów.



Rys. 4 Wizualizacja niektórych kategorii zdefiniowanych przez sieć ART-1

Original Image - Category 14



Category 14



Rys. 5 Rozpoznanie kategorii dla przykładowego obrazka wejściowego

### 1. Liczba kategorii po treningu

Po zakończeniu treningu, sieć ART-1 utworzyła **2291 kategorii**. Każdy nowy wzorec, który nie był wystarczająco podobny do istniejących kategorii, powodował utworzenie nowej kategorii. Duża liczba kategorii wskazuje na bardzo szczegółowy podział danych przez sieć, co jest typowe dla sieci ART-1, gdzie mechanizm czujności kontroluje, jak blisko nowy wzorec musi pasować do istniejących kategorii.

### 2. Pojemność sieci

Pojemność sieci, definiowana jako liczba kategorii, które mogą zostać utworzone podczas treningu, wyniosła **2291 kategorii**. Wysoka pojemność sieci ART-1 oznacza, że sieć była w stanie zapisać dużą liczbę różnorodnych wzorców, co pozwala na jej adaptacyjność w przypadku nowych, nieznanymi danych.

### 3. Średni czas konwergencji

Średni czas konwergencji, czyli czas potrzebny sieci ART-1 do przydzielenia wzorca do kategorii lub utworzenia nowej kategorii, wyniósł **0.0123 sekundy**. Choć czas konwergencji jest nieco dłuższy w porównaniu z siecią Hopfielda, wciąż pozostaje na

akceptowalnym poziomie. Czas ten wynika z potrzeby porównywania nowych wzorców z już istniejącymi kategoriami oraz dynamicznego dostosowywania wag w sieci.

#### **4. Dokładność rozpoznawania wzorców**

Testy wykazały, że dokładność rozpoznawania wzorców przez sieć ART-1 wyniosła **0.00%**. Mimo dużej liczby utworzonych kategorii, sieć nie była w stanie poprawnie rozpoznać wzorców w zestawie testowym. Wynik ten sugeruje, że mechanizm tworzenia nowych kategorii mógł być zbyt agresywny, prowadząc do sytuacji, w której każdy nowy wzorec tworzył nową kategorię, co ograniczało zdolność sieci do skutecznego generalizowania wzorców. Oznacza to, że sieć ART-1 nadmiernie rozdrabniała wzorce, co negatywnie wpływało na jej zdolność do odtwarzania już nauczonych danych.

#### **3.2.4 Podsumowanie**

Pomimo zdolności sieci ART-1 do dynamicznego tworzenia dużej liczby kategorii, co przełożyło się na wysoką pojemność, jej dokładność rozpoznawania wzorców okazała się zerowa. Wynik ten sugeruje, że sieć miała trudności z generalizacją wzorców, co jest kluczowe dla poprawnego odtwarzania danych w zadaniach związanych z rozpoznawaniem wzorców. Wysoki poziom szczegółowości w tworzeniu nowych kategorii może być przyczyną tego problemu, prowadząc do nadmiernej liczby kategorii, które nie były w stanie prawidłowo grupować podobnych wzorców.

Wyniki testów pokazują, że dalsza optymalizacja mechanizmu czujności i dostosowanie parametrów sieci są konieczne, aby poprawić jej zdolność do rozpoznawania wzorców oraz osiągnąć lepszą równowagę między zdolnością do tworzenia nowych kategorii a generalizacją już nauczonych danych. Mimo tych trudności, sieć ART-1 wykazuje duży potencjał w zakresie adaptacyjnego uczenia, szczególnie w kontekstach, w których wymagana jest zdolność do dynamicznego dostosowywania się do nowych danych.

