

What did I do?

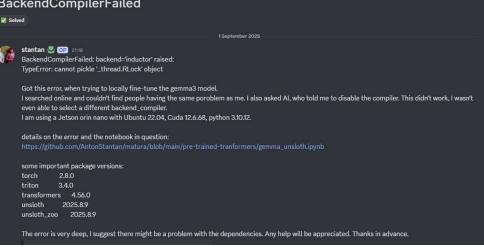
Saturday, 7 June 2025 12:51

Date	What did I do?	Time in hours
Tuesday, 8 April 2025	started systematic literature review, set up the overleaf document I'll be working on. Thinking about the main question of the review, I've got a general idea, but still unsure. I was able to narrow down the search on google scholar to 27 results with some keywords.	
Wednesday, 9 April 2025	I've decided I'm going to be doing multiple searches with different keywords or approaches(sorting by relevancy). I've picked 5 databases which I'll be searching with the keywords. THE NVIDIA GPU IS GONNA ARRIVE TOMORROW!!!	
Thursday, 10 April 2025	The package arrived! I'm still setting up the os system, but so far, so good. The Jetson orin Nano looks amazing.	
Friday, 11 April 2025	I've made a precise workflow for the research part of my lit. review. I also ran into some troubles with the WoS and scopus websites.	
Saturday, 12 April 2025	The Jetson orin nano setup is driving me crazy. When I first installed docker on it, I think I did something wrong, because a lot of errors came up. Neither I, nor the AI's I've asked have any clue on how to fix the issue. I think I might have to start over. On the other front: I decided on just using the 3 websites that did work and I picked out the 15 best sounding titles out of their top 25 results.	
Sunday, 13 April 2025	after having looked through the titles of the different websites, only ACM and ScienceDirect had good papers. Altogether I found 18 promising titles, which I then cut down to 3 papers. Tomorrow I will read those. I had to deviate from the workflow due to the circumstances.	
Monday, 14 April 2025		
Tuesday, 15 April 2025		
Wednesday, 16 April 2025		
Thursday, 17 April 2025		
Friday, 18 April 2025		
Saturday, 19 April 2025		
Sunday, 20 April 2025		
Monday, 21 April 2025		
Tuesday, 22 April 2025		
Wednesday, 23 April 2025		
Thursday, 24 April 2025		
Friday, 25 April 2025		
Saturday, 26 April 2025		
Sunday, 27 April 2025		
Monday, 28 April 2025		
Tuesday, 29 April 2025		
Wednesday, 30 April 2025		
Thursday, 1 May 2025		
Friday, 2 May 2025	Ich habe heute angefangen das erste (auch längste) Papier zu gründlich lesen. Ich habe schon vorher einbisschen daran gelesen, aber nicht gründlich und ohne Notizen/Markierungen.	1.75
Saturday, 3 May 2025	weiter gelesen an der ersten Studie über XAI und dessen Kriterien.	2
Sunday, 4 May 2025		
Monday, 5 May 2025		
Tuesday, 6 May 2025		
Wednesday, 7 May 2025		
Thursday, 8 May 2025	1. Studie weitergelesen und Orte welche ich gerne Zitieren würde markiert. Mit zweiter Studie über den Hawkes-Prozess angefangen. Problem: Ich bin mir nicht sicher ob ich den Hawkes prozess anwenden werde können. Obwohl sie es ein FNN nennen, tönt das Model welches sie in der Studie als Beispiel verwendet haben wie ein CNN.	1.5
Friday, 9 May 2025	1 Stunde die Studie über den Hawkes probiert zu verstehen. Ich glaube aber, dass sie viel zu kompliziert ist. Ich werde einfach die Anwendung wahrscheinlich benutzen. Dannach habe ich an meinen Jetson gearbeitet und ich habe einen Setup guide benutzt, jetzt sollte mein Problem von früher wahrscheinlich behebt sein.	4
Saturday, 10 May 2025	Nvidia Jetson gearbeitet mit docker jetson-containers	4
Sunday, 11 May 2025	Jetson plus minus zum arbeiten gebracht. Ganzen tag sich damit beschäftigt lol.	12

