Application of the Lottery Ticket Hypothesis in NLP and Early Pruning (Proposal)

Anwendung der "Lottery Ticket"-Hypothese in NLP und frühem Pruning (Proposal)
Bachelor-Arbeit
Tim Unverzagt
KOM-type-number ???

TECHNISCHE
UNIVERSITÄT
DARMSTADT

Fachbereich Informatik
Fachbereich ??? (Zweitmitglied)

Fachgebiet Natural Language
Processing
|Gutachter|

Application of the Lottery Ticket Hypothesis in NLP and Early Pruning (Proposal)
Anwendung der "Lottery Ticket"-Hypothese in NLP und frühem Pruning (Proposal)

Bachelor-Arbeit
Studiengang: Computational Engineering
KOM-type-number ???

Eingereicht von Tim Unverzagt
Tag der Einreichung: dd. month yyyy

Gutachter: ???
Betreuerin: Anna Filighera

Technische Universität Darmstadt
Fachbereich Informatik
Fachbereich ??? (Zweitmitglied)

Fachgebiet Natural Language Processing (KOM)
|Gutachter|

Erklärung zur Abschlussarbeit gemäß § 23 Abs. 7 APB der TU Darmstadt

Hiermit versichere ich, Tim Unverzagt, die vorliegende Bachelor-Arbeit ohne Hilfe Dritter und nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die Quellen entnommen wurden, sind als solche kenntlich gemacht worden. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Mir ist bekannt, dass im Falle eines Plagiats (§38 Abs.2 APB) ein Täuschungsversuch vorliegt, der dazu führt, dass die Arbeit mit 5,0 bewertet und damit ein Prüfungsversuch verbraucht wird. Abschlussarbeiten dürfen nur einmal wiederholt werden.

Bei der abgegebenen Bachelor-Arbeit stimmen die schriftliche und die zur Archivierung eingereichte elektronische Fassung überein.

Darmstadt, den dd. month yyyy

Tim Unverzagt

Contents

Abstract

The abstract goes here...

1 Introduction

Hint:

This chapter should motivate the thesis, provide a clear description of the problem to be solved, and describe the major contributions of this thesis. The chapter should have a length of about two pages!

1.1 Motivation

What is the motivation for doing research in this area?

1.2 Problem Statement and Contribution

What is the problem that should be solved with this thesis?

1.3 Outline

How is the rest of this thesis structured?

2 Background

Hint:

This chapter should give a comprehensive overview on the background necessary to understand the thesis. The chapter should have a length of about five pages!

2.1 Basics of Neural Networks *WIP*

Neural networks are a part of most major AI-breakthrough in the last decade enabling computers to compete in fields formerly championed by humans.[1] They implement a statistical understanding of AI, which is to say that they try to find a specific model optimizing the likelihood of reproducing input-output pairs similar to some training data. The competing philosophy directly divines behaviour rules, frequently from expert knowledge, and as such is far less dependant from data. [citation needed]
For the former concept its model classes are the essential point of design. A multitude of properties maybe sought after in a model class of which a few important ones are:

- **Richness:**
  The diversity of single models in the class and thus the ability to fit a wide field of different input-output landscapes.[2]
  If a model class is inherently restricted the underlying relation between inputs and outputs might simply be beyond the expressive capabilities of all its models.
  In other words: If a model class is not rich enough all of its models will underfit the given training data.
- **Stability:**
  Tendency of similar models in the class to handle inputs in a similar way.
  If your model class shows unstable behavior defining a sensible way to search it for good models becomes difficult.
- **Interpretability of Models:**
  Ease of formulating knowledge out of any given model in the class.
  As fields exist in which statistical AI outperform experts the extraction of knowledge understandable and applicable by humans is of special interest.
- [citation needed]

[1]

- 2011: "Watson" of IBM defeats two former grand champions in "Jeopardy!" [LF11]
- 2011: "Siri" enables users to use natural language to interact with their phones [Aro11]
- 2015: A convolutional neural network classifies images from the ImageNet dataset more accurately than human experts [RDS$^+$15] [HZRS15]
- 2016: "AlphaGo" beats Lee Sedol, one of the world's strongest Go players [Gib16] [SSS$^+$17]

[2] More formally the richness of a model class can be described as the amount of different functions from the input-space to the output-space which can be expressed through a model of said class.

