

Rekomendacje oparte na grafach wiedzy

Systemy Rekomendacyjne 2024/2025

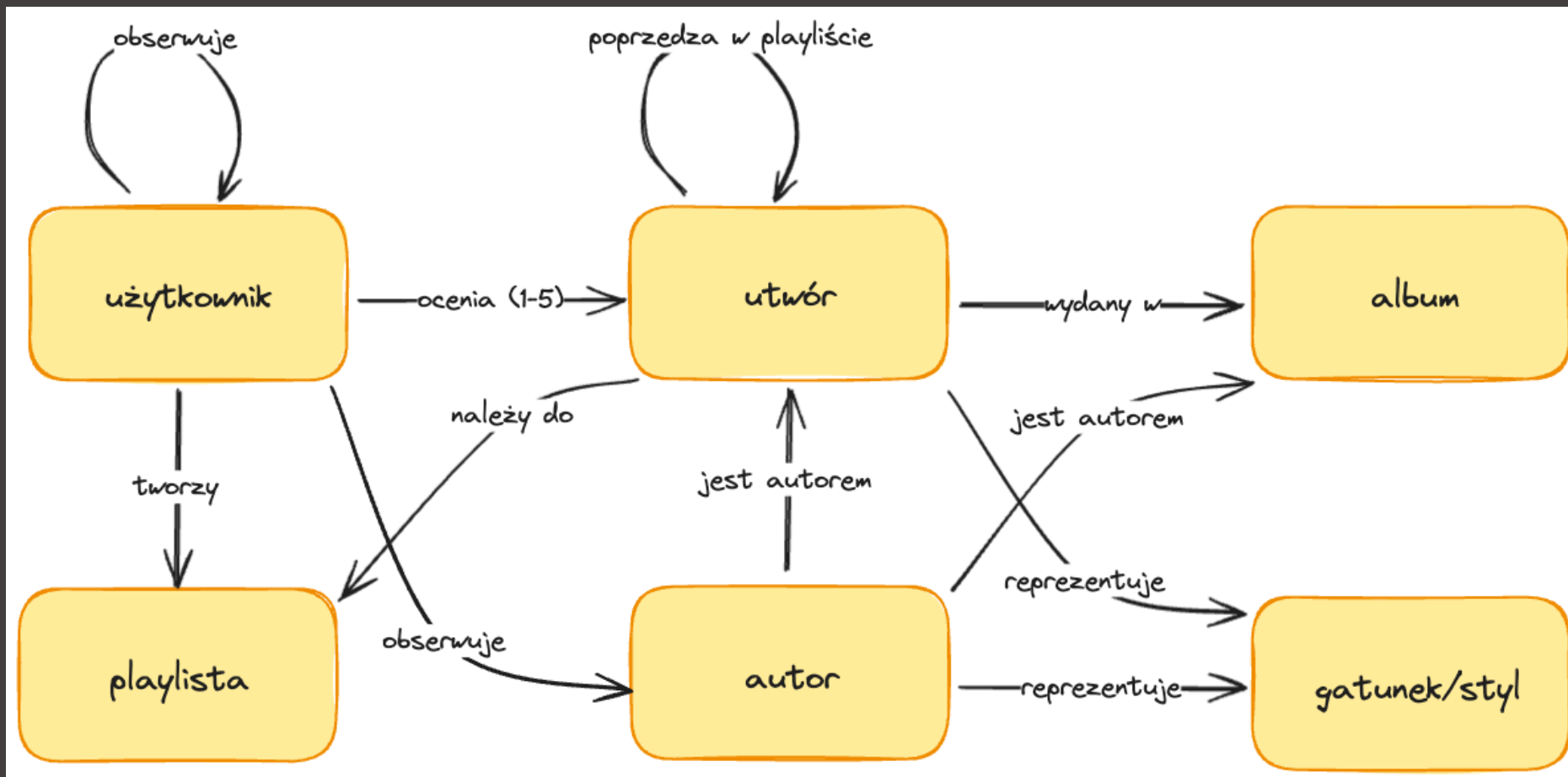
Reprezentacja danych jako graf

- Wierzchołkami grafu mogą być np.
 - użytkownicy
 - treści (np. filmy, utwory)
 - gatunki, albumy, autorzy
 - playlisty utworzone przez użytkowników
- Krawędziami są (niemal) dowolne relacje między wierzchołkami

