SerialExperimentsOlga

Olga Yakobson

February 4, 2022
Version: My First Draft

Department of Mathematics,
Informatics and Statistics
Institute of Informatics

Artificial Intelligence and
Machine Learning

Bachelor's Thesis

SerialExperimentsOlga

Olga Yakobson

1. Reviewer	Prof. Dr. Eyke Hüllermeier Institute of Informatics LMU Munich
2. Reviewer	John Doe Institute of Informatics LMU Munich
Supervisors	Jane Doe and John Smith

February 4, 2022

Olga Yakobson
SerialExperimentsOlga
Bachelor’s Thesis, February 4, 2022
Reviewers: Prof. Dr. Eyke Hüllermeier and John Doe
Supervisors: Jane Doe and John Smith

LMU Munich
Department of Mathematics, Informatics and Statistics
Institute of Informatics
Artificial Intelligence and Machine Learning (AIML)
Akademiestraße 7
80799 Munich

Abstract

Abstract (different language)

Acknowledgement

Contents

1 Introduction

1.1 Motivation

Graphs are pretty useful for modelling real world systems as they are able to model entities and thier relationships.

Examples can be found across various domains such as Natural sciences (atoms + chemical bonds)

Graph Neural networks are a highly scalable class of models that can "learn" on graph structured data

TEST

1.1.1 Weisfeiler-Lehman and the graph Isomorphism test

There exist a variety of different and powerful graph neural network (GNN)

1.2 Research Questions

1.3 Structure

2 Related Work

> *Speech 100*
>
> — **Olga Yakobson**
> (Ph.Neutral)

2.1 Intro to GNN

GNN acess features of the neighbouring nodes using a mechanism called message passing. Message passing embedds into every node information about it's neighbourgood.GNNs can be classified by the corresponding message passing machanism e.g convolutional or attention.

More generally the message passing mechanism uses two functions, AGGREGATE and COMBINE [Xu+19]. The AGGREGATE function (max, sum...)

$$a_v^k = AGGREGATE^k(\{h_u^{(k-1)} : u \in \mathcal{N}_{(v)}\})\ , h_v^{(k)} = COMBINE^{(k)}(h_v^{(k-1)}, a_v^{(k)})$$

The UPDATE funcrtion then combines the each node uses the information from its neighbors to update its embeddings, thus a natural extension is to use the information from the neighbors of its neighbors(or second-hop neighbors) to increase its receptive field and become more aware of the graph structure. This is what makes the second layer of our GNN model.

Therefore by stacking k layers togheter, we can reach the k-hop neighbourhood. The combine function then combines the representation of the node with Various types of GNN have been used for a variety of machine learning tasks. Such tasks include

1. Link prediction:
2. Vertex classification & regression:

3. Graph classification & regression:

GNNs in particular graph convolutional networks (GCNs) tend to suffer from two main obstacles: overfitting and obersmoothing Over-fitting: weakens the generalization ability on small dataset

Over-smoothing impedes model training by isolating output representations from the input features with the increase in network depth.

To address these issues a variety of regularization techniques have been developed.

The aim is to apply GraphDropConnect (GDC) to two GNN architectures, namely GCN and graph isomorphism network (GIN)

2.2 Different GNN architectures and 1WL-Test

Graph Isomorphism Network $h_v^{(k)} = MLP^{(k)}((1+\epsilon^{(k)}) * h_v^{(k-1)} + \sum_{u \in \mathcal{N}(v)} h_u^{(k-1)})$

2.3 Regularization techniques

DropOut

$H^{(l+1)} = \sigma(\mathfrak{R}(A)(Z^{(l)} \odot H^{(l)})W^{(l)})$

DropEdge

$H^{(l+1)} = \sigma(\mathfrak{R}(A \odot Z^{(l)})H^{(l)}W^{(l)})$

Node Sampling

$H^{(l+1)} = \sigma(\mathfrak{R}(A)diag(z^{(l)})H^{(l)}W^{(l)})$

Graph DropConnect

$H^{(l+1)}[:,j] = \sigma(\sum_{i=1}^{f_t} \Re(A \odot Z_{i,j}^{(l)}) H^{(l)}[:,i] W^{(l)}[i,j])$

Graph Convolutional Network GCN as proposed by the authors [KW17] has the following layer-wise propaation rule

$H^{(l+1)} = \sigma(\tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} H^{(l)} W^{(l)})$
and is very broadly used for a variety of different tasks, such as for instance node Classification tasks. Despite not being as powerful as the GIN architecture, this architecture is sufficient for a lot of tasks, especially when there is no need for distinguishing different structures/ substructures of a graph and the prediction can be done with.

The authors claim, the model scales linearly in the number of graph edges and learns hidden layer representations that encode both local graph structure and features of nodes.

2.4 typical problems in GNN

Over-fitting: weakens the generalization ability on small dataset

Over-smoothing impedes model training by isolating output representations from the input features with the increase in network depth.

To address these issues a variety of regularization techniques have been developed.

2.5 Regularization

2.6 Conclusion

3 System

> *I would start with related work*
>
> — **C.Damke**
> (PhD)

3.1 System Section 1

Fig. 3.1.: Figure example: *(a)* example part one, *(c)* example part two; *(c)* example part three

3.2 System Section 2

Fig. 3.2.: Another Figure example: *(a)* example part one, *(c)* example part two; *(c)* example part three

3.3 System Section 3

3.4 Conclusion

4 Concepts: This text is here to test a very long title, to simulate the line break behavior, to show that an extremely long tilte also works

4.1 Concepts Section 1

4.2 Concepts Section 2

4.3 Concepts Section 3

4.4 Conclusion

5 Conclusion

5.1 System Section 1

5.2 System Section 2

5.3 Future Work

A Example Appendix

A.1 Appendix Section 1

Alpha	Beta	Gamma
0	1	2
3	4	5

Tab. A.1.: This is a caption text.

A.2 Appendix Section 2

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like "Huardest gefburn"? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

Alpha	Beta	Gamma
0	1	2
3	4	5

Tab. A.2.: This is a caption text.

Bibliography

[KW17] Thomas N. Kipf and Max Welling. "Semi-Supervised Classification with Graph Convolutional Networks". In: *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017 (cit. on p. 5).

[Xu+19] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. "How Powerful are Graph Neural Networks?" In: *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019 (cit. on p. 3).

List of Figures

List of Tables

Colophon

This thesis was typeset with $\LaTeX\,2_\varepsilon$. It uses the *Clean Thesis* style developed by Ricardo Langner. The design of the *Clean Thesis* style is inspired by user guide documents from Apple Inc.

Download the *Clean Thesis* style at `http://cleanthesis.der-ric.de/`.

Declaration

Ich, Olga Yakobson (Matrikel-Nr. 11591478), versichere, dass ich die Masterarbeit mit dem Thema SerialExperimentsOlga selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe. Die Stellen der Arbeit, die ich anderen Werken dem Wortlaut oder dem Sinn nach entnommen habe, wurden in jedem Fall unter Angabe der Quellen der Entlehnung kenntlich gemacht. Das Gleiche gilt auch für Tabellen, Skizzen, Zeichnungen, bildliche Darstellungen usw. Die Bachelorarbeit habe ich nicht, auch nicht auszugsweise, für eine andere abgeschlossene Prüfung angefertigt. Auf § 63 Abs. 5 HZG wird hingewiesen. München, 1. Februar 2023

Munich, February 4, 2022

Olga Yakobson