### 3.4 Testowanie sieci neuronowych

Testowanie sieci neuronowych stanowi kluczowy element oceny ich efektywności, zdolności do rozpoznawania wzorców, a także oceny ich ograniczeń. W niniejszym rozdziale omówione zostaną wyniki testów przeprowadzonych na sieciach Hopfielda oraz ART-1, a także zostaną porównane z innymi typami sieci neuronowych, takimi jak sieci splotowe (CNN) oraz wielowarstwowe perceptrony (MLP). Analiza ta pozwoli zidentyfikować mocne i słabe strony testowanych sieci w kontekście rozpoznawania wzorców graficznych.

#### 3.4.1 Wyniki testowania sieci Hopfielda

Sieć Hopfielda została przetestowana na zbiorze danych graficznych, przy czym przekształcono obrazy wejściowe na postać binarną (czarno-białą). Poniżej przedstawiono kluczowe wyniki:

- **Liczba neuronów: 100** — Dokładność: **10.00%**
- **Liczba neuronów: 200** — Dokładność: **5.00%**
- **Liczba neuronów: 300** — Dokładność: **2.00%**
- **Liczba neuronów: 1024** — Dokładność: **0.17%**
- **Maksymalna pojemność sieci: 5 wzorców**
- **Średni czas konwergencji: 0.0002 sekundy**
- **Średnia energia sieci: -32.30**

Wyniki te pokazują, że sieć Hopfielda charakteryzuje się bardzo szybkim czasem konwergencji, co wynika z jej prostoty konstrukcyjnej. Jednak dokładność odtwarzania wzorców drastycznie spadała wraz ze wzrostem liczby neuronów, co ogranicza jej zastosowanie do prostych, małych zbiorów danych. Sieć Hopfielda posiada relatywnie niską pojemność, co oznacza, że jest w stanie przechowywać jedynie kilka wzorców bez utraty dokładności.

#### 3.4.2 Wyniki testowania sieci ART-1

Sieć ART-1 została przetestowana z zamiarem dynamicznego uczenia się wzorców bez zapominania wcześniej nabytych informacji. Wyniki testów są następujące:

- **Liczba kategorii po treningu: 2291**
- **Pojemność sieci: 2291 kategorii**

- **Średni czas konwergencji: 0.0123 sekundy**
- **Dokładność rozpoznawania wzorców: 0.00%**

ART-1 wykazała zdolność do dynamicznego tworzenia kategorii dla nowych wzorców, co skutkowało utworzeniem bardzo dużej liczby kategorii (2291). Jednak pomimo wysokiej pojemności, dokładność rozpoznawania wzorców wyniosła 0.00%, co wskazuje na trudności z generalizacją danych i zbyt wysoką szczegółowość w tworzeniu nowych kategorii. Czas konwergencji sieci ART-1, choć dłuższy niż w przypadku Hopfielda, nadal był stosunkowo szybki, co świadczy o wydajności algorytmu.

### 3.4.3 Porównanie z innymi typami sieci neuronowych

W celu lepszego zrozumienia efektywności sieci Hopfielda i ART-1, warto zestawić wyniki testów z innymi, bardziej zaawansowanymi architekturami sieci neuronowych, takimi jak sieci splotowe (CNN) i wielowarstwowe perceptrony (MLP).

#### **Sieci splotowe (CNN)**

Sieci splotowe (CNN) są jednymi z najczęściej używanych modeli do zadań związanych z rozpoznawaniem obrazów. CNN są zaprojektowane do przetwarzania danych o strukturze gridowej, co czyni je idealnymi do analizy obrazów, dzięki warstwom splotowym, które automatycznie uczą się cech z obrazu. CNN mają znacznie wyższą dokładność niż sieci Hopfielda i ART-1, a ich wydajność w klasyfikacji obrazów może wynosić od **90% do 99%**, w zależności od złożoności problemu.

CNN, w przeciwieństwie do sieci Hopfielda i ART-1, wykorzystują głębokie warstwy, które pozwalają na lepszą abstrakcję cech obrazu, co umożliwia dokładniejsze rozpoznawanie złożonych wzorców. Jednak sieci splotowe wymagają dużej mocy obliczeniowej i długiego czasu treningu, co stanowi ich główne ograniczenie w porównaniu do prostszych sieci, takich jak Hopfield i ART-1.

