Teoria Kategorii

Patryk Gronkiewicz

KN Machine Learning

2022-11-15

Ale co to w ogóle jest?

~~Straszna~~ matematyka, blisko spokrewniona z topologią algebraiczną. Szczęśliwie, topologii tutaj dotykać nie będziemy. Na początek musimy zdefiniować sobie kilka podstawowych pojęć:

- Kategoria
- Morfizm

No dobra, kategoria kategorią, ale co to jest?!

Na kategorię składa się kilka bytów matematycznych (które nazywamy klasami):

- $Ob(C)$ — obiekty
- $hom(C)$ — morfizmy/strzałki/mapy
- $|\circ|$ operator złożenia funkcji ($f \circ g = f(g)$)

O każdym z nich porozmawiamy za chwilę.

Obiekty

Może to być... cokolwiek — liczby, kształty, zwierzątka. Do tego przejdziemy później. Ważne, żeby dało się zmienić jeden obiekt w drugi z małą pomocą.

Złożenia

Na początku dobrze jest pokazać na czym polega operator złożenia funkcji.

Przykład

$$f(x) = 2x,\ g(x) = x^2$$

$$f \circ g = f(g(x)) = 2\left(x^2\right) \neq g \circ f = g(f(x)) = (2x)^2 = 4x^2$$

Morfizmy

Morfizmy to przekształcenia między konkretnymi obiektami

$$f : a \mapsto b$$

Taki obiekt nazywamy morfizmem f mapującym obiekt a w obiekt b. Jednym z podstawowych jest morfizm identycznościowy $id_a : a \mapsto a$

Typy morfizmów

Jest ich trochę, więc każdy dostanie swój slajd

Typy morfizmów — monomorfizm

ang. *monomorphism/monic*
Jeśli $f \circ g_1 = f \circ g_2$, to $g_1 = g_2$ dla wszystkich morfizmów $g_1, g_2 : x \mapsto a$

$$f \circ g_1 = f \circ g_2 \implies \forall (g_1, g_2 : x \mapsto a) : g_1 = g_2$$

Typy morfizmów — epimorfizm

ang. *epimorphism/epic*
Jeśli $g_1 \circ f = g_2 \circ f$, to $g_1 = g_2$ dla wszystkich morfizmów $g_1, g_2 : x \mapsto a$

$$g_1 \circ f = g_2 \circ f \implies \forall(g_1, g_2 : x \mapsto a) : g_1 = g_2$$

Typy morfizmów — bimorfizm

ang. *bimorphism*
jest zarówno monomorfizmem i epimorfizmem

Typy morfizmów — izomorfizm

ang. *isomorphism*
Jeśli istnieje morfizm $g : b \mapsto a$, taki, że $f \circ g = id_b$ i $g \circ f = id_a$

Uwaga

Nie każdy bimorfizm jest izomorfizmem! Kategoria złożona z A, B, morfizmów identycznościowych (id_a, id_b) i pojedynczego morfizmu $f : a \mapsto b$ nie ma izomorfizmów, natomiast f jest bimorfizmem.

Typy morfizmów — endomorfizm

ang. *endomorphism*
Każdy morfizm taki, że $f : a \mapsto a$. Oznaczamy je $End(a)$

ang. *automorphism*
Morfizm, który jest zarówno izomorfizmem i endomorfizmem.
Oznaczamy je przez $Aut(a)$.

Przykład

Najprostszym przykładem jest morfizm identycznościowy. Zachowuje się on tak samo, niezależnie od strony z której zostanie zaaplikowany

$$id_a \circ a = a \circ id_a = a$$

Typy morfizmów — retrakcja

ang. *retraction*
Jeśli istnieje prawa odwrotność, np. $g : b \to a$ dla $f \circ g = id_b$

Typy morfizmów — sekcja

ang. *section*
Jeśli istnieje lewa odwrotność, np. $g : b \to a$ dla $g \circ f = id_b$

Kto powiedział, że nie można zmieniać całych kategorii w inne?
Funktory to morfizmy między kategoriami.
Funktor każdemu obiektowi z kategorii C przypisuje obiekt z kategorii D i tak samo dla morfizmów — każdy ma swój odpowiednik. Istnieją dwa typy funktorów — kowariantne (niezmieniające zwrotu mapowania) i kontrawariantne (zmieniające zwrot mapowania)

Dodatkowe materiały

- `https://github.com/BartoszMilewski/Publications/blob/master/TheDaoOfFP/DaoFP.pdf` (ciągle aktualizowane)
- `https://www.youtube.com/user/DrBartosz` (Wykłady po angielsku nt. teorii kategorii, programowania funkcyjnego itp.)
- `https://blog.ploeh.dk/2017/10/04/from-design-patterns-to-category-theory/` (na 15 listopada 2022 jeszcze niedokończone, ale bardzo obszerne)
- `https://www.cs.princeton.edu/~dpw/courses/cos326-12/notes/basics.php` (kurs COS326 prowadzony przez Princeton University)