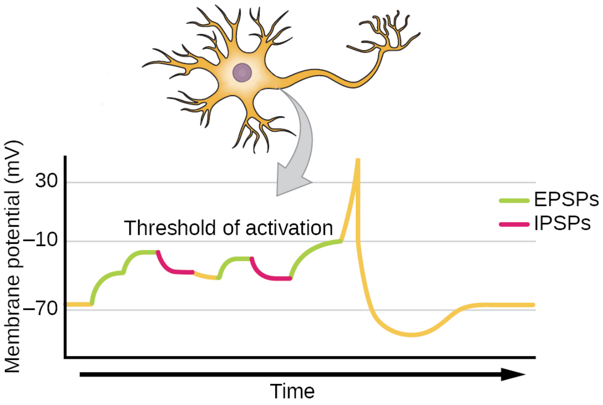

Figure 2.1.: Representation of a biological Neuron [CDC18] edited

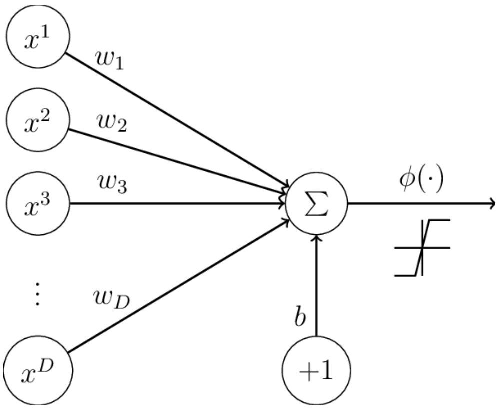

Figure 2.2.: Abstraction of a Neuron [DMK+12]

If one knows an entity that already performs well on a given task it is a sensible approach to design ones model class to reproduce its decision process. Humans usually are such entities for many tasks of interest to AI research so they are a natural source of inspiration. Neural networks essentially are simplified models of a human central nervous system.
The most basic building block of the human central nervous system is a neuron which can receive multiple stimuli and is able to produce an output if the combined stimulation exceeds a threshold.[citation needed] One such neuron and its stimulus measure are depicted in 2.1. Another functionality observed in nature is the ability of a neuron to strengthen the connection to any source of stimulus thus giving said source more influence on whether the neuron produces an output. [citation needed]

The canonical mathematical model of a neuron, as seen in 2.2, is defined as:

- **Inputs** x_i **:**
  All stimuli of a neuron are simply referred to as its inputs
- **Weights** w_i **:**
  The ability to assign importances is modelled as weights which are coupled to specific stimuli
- **Combined Weighted Inputs** $\sum_{i=1}^{n} w_i x_i$ **:**
  After the inputs are scaled by their according weight they superpose to form the total excitation of the neuron
- **Activation Function** $\Phi(\sum_{i=1}^{n} w_i x_i)$ **:**
  ...
- **Bias** b **:**
  ...

As an individual neurons is too simple to model any complex relations between inputs and outputs the next step is to aggregate multiple neurons. Figure 2.3 displays a few neurons coming together to form a simple fully-connected feed-forward network. [3]

- TODO:
- Issue of computational expense
- CNNs (and other forms of NN?)

2.2 The Lottery Ticket Hypothesis

- TODO:

[3] Inputs of neural networks are often called "features" and fully-connected networks are frequently referred to as "dense"

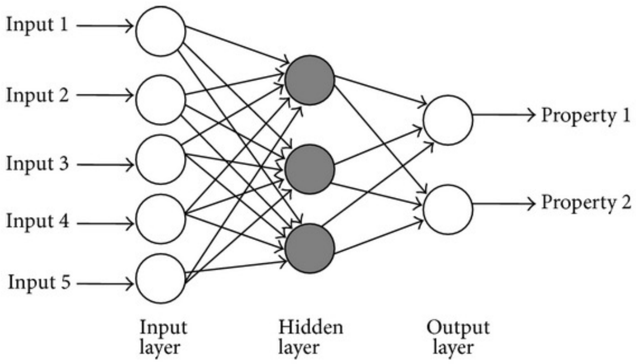

Figure 2.3.: A small fully-connected network
[Bel18]

- Issue of overloading on parameters
- Clarifying the task (image classification)
- Idea of trainable subnets

2.3 Basics of Natural Language Processing

- TODO:
- (?) Corpora
- Tokenizing
- Language Models
- (?) Handeling missing words in the Language model