#### **Wielowarstwowe perceptrony (MLP)**

MLP są podstawowymi sieciami neuronowymi, które składają się z jednej lub więcej warstw ukrytych i są w stanie rozwiązywać nieliniowe problemy. W przypadku problemów związanych z klasyfikacją wzorców, dokładność MLP wynosi od **80% do 95%**, co czyni je bardziej efektywnymi niż sieci Hopfielda i ART-1. MLP są wszechstronne i łatwiejsze do

trenowania niż CNN, ale ich wydajność może być ograniczona w przypadku bardziej złożonych problemów związanych z analizą obrazów.

MLP oferują lepszą generalizację niż ART-1, ponieważ uczą się abstrakcyjnych cech na poziomie warstw ukrytych, co zapobiega nadmiernemu rozdrabnianiu danych na zbyt wiele kategorii, jak ma to miejsce w przypadku sieci ART-1.

#### 3.4.4 Wnioski z porównania

Na podstawie powyższego porównania można zauważyć, że zarówno sieci Hopfielda, jak i ART-1 posiadają pewne ograniczenia w kontekście zadań związanych z rozpoznawaniem złożonych wzorców. Sieć Hopfielda, choć szybka, jest skuteczna tylko w przypadku prostych i małych zbiorów danych. Z kolei ART-1, pomimo dużej pojemności, nie jest w stanie efektywnie generalizować danych, co skutkuje niską dokładnością.

W porównaniu do bardziej zaawansowanych architektur, takich jak CNN i MLP, obie omawiane sieci są mniej wydajne. CNN oferują najlepsze wyniki w przypadku analizy obrazów, podczas gdy MLP zapewniają dobrą równowagę pomiędzy prostotą a efektywnością w zadaniach klasyfikacyjnych.



## Zakończenie

Niniejsza praca skupiła się na rozwoju i ocenie różnych podejść do rozpoznawania wzorców z wykorzystaniem sztucznych sieci neuronowych. Głównym celem było porównanie efektywności dwóch klasycznych modeli: sieci Hopfielda i sieci adaptacyjnej teorii rezonansu (ART-1), które zostały zaimplementowane i przetestowane w kontekście rozpoznawania wzorców graficznych. Wyniki badań dostarczyły cennych informacji na temat możliwości i ograniczeń tych metod, wskazując na konieczność świadomego doboru architektury sieci w zależności od specyficznych wymagań zadania oraz charakterystyki danych.

Sieć Hopfielda okazała się efektywna w przechowywaniu i przywracaniu prostych wzorców, jednak jej ograniczenia w kontekście bardziej złożonych, szumowych danych sugerują, że nie jest odpowiednia do wszystkich zastosowań. Z kolei sieć ART-1, mimo większej elastyczności i zdolności do dynamicznego uczenia się nowych wzorców, również wymaga optymalizacji i precyzyjnego dostrojenia parametrów, takich jak wartość czujności, aby osiągać satysfakcjonujące wyniki.

Ostateczne wnioski z badań wskazują, że stabilność, zdolność do adaptacji oraz precyzyjne rozpoznawanie nowych wzorców to kluczowe czynniki dla skuteczności systemów rozpoznawania wzorców. Jednocześnie, analiza wyników podkreśla znaczenie dalszej optymalizacji, szczególnie w zakresie poprawy dokładności klasyfikacji i możliwości radzenia sobie z bardziej złożonymi danymi.