Reprezentacja danych jako graf

- Graf zazwyczaj jest skierowany (np. użytkownik polubił film, utwór reprezentuje gatunek)
- Graf może być ważony (np. gdy użytkownik ocenia film w skali 1-5)
- Jeśli nie mamy relacji między tymi samymi klasami wierzchołków (np. między użytkownikami), to graf jest N-dzielny
- Do reprezentacji w grafie nie nadają się skalary (np. cechy wyrażone jako liczby z przedziału $[0, 1]$)

Reprezentacja danych jako graf



Vertex embeddings

Deep Walk

- <https://dl.acm.org/doi/abs/10.1145/2623330.2623732>
- Modyfikacja Word2Vec
- Generujemy (bardzo, bardzo dużo) losowych ścieżek w grafie
- Ścieżki traktujemy jak zdania w korpusie
- Na wygenerowanych ścieżkach uruchamiamy Word2Vec
- Autorzy sugerują 30-60 ścieżek wygenerowanych z każdego z wierzchołków oraz okienko na 10+10 sąsiednich wierzchołków

Deep Walk - wady

- Brak wag w podstawowej implementacji - wierzchołki o wysokim stopniu będą częściej odwiedzane i przez to nadreprezentowane w zbiorze treningowym
- Brak embeddingów dla krawędzi

Node2Vec

- <https://dl.acm.org/doi/abs/10.1145/2939672.2939754>
- Modyfikacja Deep Walk rozwiązująca problem braku lokalności
- Dwa parametry:
 - P – steruje prawdopodobieństwem odwiedzenia już wcześniej odwiedzonych wierzchołków
 - Odpowiada za odzwierciedlenie lokalnych zależności
 - Q – steruje prawdopodobieństwem odwiedzenia jeszcze nieodwiedzonych wierzchołków
 - Odpowiada za uwzględnienie kształtu całego grafu

Ograniczenia metody

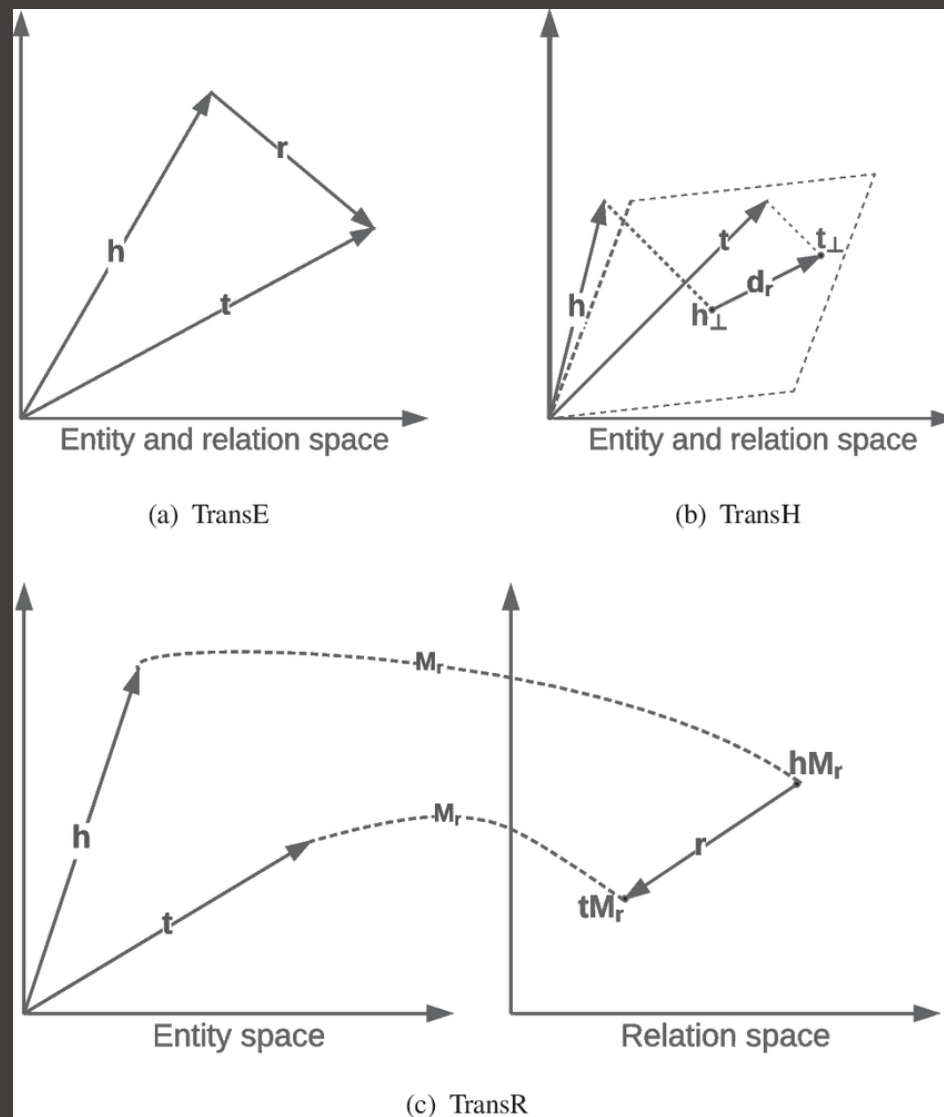
- Brak reprezentacji krawędzi
- Metoda dostosowana do grafów o dowolnej strukturze wierzchołków i krawędzi
 - A grafy wiedzy mają zamknięty katalog "typów" krawędzi
- Trening modelu oparty o losowe spacery po grafie
 - Może słabo działać w grafach rzadkich, niespójnych itp.

