

**КОМПАРАТИВНА АНАЛИЗА ПРИМЕНЕ ТРАДИЦИОНАЛНИХ МЕТОДА И ГРАФ НЕУРОНСКИХ МРЕЖА У ОБЛАСТИ СИСТЕМА ЗА ПРЕПОРУКУ****COMPARATIVE ANALYSIS OF THE APPLICATION OF TRADITIONAL METHODS AND GRAPH NEURAL NETWORKS IN THE AREA OF RECOMMENDER SYSTEMS**

Јелена Достих, Факултет техничких наука, Нови Сад

**Област – ЕЛЕКТРОТЕХНИКА И РАЧУНАРСТВО**

**Кратак садржај** – Системи за препоруку релевантних објеката све чешће постају део наше свакодневице и утичу на наше одабире и активности. Подаци коришћени у системима за препоруку природно могу бити представљени у виду графа у којем кориснике и објекте представљамо као чворове тог графа, а њихове интеракције као везе између њих. Овај рад пореди ефикасност традиционалних система за препоруку и система за препоруку базираних на граф неуронским мрежама. У циљу поређења, имплементирана су два система - хибридни традиционални систем и систем базиран на граф неуронским мрежама. Перформансе, односно резултати експеримената извршених над оба система евалуирани су и упоређени на основу прецизности, одзива и  $F1$  мере, односно средњеквадратне грешке у случају континуалних вредности.

**Кључне речи:** системи за препоруку, граф неуронске мреже, матрица сличности, граф конволуција

**Abstract** – Recommendation systems have increasingly become part of our everyday life and influence our choices and activities. Data used in recommendation systems can naturally be represented as a graph in which we present users and objects as nodes of that graph and their interactions as links between them. This paper compares the effectiveness of traditional recommendation systems and recommendation systems based on graph neural networks. For comparison, two systems were implemented - a traditional hybrid system and a system based on graph neural networks. Performance and results of experiments performed on both systems were evaluated and compared based on precision, recall, and  $F1$  measure, but also mean square error in the case of continuous values.

**Keywords** recommendation systems, graph neural networks, similarity matrix, graph convolution

**1. УВОД**

Порастом количине информација које нас окружују, појављује се потреба за издвајањем, филтрирањем и рангирањем корисних информација.

**НАПОМЕНА:**

Овај рад проистекао је из мастер рада чији ментор је била др Јелена Сливка, ванр. проф.

Системи за препоруку релевантних објеката све чешће постају део наше свакодневице и утичу на наше одабире и активности. Објекат препоруке се разликује у зависности од самог домена у којем се систем користи.

Подаци који се користе за креирање система за препоруку су подаци о корисницима, подаци о објектима који се препоручују и прикупљене оцене које су корисници доделили објектима. Овакви подаци природно могу бити представљени у виду графа у којем кориснике и објекте представљамо као чворове тог графа, а њихове интеракције као везе између њих. Граф неуронске мреже представљају један од облика дубоког машинског учења. За улазне податке користе податке структуриране у виду графа, те их можемо искористити приликом креирања система за препоруку.

Овај рад пореди ефикасност традиционалних система за препоруку и система за препоруку базираних на граф неуронским мрежама.

У циљу поређења, имплементирана су два система - традиционални и систем базиран на граф неуронским мрежама. Перформансе, односно резултати експеримената извршених над оба система приказани су у поглављу 4, где су се као основне метрике користиле прецизност, одзив,  $F1$  мера и средњеквадратна грешка у случају континуалних вредности.

Поглавље 2 описује неке од досадашњих приступа који су увели појам граф неуронских мрежа у област система за препоруку. Поглавље 3 се фокусира на методологију и детаљније описује скуп података, а затим и начин на који су добијени и евалуирани резултати креираних система који ће затим бити продискутовани у поглављу 4. На самом крају, у поглављу 5 описани су главни закључци, као и предности и мане које се могу запазити у оба приступа.