## Przyszłe plany

W oparciu o uzyskane wyniki, dalszy rozwój systemu będzie obejmował kilka kluczowych obszarów:

1. **Automatyczna optymalizacja parametrów sieci ART-1:** Jednym z głównych wyzwań w pracy z siecią ART-1 jest odpowiednie dostrojenie parametrów, takich jak wartość czujności, które wpływają na zdolność sieci do skutecznego rozpoznawania nowych wzorców i tworzenia kategorii. Planuje się wdrożenie algorytmów optymalizacji hiperparametrów, takich jak wyszukiwanie losowe (random search) lub optymalizacja bayesowska, aby zautomatyzować ten proces. Dzięki temu, sieć ART-1 będzie mogła dynamicznie dostosowywać się do różnych zestawów danych, poprawiając swoją dokładność i wydajność.
2. **Integracja z głębokimi sieciami neuronowymi:** Sieci Hopfielda i ART-1 działają dobrze w określonych warunkach, jednak ich skuteczność w rozpoznawaniu bardziej złożonych wzorców może być ograniczona. W przyszłych pracach planuje się integrację tych klasycznych architektur z nowoczesnymi głębokimi sieciami neuronowymi (DNN), takimi jak sieci splotowe (CNN) czy sieci rekurencyjne (RNN). Tego rodzaju połączenie mogłoby połączyć zalety klasycznych podejść z nowoczesnymi technologiami, co może znacząco poprawić zdolności systemu do rozpoznawania złożonych wzorców w dużych i zróżnicowanych zbiorach danych.
3. **Zastosowanie na większych i bardziej zróżnicowanych zbiorach danych:** Dotychczasowe testy przeprowadzono na relatywnie prostych zestawach danych graficznych. W przyszłości system zostanie przetestowany na bardziej złożonych i zróżnicowanych zbiorach danych, obejmujących zarówno dane graficzne, jak i inne typy informacji, takie jak tekst czy dane czasowe. Zastosowanie bardziej złożonych danych pozwoli na lepsze zrozumienie ograniczeń i możliwości systemu, a także na wprowadzenie koniecznych usprawnień.
4. **Hybrydowe podejścia do rozpoznawania wzorców:** Planowane jest opracowanie hybrydowych modeli, które połączą różne podejścia do rozpoznawania wzorców. Możliwe jest połączenie sieci Hopfielda lub ART-1 z technikami głębokiego uczenia, co pozwoliłoby na skuteczniejsze przechowywanie wzorców i dynamiczne dostosowywanie się do nowych danych. Wprowadzenie hybrydowych systemów

mogłoby poprawić zdolność do rozpoznawania wzorców w kontekście bardziej złożonych i nieliniowych problemów.

5. **Eksploracja nowych architektur sieci neuronowych:** W związku z dynamicznym rozwojem sztucznej inteligencji, warto rozważyć eksplorację nowych architektur, takich jak sieci transformatorowe (Transformers), które zdobywają popularność w zadaniach przetwarzania języka naturalnego i rozpoznawania obrazów. Badanie możliwości tych architektur w kontekście rozpoznawania wzorców mogłoby otworzyć nowe ścieżki rozwoju systemów.

Podsumowując, praca ta stanowi solidną podstawę do dalszych badań i rozwoju w dziedzinie rozpoznawania wzorców z wykorzystaniem sztucznych sieci neuronowych. Wprowadzenie planowanych usprawnień i rozszerzeń umożliwi stworzenie bardziej wszechstronnych i skutecznych systemów, zdolnych do przetwarzania i interpretacji złożonych danych w dynamicznie zmieniającym się środowisku technologicznym.