2.4 Combining LMs & CNNs

- TODO:
- Interpreting the tensor representation of a sentence/document as an image to be classified
- (?) Validation through results
- (?) Handeling different sizes of inputs

3 Related Work

To quantify the goals previously defined the context of current research is needed. The importance of any work assuming an underlying architecture can not be correctly evaluated without knowledge about the quality of said architecture. As such this section shortly presents state-of-the-art approaches to the tasks relevant to this thesis. Additionally an overview over previous compression methods and their achievements is given. Furthermore the difference between image classification on the MNIST-dataset and topic classification on the Reuters-21578-dataset, as staple tasks of their respective fields, is specified. At the end of this section reasons are given why the collected related work does not yet satisfy the discussed goals.

3.1 Image Classification: MNIST

The MNIST-dataset contains 70.000 datapoints encoding 28x28-gray-scale-images of which 60.000 are designated for training and 10.000 for validation. The task is classification of a datapoint into one of ten digit-classes [YL].
Currently the best performing system combines fully connected, convolutional and long-short-term-memory networks achieving a 0.18% error-rate [KHB^{+}18]. Multiple approaches share second place with an error-rate of 0.20% correctly classifying 2 fewer examples [CMS12] [SNY15] [CC15] [VKC^{+}15] [HRFS16]. Restricted to simple models the lowest error-rate is achieved by a pruned fully-connected architecture with 2 hidden layers (300-100-Lenet) at 1.26% [DZ19]. [1]
While J. Frankle & M. Carbin do not provide exact values their figures indicate that the fully-connected Lenet-Architecture they study achieves roughly 98% accuracy on the test-data which translates to an error-rate of 2% and about 200 wrongly classified datapoints out of 10000 [FC18]. This result is reproducible with the source code provided alongside this thesis.

3.2 Pruning

Beginning around 1990 with M.C. Mozer & P. Smolensky [MS89] as well as LeCun et al. [LDS90] weights were being removed from neural networks after training them for a task. Shortly thereafter the idea of further training a pruned network was proposed [HS93] which became common practice over the next decade. While LeCun et al. describe a network compression factor of ×4, more recent works achieve a factor of ×9 to ×16.6 while loosing no or close to no accuracy [HPTD15] [LWL17].
In their paper on the Lottery-Ticket-Hypothesis (now LTH) J. Frankle & M. Carbin report pruning over 98,5% of weights in one of their networks while maintaining network capabilities which amounts to a compression rate of over ×50

3.3 Topic Classification: Reuters-21578

The Reuters-21578-dataset contains 21578 articles published by the Reuters News Agency in 1987 [Lew]. Reuters-21578 differs from MNIST in the sense that it lacks a few fundamental properties. In particular Reuters-21578 is not only multi-class but rather multi-label meaning that any one data point can satisfy multiple categories. Additionally there are categories in Reuters-21578 that have no associated positive example and even for all remaining ones the amount of samples is heavily skewed. In order to restore parts of the missing properties with minimal change to the dataset different subsets of Reuters-21578

[1] State-of-the-Art architectures are presented as described on https://paperswithcode.com/sota

have been chosen by different researchers.
F. Debole & F. Sebastiani [DS05] describe those subsets, starting out with the fact that close to half of the data points are unusable leaving 12,902 documents. 9,603 are marked for training and 3,299 for validation.[2] They also point out the different groups of categories used for classification:

- **R**(115)
  The group with the 115 categories with at least one positive training example.

- **R**(90)
  The group with the 90 categories with at least one positive training and test example.

- **R**(10)
  The group with the 10 categories with the most examples.

State of the art approaches to Reuters-21578 topic classification consist of Long-Short-Term-Memory [ARTL19] or mixed architectures [KHB+18] achieving an F1-score of 0.87 and 90.69% accuracy respectively.

3.4 Topic Classification: Convolutional Neural Networks For NLP

While the LTH is not yet extended to Long-Short-Term-Memory architectures there are approaches to NLP-tasks which are more closely aligned with computer vision. In a paper from 2014 [Kim14] Y. Kim describes how a simple convolutional neural network architecture, utilizing word embedding through language model, demonstrates capabilities similar to state-of-the-art approaches for various NLP-tasks.