**2. ПРЕТХОДНА РЕШЕЊА**

У овом поглављу ће бити представљено неколико значајних радова који су се бавили архитектуром граф неуронских мрежа, као и њиховом применом у области система за препоруку. Ова област нуди неколико праваца и другачијих начина употребе графова у овој улози.

Хронолошки гледано, међу првима се истакао рад који говори о структури и начину функционисања граф конволутивне мреже (енг. *Graph Convolutional Network, GCN*) [1]. Једно од проширења овог рада је и опис колаборативног филтрирања заснованог на граф архитектури мреже (енг. *Neural Graph Collaborative Filtering, NGCF*) [2]. У њему је објашњено како граф неуронске мреже доприносе побољшању крајњих препорука, увођењем информација о интеракцијама између корисника и објеката.

Мало другачије приступе представљају *GraphSage* [3] и његов наследник *PinSage* [4] алгоритми, који уводе унапређење тиме што у процесу проналажења препорука узимају у обзир само фиксни, мањи подскуп најближих комшија, за разлику од њихових претходника који користе целе, врло често веома велике скупове података.

Унапређење претходних приступа је алгоритам представљен у раду “*Multi-Component Graph Convolutional Collaborative Filtering*” (*MCCF*) [5] који додатно уводи и механизам пажње у процес формирања препорука циљаног чвора. Механизам пажње омогућава разликовање тежине и нивоа утицаја комшијских чворова на циљани чвор.

Поред поменутих радова, један другачији правац креирања препорука представљају системи који приликом креирања препорука уводе и информације о социјалним интеракцијама међу корисницима. У овој категорији најбољи представници су *DiffNet++* [6] и *GraphRec* [7] алгоритми. Сличност коју деле је увођење информација о међусобним интеракцијама корисника, међутим *DiffNet++* користи два одвојена бипартитивна графа за њихову репрезентацију, док *GraphRec* користи један хетерогени граф који другачијим врстама веза прави разлике у горе поменутих интеракцијама.

Још један интересантан приступ који је настао као комбинација две сложеније структуре је *Knowledge Graph Attention Network (KGAT)* [8]. Његове карактеристике су то да користи бенефите механизма пажње, али и повећава комплексност тиме што у репрезентацији графа садржи чворове који не представљају само кориснике и објекте, већ додатно и њихове атрибуте.

### 3. МЕТОД

Методологија, односно поступак креирања два система за препоруку приказан је у овом поглављу. Један од система заснован је на традиционалним методама, док је други заснован на граф неуронским мрежама. У поглављима 3.2 и 3.3 детаљније је описан сваки од њих. Оба система су развијена у програмском језику *Python*.

#### 3.1. Скуп података

Ради већег фокуса на имплементацији различитих архитектура модела система за препоруку, у експерименталном кораку коришћен је једноставан и често употребљаван *MovieLens* скуп података.

*MovieLens* представља један од најчешће коришћених скупова података у домену система за препоруку,

углавном примењен у едукативне сврхе као и у истраживачким радовима у циљу поређења перформанси различитих система. Садржи информације о филмовима и корисницима, као и оценама, односно рејтинзима који су додељени филмовима од стране корисника.

Постоји неколико верзија и неколико различитих величина овог скупа података. У овом раду коришћен је скуп који садржи 100,000 забележених рејтинга додељених 1700 различитих филмова од стране 1000 различитих корисника. Тренинг и тест подела података је одрађена у односу 90:10, односно, 90,570 рејтинга је употребљено за обучавање модела, док је преосталих 9,430 употребљено за евалуацију.

Сваки рејтинг описан је идентификатором корисника, идентификатором филма који је оценио и оценом која може имати вредност између 1 и 5.

Корисници су поред идентификатора описани још и бројем година, полом и занимањем, док су филмови описани називом и годином објављивања. Поред тога, за сваки филм постојала је информација којим жанровима припада. У случају традиционалног система базираног на садржају, ради квалитетнијег формирања вектора филмова, додатно су коришћене и кључне речи, односно тагови преузети из новије верзије *MovieLens* скупа података.