## Bibliografia

- [1] Grossberg, S. (2013). Adaptive resonance theory. *Scholarpedia*, 8(5).
- [2] Dmitrienko, V. D., Zakovorotnyi, A. Y., Leonov, S. Y., & Khavina, I. P. (2014). Neural networks ART: Solving problems with multiple solutions and new teaching algorithm. *Open Neurology Journal*, 8.
- [3] Silva, L. E. B. da, Elnabarawy, I., & Wunsch, D. C. (2019). A survey of adaptive resonance theory neural network models for engineering applications. *Neural Networks*, 120.
- [4] Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7(2).
- [5] Duda, R. O., Hart, P. E., & Stork, D. G. (2000). *Pattern Classification* (2nd ed.). John Wiley & Sons.
- [6] Vapnik, V. N. (1995). *The Nature of Statistical Learning Theory*. Springer.
- [7] Conte, D., Foggia, P., Sansone, C., & Vento, M. (2004). Thirty years of graph matching in pattern recognition. *International Journal of Pattern Recognition and Artificial Intelligence*, 18(3).
- [8] Fu, K. S. (1982). *Syntactic Pattern Recognition and Applications*. Prentice-Hall.
- [9] Pavlidis, T. (1977). *Structural Pattern Recognition*. Springer.
- [10] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553).
- [11] Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323 (6088).
- [12] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. *Advances in neural information processing systems*.
- [13] Grossberg, S. (1980). How does a brain build a cognitive code? *Psychological Review*, 87(1).
- [14] Carpenter, G. A., & Grossberg, S. (1987). ART 2: Self-organization of stable category recognition codes for analog input patterns. *Applied Optics*, 26(23).
- [15] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015). Going deeper with convolutions. *Proceedings of the IEEE conference on computer vision and pattern recognition*.
- [16] Zhou, Z., Siddiquee, M. M. R., Tajbakhsh, N., & Liang, J. (2019). UNet++: A nested U-Net architecture for medical image segmentation. *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*. Springer.
- [17] McCulloch, W. S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5(4).
- [18] Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6).

- [19] Minsky, M., & Papert, S. (1969). *Perceptrons: An Introduction to Computational Geometry*. MIT Press.
- [20] Crevier, D. (1993). *AI: The Tumultuous Search for Artificial Intelligence*. Basic Books.
- [21] Werbos, P. J. (1974). *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences*. PhD Thesis, Harvard University.
- [23] LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1(4).
- [24] Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., & Riedmiller, M. (2015). Human-level control through deep reinforcement learning. *Nature*, 518(7540).
- [25] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*.
- [26] Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., van den Driessche, G., ... & Hassabis, D. (2016). Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587).
- [27] Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
- [28] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11).
- [29] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE conference on computer vision and pattern recognition*.
- [30] Carpenter, G. A., Grossberg, S., & Reynolds, J. H. (1991). ARTMAP: Supervised real-time learning and classification of nonstationary data by a self-organizing neural network. *Neural Networks*, 4(5).
- [31] Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M., & Monfardini, G. (2009). The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1).
- [32] Zhang, Z., Cui, P., & Zhu, W. (2018). Deep learning on graphs: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 30(1).
- [33] Hochreiter, S. (1991). Untersuchungen zu dynamischen neuronalen Netzen. *Diploma, Technische Universität München*.
- [34] Bengio, Y., Simard, P., & Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2).
- [35] Glorot, X., & Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*.
- [36] Dietterich, T. G. (1995). Overfitting and undercomputing in machine learning. *ACM Computing Surveys (CSUR)*, 27(3).
- [37] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1).

- [38] Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- [39] Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *International Conference on Machine Learning*.
- [40] Halevy, A., Norvig, P., & Pereira, F. (2009). The unreasonable effectiveness of data. *IEEE Intelligent Systems*, 24(2).
- [41] Zoph, B., Ghiasi, G., Lin, T. Y., Cui, Y., Liu, H., Cubuk, E. D., & Le, Q. V. (2020). Rethinking pre-training and self-training.
- [42] Glorot, X., Bordes, A., & Bengio, Y. (2011). Deep sparse rectifier neural networks. *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*.
- [43] Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79(8).
- [44] Amit, D. J. (1989). *Modeling Brain Function: The World of Attractor Neural Networks*. Cambridge University Press.
- [45] Grossberg, S. (1987). Competitive learning: From interactive activation to adaptive resonance. *Cognitive Science*.
- [46] Hertz, J., Krogh, A., & Palmer, R. G. (1991). *Introduction to the Theory of Neural Computation*. Addison-Wesley.
- [47] Haykin, S. (1998). *Neural Networks: A Comprehensive Foundation*. Prentice Hall.