3.5 Early Pruning

In a recent paper [LSZ+18] Z.Liu et al. observe that if pruned networks are trained with randomly reinitialized weights instead of fine-tuning their previous ones they retain from the original network, the pruned networks keep their capabilities. They conclude that said weights can not be essential to a pruned networks quality, contrary to prior common belief. Thus Z.Liu et al. claim that the architecture of pruned networks is responsible for its capabilities and furthermore that pruning can be interpreted as a kind of network architecture search .
After the effectiveness of pruning is established and its interpretation as network architecture search becomes available there is a legitimate question whether all the weights in a network are really necessary for all of the training. In a paper of Y. Li & W. Zhao & L. Schang from early 2019 [LZS19] they describe a method named IPLT to prune common convolutional network architectures at the filter level and especially before convergence. Thus they do not only compress the networks by a factor of ×10 but also speed up training by a similar magnitude.

3.6 Additions to the Lottery Ticket Hypothesis

Even though the Lottery-Ticket-Hypothesis was only proposed earlier this year additional papers on the topic exist. In a paper from June 2019 J. Frankle & M. Carbin et al. [FDRC19] expand their method to find winning tickets on deep convolutional network architectures that proved difficult before. They attribute this achievement to the decision of not returning to the very first state of the network but to one a few iterations into training.

[2] While different training-splits were proposed for Reuters-21578 "ModApté" has become the canonical choice

Additionally H. Zhou et al. [ZLLY19] document an ablation study on the phenomenon of lottery tickets. They reaffirm the initially naive magnitude-based pruning and describe "supermasks" that improve accuracy when applied to the initial network even without additional training. Finally they find that a replacement of all weights in the pruned network by a constant with same sign as said weights does not significantly influence the networks capabilities. As such H. Zhou et al. conclude that the sign of weights are the essential property for such neural networks.

3.7 Task in context of Related Work

To conclude this chapter the given information about related work is interpreted in regards to the three defined subtasks:

- **1. Reproduction**
  Reproduction of the LTH is highly independent of the related work mentioned so far.[3]

- **2. Extending to NLP**
  Even though the LTH has not yet been applied to Long-Short-Term-Memory architectures and doing so is beyond the scope of this thesis the question whether it can hold for a more disorderly dataset at all is still open and of interest. The approach described in 3.4 is generally applicable to the Reuters-21578 dataset as well as already performing well on other NLP-tasks while also being compatible with the current Application of the LTH.

- **3. Early pruning**
  While IPLT [LZS19] already provides an early pruning approach the LTH has shown capabilities of much stronger compression. While this compression does not yield any speed-up yet because unstructured sparse layers need a special infrastructure to accelerate, verification of so significant compressions rates found by the LTH early in training might motivate such infrastructures.

[3] Side note: While tensorflow 1.x source code exists at https://paperswithcode.com, for the first LTH-paper, tensorflow 2.0 is much easier to operate and code for so rebuilding the framework is still valuable.

4 Design

Hint:

This chapter should describe the design of the own approach on a conceptional level without mentioning the implementation details. The section should have a length of about five pages.

4.1 Requirements and Assumptions

4.2 System Overview

4.2.1 Component 1

4.2.2 Component 2

4.3 Summary

5 Implementation

Hint:

This chapter should describe the details of the implementation addressing the following questions:

1. What are the design decisions made?
2. What is the environment the approach is developed in?
3. How are components mapped to classes of the source code?
4. How do the components interact with each other?
5. What are limitations of the implementation?

The section should have a length of about five pages.

5.1 Design Decisions

5.2 Architecture

5.3 Interaction of Components

5.4 Summary

6 Data Sets

7 Evaluation

Hint:

This chapter should describe how the evaluation of the implemented mechanism was done.

1. Which evaluation method is used and why? Simulations, prototype?
2. What is the goal of the evaluation? Comparison? Proof of concept?
3. Wich metrics are used for characterizing the performance, costs, fairness, and efficiency of the system?
4. What are the parameter settings used in the evaluation and why? If possible always justify why a certain threshold has been chose for a particular parameter.
5. What is the outcome of the evaluation?

The section should have a length of about five to ten pages.

7.1 Goal and Methodology

7.2 Evaluation Setup

7.3 Evaluation Results

7.4 Analysis of Results

8 Conclusions

Hint:

This chapter should summarize the thesis and describe the main contributions of the thesis. Subsequently, it should describe possible future work in the context of the thesis. What are limitations of the developed solutions? Which things can be improved? The section should have a length of about three pages.