Knowledge graph embeddings

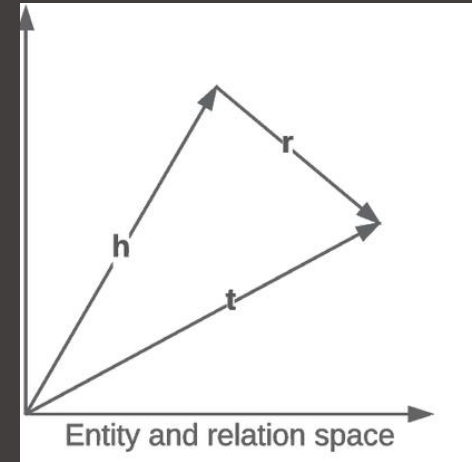
Grafy wiedzy - reprezentacja

- Metody osadzeń grafów wiedzy nie operują na poziomie ścieżek w grafie, a na poziomie **relacji między elementami**
- Pojedyncza relacja to trójka (*head, relation, tail*)
 - w skrócie: (*h, r, t*)
- Graf reprezentowany jest jako lista takich trójek

Metody translacyjne



TransE



- https://proceedings.neurips.cc/paper_files/paper/2013/file/1ccc7a77928ca8133fa24680a88d2f9-Paper.pdf
- Wierzchołki i relacje osadzone są w tej samej przestrzeni
- $emb(h) + emb(r) = emb(t)$
- *Corrupted triplets* – sztucznie wygenerowane negatywne (niestniejące) relacje

TransE

Algorithm 1 Learning TransE

input Training set $S = \{(h, \ell, t)\}$, entities and rel. sets E and L , margin γ , embeddings dim. k .

1: **initialize** $\ell \leftarrow \text{uniform}(-\frac{6}{\sqrt{k}}, \frac{6}{\sqrt{k}})$ for each $\ell \in L$

2: $\ell \leftarrow \ell / \|\ell\|$ for each $\ell \in L$

3: $\mathbf{e} \leftarrow \text{uniform}(-\frac{6}{\sqrt{k}}, \frac{6}{\sqrt{k}})$ for each entity $e \in E$

4: **loop**

5: $\mathbf{e} \leftarrow \mathbf{e} / \|\mathbf{e}\|$ for each entity $e \in E$

6: $S_{batch} \leftarrow \text{sample}(S, b)$ // sample a minibatch of size b

7: $T_{batch} \leftarrow \emptyset$ // initialize the set of pairs of triplets

8: **for** $(h, \ell, t) \in S_{batch}$ **do**

9: $(h', \ell, t') \leftarrow \text{sample}(S'_{(h, \ell, t)})$ // sample a corrupted triplet