#### 3.2. Традиционалне методе

Ово поглавље описује кораке припреме података, креирања два подсистема која ће формирати крајњи модел као и начин на који су они агрегирани формирајући финални хибридни систем за препоруку.

##### 3.2.1 Систем заснован на садржају

Код система заснованог на садржају потребно је користити особине објеката, односно филмова. Како почетни скуп садржи релативно мали број колона, односно особина, било је неопходно одрадити неколико трансформација над њим. У оригиналном формату података, филмови су описани кроз једну особину која је сачињена од назива и године објављивања филма и, поред тога, кроз листу жанрова којима филм припада. Оно што је додатно коришћено у приступу приказаном у овом раду јесте информација о таговима, односно кључним речима, које су поједини корисници додељивали појединим филмовима.

Измене оригиналних података резултовале су новим скупом особина, где је сваки филм садржао назив, годину објављивања, деценију објављивања и као и листу жанрова и листу агрегираних кључних речи у “*bag of words*” формату. Приликом креирања латентних вектора креиране су различите комбинације ових особина. Осим различитих особина које улазе у процес креирања вектора, оно што се разликује јесте и метода којом се они креирају. С обзиром на то да су коришћене особине у текстуалном формату, било је неопходно применити одређену трансформацију текста у нумерички формат. За ове потребе могуће је користити различите векторајзере. У овом раду приме-

њена су два једноставна модела - “*Count*” и “*TF-IDF*” векторајзери.

### 3.2.2 Систем заснован на колаборативном филтрирању

Приликом креирања система заснованог на колаборативном филтрирању, кораци припреме података су поједностављени с обзиром да се користе само информације о рејтинзима. Као што је случај код сваког система за препоруку базираног на колаборацији, креирана је матрица односа корисника и филмова. Након тога формирана је матрица сличности филмова коришћењем *Pearson* мере сличности, на основу рејтинга додељених од стране корисника. Креирана матрица је затим употребљена у циљу проналажења препорука за конкретног корисника на основу филмова које је он до тада оценио. Добијени вектори су затим употребљени за креирање матрице сличности, приликом чега је коришћена косинусна мера сличности. Вредности резултујуће матрице представљале су меру сличности сваког филма са преосталим филмовима из коришћеног скупа података.

### 3.2.2 Креирање хибридног система

Системи су агрегирани на једноставан начин у циљу смањења проблема хладног почетка (енг. “*cold start*”), али истовремено и креирања семантички богатијих препорука. Примењено правило агрегације је у основи врло једноставно - након провере броја филмова које је корисник до тада погледао, систем се преусмерава на одговарајући систем.

Уколико би корисник погледао мање од два филма, као резултат издвојени су тренутно најбоље ранжирани филмови. Са друге стране, уколико би корисник погледао више од десет филмова систем је преусмерен на логику колаборативног филтрирања. Коначно, у случајевима где је корисник погледао мањи број филмова, коришћена логика система базираног на садржају.

### 3.3. Систем заснован на граф неуронским мрежама

Други систем креиран је уз примену граф неуронских мрежа. С обзиром на специфичан формат података над којима граф неуронске мреже могу да се обуче, било је неопходно одрадити одговарајућу трансформацију улазних података, што је уједно и први корак овог приступа који је детаљније описан у поглављу 3.3.1 Припремљени подаци у форми графа се у наставку користе за тренирање модела граф неуронске мреже. Овај корак је детаљније описан у поглављу 3.3.2.

#### 3.3.1 Припрема података

Од почетне форме података, који су описани у поглављу 3.1, на почетку је неопходно креирати њихову нумеричку репрезентацију. Подаци се примарно представљају у виду тензора. Посебно су креирани тензори који представљају *кориснике*, *филмове* и *рејтинге*, а посебно они који представљају њихове особине - *године*, *радно место* и *пол* у случају корисника, односно *жанр* у случају филмова.