8.1 Summary

8.2 Contributions

8.3 Future Work

8.4 Final Remarks

Bibliography

[Aro11] Jacob Aron. How innovative is apple's new voice assistant, siri? *New Scientist*, 212(2836):24, 2011.

[ARTL19] Ashutosh Adhikari, Achyudh Ram, Raphael Tang, and Jimmy Lin. Rethinking complex neural network architectures for document classification. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4046–4051, 2019.

[Bel18] Soufiane Belharbi. Neural networks regularization through representation learning, 07 2018.

[CC15] Jia-Ren Chang and Yong-Sheng Chen. Batch-normalized maxout network in network. *CoRR*, abs/1511.02583, 2015.

[CDC18] Mary Ann Clark, Matthew Douglas, and Jung Choi. *Biology 2e*. OpenStax, 2018.

[CMS12] Dan C. Ciresan, Ueli Meier, and Jürgen Schmidhuber. Multi-column deep neural networks for image classification. *CoRR*, abs/1202.2745, 2012.

[DMK$^+$12] Jelena Djuris, Djordje Medarević, Marko Krstić, Ivana Vasiljević, Ivana Aleksić, and Svetlana Ibrić. Design space approach in optimization of fluid bed granulation and tablets compression process. *TheScientificWorldJournal*, 2012:185085, 07 2012.

[DS05] Franca Debole and Fabrizio Sebastiani. An analysis of the relative hardness of reuters-21578 subsets. *Journal of the American Society for Information Science and Technology*, 56(6):584–596, 2005.

[DZ19] Tim Dettmers and Luke Zettlemoyer. Sparse networks from scratch: Faster training without losing performance. *CoRR*, abs/1907.04840, 2019.

[FC18] Jonathan Frankle and Michael Carbin. The lottery ticket hypothesis: Training pruned neural networks. *CoRR*, abs/1803.03635, 2018.

[FDRC19] Jonathan Frankle, Gintare Karolina Dziugaite, Daniel M. Roy, and Michael Carbin. The lottery ticket hypothesis at scale. *CoRR*, abs/1903.01611, 2019.

[Gib16] Elizabeth Gibney. Google ai algorithm masters ancient game of go. *Nature News*, 529(7587):445, 2016.

[HPTD15] Song Han, Jeff Pool, John Tran, and William Dally. Learning both weights and connections for efficient neural network. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 1135–1143. Curran Associates, Inc., 2015.

[HRFS16] Seyyed Hossein HasanPour, Mohammad Rouhani, Mohsen Fayyaz, and Mohammad Sabokrou. Lets keep it simple, using simple architectures to outperform deeper and more complex architectures. *CoRR*, abs/1608.06037, 2016.

[HS93] Babak Hassibi and David G Stork. Second order derivatives for network pruning: Optimal brain surgeon. In *Advances in neural information processing systems*, pages 164–171, 1993.

[HZRS15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[KHB+18] Kamran Kowsari, Mojtaba Heidarysafa, Donald E. Brown, Kiana Jafari Meimandi, and Laura E. Barnes. Rmdl: Random multimodel deep learning for classification. In *Proceedings of the 2Nd International Conference on Information System and Data Mining*, ICISDM '18, pages 19–28, New York, NY, USA, 2018. ACM.

[Kim14] Yoon Kim. Convolutional neural networks for sentence classification. *CoRR*, abs/1408.5882, 2014.

[LDS90] Yann LeCun, John S Denker, and Sara A Solla. Optimal brain damage. In *Advances in neural information processing systems*, pages 598–605, 1990.

[Lew] David D. Lewis. Reuters-21578. http://www.daviddlewis.com/resources/testcollections/reuters21578/.

[LF11] Adam Lally and Paul Fodor. Natural language processing with prolog in the ibm watson system. *The Association for Logic Programming (ALP) Newsletter*, 2011.

[LSZ+18] Zhuang Liu, Mingjie Sun, Tinghui Zhou, Gao Huang, and Trevor Darrell. Rethinking the value of network pruning. *CoRR*, abs/1810.05270, 2018.

[LWL17] Jian-Hao Luo, Jianxin Wu, and Weiyao Lin. Thinet: A filter level pruning method for deep neural network compression. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[LZS19] Yue Li, Weibin Zhao, and Lin Shang. Really should we pruning after model be totally trained? pruning based on a small amount of training. *CoRR*, abs/1901.08455, 2019.