10: $T_{batch} \leftarrow T_{batch} \cup \{((h, \ell, t), (h', \ell, t'))\}$

11: **end for**

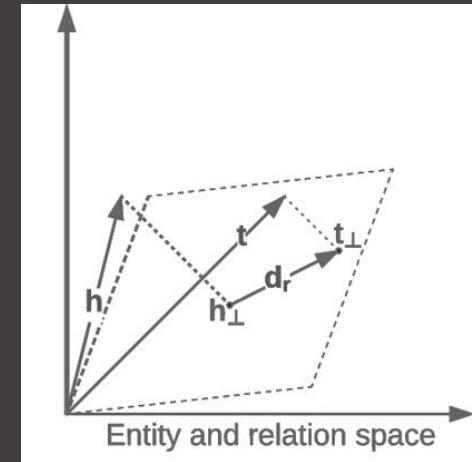
12: Update embeddings w.r.t.
$$\sum_{((h, \ell, t), (h', \ell, t')) \in T_{batch}} \nabla [\gamma + d(\mathbf{h} + \ell, \mathbf{t}) - d(\mathbf{h}' + \ell, \mathbf{t}')]_+$$

13: **end loop**

TransE

- Zalety:
 - prosta abstrakcja
 - stosunkowo szybki trening
 - Reprezentacja bytów w jednej przestrzeni - łatwa predykcja
- Wady:
 - źle odzwierciedla relacje symetryczne i $1:n$

TransH



- <https://ojs.aaai.org/index.php/AAAI/article/view/8870>
- Rozwinięcie TransE
- Wierzchołki i relacje osadzone są w tej samej przestrzeni
- Relacje reprezentowane jako para wektorów - wektor normalny płaszczyzny i wektor przesunięcia w tej płaszczyźnie

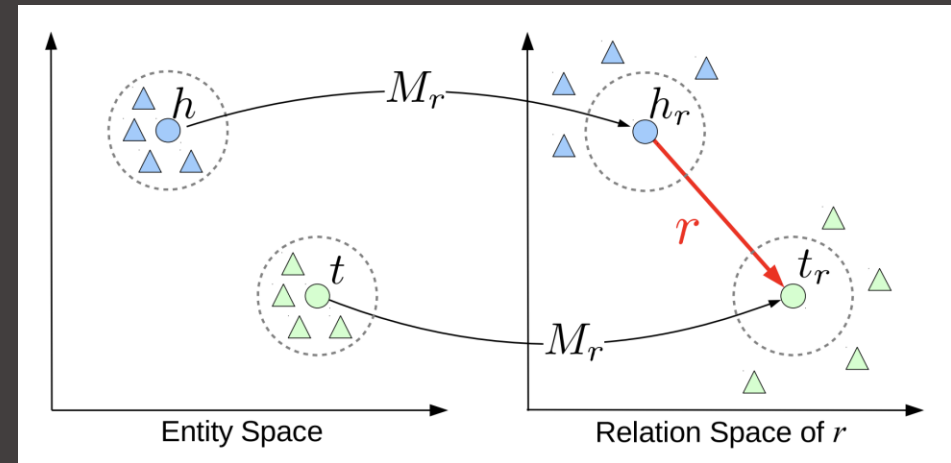
TransH

- w_r – wektor normalny płaszczyzny relacji r
- d_r – wektor przesunięcia wzdłuż płaszczyzny relacji r
- rzutowanie wierzchołków na płaszczyznę relacji:
$$emb'(h) = emb(h) - w_r^t * emb(h) * w_r$$
- dla relacji (h, r, t) zachodzi:
$$emb'(h) + d_r = emb'(t)$$

TransH

- Zalety:
 - lepsza reprezentacja relacji symetrycznych i $1:n$
 - zazwyczaj wyższa jakość osadzeń i predykcji
- Wady:
 - dłuższy trening modelu
 - trudniejsza predykcja – przez konieczność rzutowania na płaszczyznę

TransR



- https://linyankai.github.io/publications/aaai2015_transr.pdf
- Rozwinięcie TransE i TransH
- Wierzchołki i relacje osadzone są w **różnych** przestrzeniach
- Relacje reprezentowane jako para – wektor przesunięcia w przestrzeni tej konkretnej relacji oraz macierz przekształcenia dla wierzchołków

TransR

- M_r – macierz projekcji z przestrzeni wierzchołków do przestrzeni relacji r
- d_r – wektor przesunięcia w przestrzeni relacji r
- projektowanie wierzchołków do przestrzeni relacji:
 $emb'(h) = h * M_r$
- dla relacji (h, r, t) zachodzi:
 $emb'(h) + d_r = emb'(t)$

TransR - trening

- funkcja celu dla pojedynczej trójki:

$$f_r(h, t) = \|\mathbf{h}_r + \mathbf{r} - \mathbf{t}_r\|_2^2.$$

- funkcja celu dla całego modelu:

$$L = \sum_{(h,r,t) \in S} \sum_{(h',r,t') \in S'} \max(0, f_r(h, t) + \gamma - f_r(h', t')),$$