Тензори, односно вишедимензионе матрице, су даље коришћени за формирање финалног облика улазних

података, односно, **графова**. Овај корак јесте и један од кључних који прави разлику између граф неуронских мрежа од осталих типова. При имплементацији овог система, коришћена је *DGL*, односно *Deep Graph Library* [9] - библиотека за рад са граф неуронским мрежама. Проблем који је решаван уз помоћ ове библиотеке јесте класификација постојећих веза унутар графа, што би доменски представљало предвиђање оцена које ће корисници доделити филмовима и препоручивање оних са највишом оценом. С обзиром на то да је било неопходно имати одвојене типове чворова за кориснике и филмове, подаци су били представљени у форми *хетерогеног графа*.

#### 3.3.2 Архитектура модела

Први корак унутар модела јесте креирање латентних вектора на основу информација преузетих из улазног графа. Формирани су одвојено латентни вектори корисника и филмова, сабирањем вектора њихових особина. У наставку се креиран латентни вектори прослеђују првом скривеном слоју модела.

##### 3.3.2.1 Скривени слој

Унутар сваког скривеног слоја врши се исти поступак ажурирања вредности прослеђених вектора. Вредности латентних вектора претходног слоја се прослеђују одговарајућој граф конволуцији. У овом случају, с обзиром на то да постоје два типа веза, потребно је креирати *HeteroGraphConv* инстанцу унутар *DGL* библиотеке, односно, за сваки тип везе дефинисати посебан тип коволуције. У случају модела описаног у овом раду, за оба типа веза коришћен је посебно креиран тип конволуције - *GCMCConv*. *GCMCConv*, односно “*Graph Convolutional Matrix Completion*” [10] конволуција представља посебан тип конволуције која узима у обзир постојање свих веза унутар графа, с тим да поједине иницијално немају вредност. Циљ граф неуронске мреже јесте да предвиди вредности веза које недостају.

#### 3.3.3 Тренирање модела

Приликом тренирања модела, поред улазних података, који су трансформисани на начин описан у поглављу 3.2.1, прослеђивани су додатни параметри. За потребе креирања латентних вектора било је неопходно проследити димензију вектора у скривеном слоју *hidden\_dim*, али и број корисника *num\_users*, филмова *num\_items* и рејтинга *num\_ratings* коришћених у тренингу. Поред вектора корисника и филмова, креиран су и вектори особина, тако да је било неопходно проследити и број могућих вредности корисникових година *user\_age\_num*, пола *user\_gender\_num* и занимања *user\_job\_num*, као и број жанрова којима филмови могу припадати *movies\_genres\_num*. Поред димензија, прослеђен је и број скривених слојева *num\_layers* који ће формирати финални модел.

Током пропагације у напред, латентни вектори корисника и филмова се сабирају са латентним векторима њихових особина и у таквом формату прослеђују првом слоју модела. Унутар сваког слоја иницијализује се *HeteroGraphConv* објекат, где се дефинише тип коволуције за сваки тип везе. У овом

случају је за оба типа веза искориштена GCMSSonv конволуција. Током пропагације у напред, извршеном граф конволуцијом се добијају предикције које се даље користе за израчунавање грешке модела која се затим користи за ажурирање матрице тежина. У овом кораку коришћена је средњеквадратна грешка, док су оптимизационе функције варирале у зависности од експеримента. Резултати експеримената се могу видети у поглављу 4.

#### 4. РЕЗУЛТАТИ И ДИСКУСИЈА

Ово поглавље описује конкретне перформансе модела описаних у поглављу 3 и тиме заокружује компаративну анализу ових система.

У табели 1 су приказани резултати добијени након извршених експеримената над моделима различитих параметарских конфигурација. За процену система су коришћене “Precision”, “Recall” и “F1” мере, добијене на тест скупу. Коришћена је и средњеквадратна грешка “RMSE” ради поређења континуалних вредности.