[MS89] Michael C Mozer and Paul Smolensky. Skeletonization: A technique for trimming the fat from a network via relevance assessment. In *Advances in neural information processing systems*, pages 107–115, 1989.

[RDS+15] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, Dec 2015.

[SNY15] Ikuro Sato, Hiroki Nishimura, and Kensuke Yokoi. APAC: augmented pattern classification with neural networks. *CoRR*, abs/1505.03229, 2015.

[SSS+17] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. *Nature*, 550(7676):354, 2017.

[VKC+15] Francesco Visin, Kyle Kastner, Kyunghyun Cho, Matteo Matteucci, Aaron C. Courville, and Yoshua Bengio. Renet: A recurrent neural network based alternative to convolutional networks. *CoRR*, abs/1505.00393, 2015.

[YL] Christopher J.C.Burges Yann LeCun, Corinna Cortes. The mnist database. http://yann.lecun.com/exdb/mnist/.

[ZLLY19] Hattie Zhou, Janice Lan, Rosanne Liu, and Jason Yosinski. Deconstructing lottery tickets: Zeros, signs, and the supermask. *CoRR*, abs/1905.01067, 2019.

Appendices

A A history of neural networks

- 1. wave: 1955-1970
- 2. wave: 1985-2000 ?
- 3. wave: ???

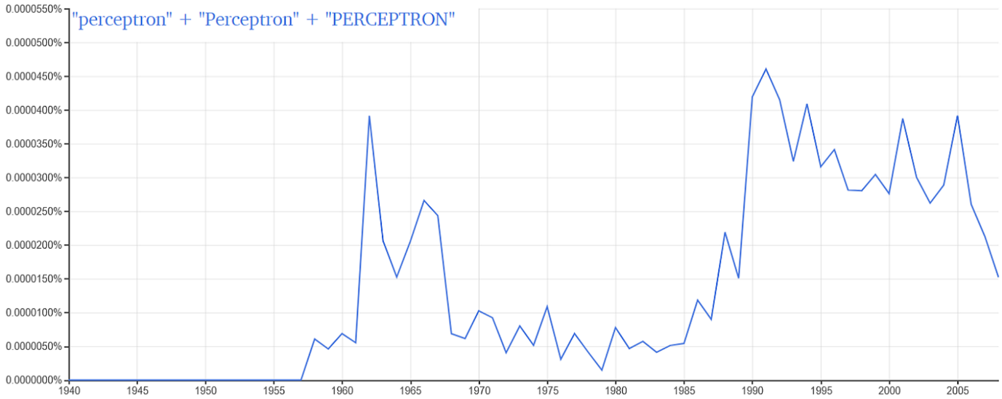

Figure A.1.: Relative amount of occurences of the word "Perceptron" in published books between 1940 and 2009

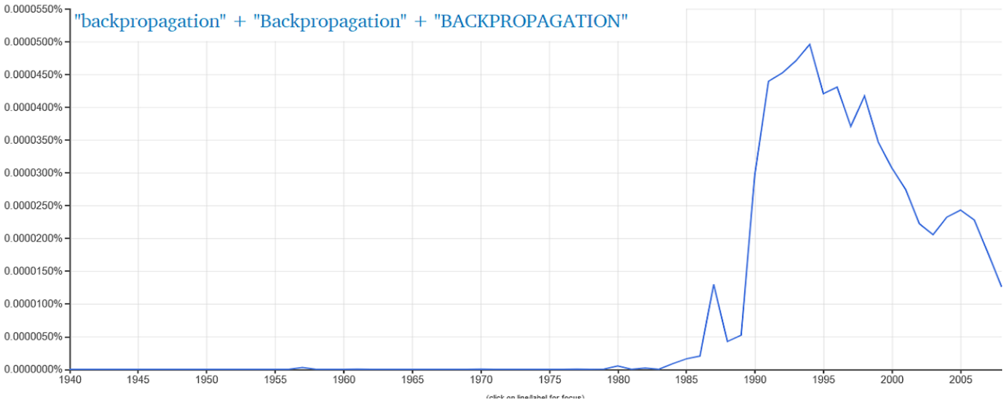

Figure A.2.: Relative amount of occurences of the word "Backpropagation" in published books between 1940 and 2009