- trening metodą gradient descent

TransR

- Zalety:
 - lepsza reprezentacja relacji symetrycznych i $1:n$
 - zazwyczaj wyższa jakość osadzeń i predykcji
 - niezależny rozmiar osadzeń dla wierzchołków i relacji
- Wady:
 - znacznie dłuższy trening modelu
 - trudniejsza predykcja – przez konieczność projekcji do innej przestrzeni (nie ma gwarancji, że istnieje macierz odwrotna)

Metody oparte o faktoryzację macierzy

- Metody translacyjne, wbrew pozorom, nie dają najlepszych rezultatów, w dodatku wraz ze wzrostem jakości modelu, rośnie jego obliczeniowe i koncepcyjne skomplikowanie
- Faktoryzacja macierzy to problem dobrze znany i popularny

RESCAL

- <https://www.cip.ifi.lmu.de/~nickel/data/paper-icml2011.pdf>
- Graf reprezentowany jest jako 3-wymiarowy tensor X
 - $X_{ijk} = 1$ oznacza, że w grafie istnieje trójka (e_i, r_k, e_j)
- Chcemy dokonać dekompozycji:
 $X_k = A * R_k * A^T$, gdzie:
 - A - macierz o wymiarach n na r , i -ty wiersz reprezentuje osadzenie i -tego wierzchołka w grafie
 - R_k – macierz o wymiarach r na r reprezentująca k -tą relację
 - dla trójki (h, k, t) zachodzi: $A_h * R_k = A_t$

Od embeddingu do rekomendacji

Metody wierzchołkowe

- Wszystkie byty osadzone są w tej samej przestrzeni
- Rekomendacje oparte są o odległości pomiędzy osadzeniami reprezentującymi poszczególne byty
 - Zazwyczaj – im bliżej, tym lepiej

Metody wierzchołkowe - przykłady

- Propozycje nowych znajomych w sieciach społecznościowych
 - inni użytkownicy najbliżsi danemu użytkownikowi
- Rekomendacje treści
 - treści (utwory, filmy, ...) najbliższe danemu użytkownikowi
- Rekomendacje kontekstowe
 - treści najbliższe kilku ostatnim treściom dobrze przyjętym przez danego użytkownika
- Metody łączone
 - treści bliskie danemu użytkownikowi, reprezentujące gatunki najbliższe danemu użytkownikowi

Metody krawędziowe

- Osadzenia grafów wiedzy oferują kompatybilne ze sobą osadzenia wierzchołków oraz relacji (krawędzi) między nimi
- Krawędzie reprezentują istotne semantycznie relacje pomiędzy bytami
- Możemy oprzeć proces rekomendacji o przewidywanie brakujących połączeń w grafie
- Możemy rekomendować tylko zgodnie z relacjami istniejącymi w grafie

Metody krawędziowe - przykłady

- Propozycje nowych znajomych w sieciach społecznościowych
 - szukamy użytkowników x pasujących do trójki $(użytkownik, znajomy, x)$
- Propozycje artystów, których warto obserwować
 - szukamy artystów y pasujących do trójki $(użytkownik, obserwuje, y)$
- Propozycje kolejnych utworów w playliście
 - szukamy utworów z pasujących do trójki $(utwór, kolejny_w_playliście, z)$

Ograniczenia

Jak nie uwięzić użytkownika w złotej klatce?

- *Filter bubble* – pozytywne sygnały zwrotne dotyczące poprzednich rekomendacji ograniczają różnorodność przyszłych rekomendacji
- *Echo chamber* – powtarzające się, wybiórcze rekomendacje utwierdzają użytkownika w przekonaniu, że system rekomendacyjny odzwierciedla jego gust, a ograniczony zbiór informacji - rzeczywistość
- Oba mechanizmy, poza aspektami etycznymi, powodują także znudzenie użytkowników

Podsumowanie

- Reprezentacja wiedzy w formie grafu
- Osadzanie wierzchołków dowolnego grafu (DeepWalk, Node2Vec)
- Osadzanie wierzchołków i relacji grafu wiedzy:
 - Metody translacyjne (TransE, TransH, TransR)
 - Metody faktoryzacyjne (RESCAL)
- Wykorzystanie osadzeń do rekomendacji
- Ograniczenia i zagrożenia