	<i>Precision</i>	<i>Recall</i>	<i>F1</i>	<i>RMSE</i>
Град. Систем 1	0.395	<b>0.4</b>	<b>0.397</b>	<b>0.84</b>
Град. Систем 2	0.395	<b>0.4</b>	<b>0.397</b>	<b>0.84</b>
ГНМ 1	0.257	0.257	0.257	1.262
ГНМ 2	0.337	0.337	0.337	1.262
ГНМ 3	<b>0.351</b>	0.351	0.351	1.256

Табела 1. Перформансе креираних система

Као што се може видети из табеле 1, традиционални системи су надмашили системе креиране употребом граф неуронских мрежа. У случају креирања једноставног система за препоруку над малом количином података, традиционалне методе су се показале као боље, што се свакако може довести у питање уколико би се комплексност података и система повећала. У циљу побољшања добијених перформанси, могуће је унети одговарајуће измене у постојеће системе.

У случају традиционалног приступа, систем је могуће унапредити развојем комплексније логике у позадини хибридног система, односно начина агрегирања једноставнијих система који се налазе у његовој основи. Базни системи су такође креирани на врло једноставан начин и нуде простор за увођење додатне комплексности која би систем учинила прецизнијим. Поред тога, подаци обogaћени додатним особинама могли би знатно да побољшају перформансе, као и сама величина скупа за обучавање.

Системи засновани на граф неуронским мрежама такође могу да пруже боље перформансе у случају богатијих података и опширнијег скупа за обучавање.

#### 5. ЗАКЉУЧАК

С обзиром на велики успех граф неуронских мрежа у појединим областима, било је за очекивати да оне остваре боље перформансе од традиционалних метода. Међутим, у овом раду се показало да то не мора увек бити случај. Граф неуронске мреже, као веома комплексна структура, у неким случајевима уведе додатну сложеност у систем у којем то није потребно. Управо је то оно што показује и овај рад, где је над веома малом количином података једноставне структуре било довољно применити традиционалне методе система за препоруку. Наравно, како будућност доноси све веће количине података, али и захтева прецизније и унапређеније моделе, граф неуронске мреже ће наставити да се употребљавају у све већој мери, како у системима за препоруку, тако и у другим областима.

#### 6. ЛИТЕРАТУРА

- [1] GCN: Semi-Supervised Classification with Graph Convolutional Networks <https://arxiv.org/abs/1609.02907> [2016]
- [2] Neural Graph Collaborative Filtering <https://arxiv.org/abs/1905.08108> [2019]
- [3] GraphSage: Inductive Representation Learning on Large Graphs <https://arxiv.org/abs/1706.02216> [2017]
- [4] PinSage: Graph Convolutional Neural Networks for Web-Scale Recommender Systems <https://arxiv.org/abs/1806.01973> [2018]
- [5] Multi-Component Graph Convolutional Collaborative Filtering <https://arxiv.org/abs/1911.10699> [2019]
- [6] DiffNet++: A Neural Influence and Interest Diffusion Network for Social Recommendation <https://arxiv.org/abs/2002.00844> [2020]
- [7] GraphRec: Graph Neural Networks for Social Recommendation <https://arxiv.org/abs/1902.07243> [2019]
- [8] KGAT: Knowledge Graph Attention Network for Recommendation <https://arxiv.org/abs/1905.07854> [2019]
- [9] Deep Graph Library <https://www.dgl.ai/>
- [10] Graph Convolutional Matrix Completion <https://arxiv.org/pdf/1706.02263v2.pdf>

#### Кратка биографија:



**Јелена Достич** рођена је 17.6.1997. године у Новом Саду. Основне академске студије завршила је 2020. године на Факултету техничких наука у Новом Саду, на ком 2022. године брани и мастер рад из области **Електротехнике и рачунарства.**