y, 11 May 2025		
Monda y, 12 May 2025		
Tuesd ay, 13 May 2025		
Wedne sday, 14 May 2025		
Thursd ay, 15 May 2025		
Friday, 16 May 2025	Heute fertig mit Papieren lesen. Ich fange heute am abend/morgen mit dem Latex Dokument an.	1
Saturd ay, 17 May 2025	Erste 5 Seiten des Literaturberichts geschrieben.	7
Sunda y, 18 May 2025	Section 3 geschrieben, aber nicht sehr hochwertig, es wird sehr wahrscheinlich bei den Erklärungen von Konzepten viel korrigiert werden müssen	8
Monda y, 19 May 2025	wrote the first half of section 4. again I'm unsure in a lot of places, so there'll be a lot to correct I guess	4
Tuesd ay, 20 May 2025	didn't have a lot of time today, but I was able to finish section 4	2
Wedne sday, 21 May 2025	Turned out my questions were wrong, so I'm gonna have to redo everything, now. Gonna be quite the time crunch	5
Thursd ay, 22 May 2025	I found some good sources and formulated a vague structure of the new literature review.	2.5
Friday, 23 May 2025	I finished the introduction and first section of the second lit review.	5
Saturd ay, 24 May 2025	Wrote a page for section 3	2.5
Sunda y, 25 May 2025		
Monda y, 26 May 2025		
Tuesd ay, 27 May 2025	I finished section 3 and learned a lot actually. I somewhat have an idea of what to write about in section 4 now.	2
Wedne sday, 28 May 2025	started section 4 but didn't really get anywhere. I need a bit more research. Also my latex document is falling apart. I should probably switch to biblatex. Also I CANNOT work today, I feel like the inner jobless me came out today. I like sleep for about 10 - 11h so sleepy through out the day. I recorded, that I worked for only 3h, but in reality it was more like 6h and the rest of the day was procrastinating or distractions.	3
Thursd ay, 29 May 2025	finished the section 4 part about previously mentioned issues. Did a LOT of research	3.5
Friday, 30 May 2025	I enjoyed the good weather, didn't work that much a part from reading and organizing all the literature. I wrote down the first approach in how to improve the model for arithmetic tasks. The goofy one with the virtual abacus thingy	3
Saturd ay, 31 May 2025	I was in the mountains today, so I didn't really do that much. Still I wrote down 4 approaches in total now	2
Sunda y, 1 June 2025	FINISHED. Well not totally, but mainly. Dedicated the whole GODDAMN DAY to this review. I just wanna go and watch some insta reels. The takeaways section needs to be improved. But that's a problem OF THE FUTURE ME!!!	8
Monda y, 2 June 2025		
Tuesd ay, 3 June 2025	Finished up the grammatical correction of the lit. review. I also wrote an abstract and other stuff	3.5
Wedne sday, 4 June 2025	I've decided to use Google Colab, for the compatibility with the Jupyter notebooks I set up for the NVIDIA Jetson GPU earlier. I fiddled around a bit and got my first prototype running.	5
Thursd ay, 5 June 2025	I added bootstrapping to the model. I drew a graph of expressions not in training data, showing the accuracy. This shows, that the program indeed is learning something atleast. Need to do interpretation of it.	5
Friday, 6 June 2025	I made test data for numbers outside the training range. Predicted results for that test data with the bootstrapped model. I drew a graph outside of training range. Refined the previous tokenizer function.	3.5
Saturd ay, 7 June 2025	I implemented test data with more terms == longer expressions. Generalized the tokenizer function, so it works with longer expressions too. Collected bootstrapped predictions. And drew a graph showing number of terms vs deviation, with a trendline.	4
Sunda y, 8 June 2025	I evaluated the longer expressions - test data, and drew a graph for the correlation between length of expression and MAE. I also did a lot of debugging. I set about defining a benchmark for models with different architectures. (the 3 different MSE proportionally to a base performance.) For this benchmark I had to implement pairs and stuff to reorganize the prediction results.	5.5
Monda y, 9 June 2025	Spent my day programming the generation of different models with different architectures. I think I succeeded, although I didn't really properly evaluate my results yet. The time it takes to train different models (36) is roughly an hour, this will make it difficult for me to work in school in the near future. P.S. The padding with 0.5 worked way better, I implemented that.	7
Tuesd ay, 10 June 2025	At school I worked on my code, found some errors and changed them (neuron_steps). I couldn't squeeze in the time for a calculation today. I met up with my dad, he corrected my text for grammatical errors (though not all of them), as well as gave me an interesting article which I ended up citing additionally to prove a point in section 2. I asked Smits.	1
Wedne sday, 11 June 2025	I re-ran my calculations with an additional calculation of a benchmark for the models. I have a couple of good looking graphics for the presentation now. The cherry on top being the heatmap. At school I corrected my literature study and added some comments where needed. After some final polishing up, I'm ready to submit my results.	4
Thursd ay, 12 June 2025	I also set up a GitHub page yesterday and restructured and re-organized it a little today (I'm still getting used to the browser-GitHub. I previously only used it to download stuff from the internet through the terminal). https://github.com/AntonStantan/matura/tree/main I also finished a rough layout for the topics I want to discuss in the presentation. I feel like I'll speak more than 15 min	2
Friday, 13 June 2025	Formulated a couple of questions to Mr. Schneider.	
Saturd ay, 14	I started and finished about half of the power point. I didn't do any design yet.	2

June 2025		
Sunda y, 15 June 2025	A lit work during school on the power point.	0.5
Monda y, 16 June 2025	Finished the Power Point. I don't have a layout though.	1.5
Wedne sday, 18 June 2025	Polished up my power point and prepared for the presentation.	3
Thurs ay, 19 June 2025		
Friday, 20 June 2025		
Satur ay, 21 June 2025	During the last couple of days I thought about the proposal made by mrs. Smits and I think she might be right. After having computed it, I'm left with a lot of interesting data to unpack. Today, I drew a couple of graphs and let her know.	2
Sunda y, 22 June 2025		
Monda y, 23 June 2025		
Tuesd ay, 24 June 2025	I implemented the suggestions provided by J Smits and drew some graphs in order to decide whether to change my benchmark. Specifically, I will now be using seaborn for better graphs with confidence intervals.	3
Wedne sday, 25 June 2025	I finalized the graphs with relative errors. And tried a new benchmark with relative errors. The results looked more promising. I also made some new entries on GitHub. I talked with mr. Schneider and stated 2 Questions.	
Thurs ay, 26 June 2025		
Friday, 27 June 2025	Today I explored introducing Drop-Out: It showed higher MSEs: From https://colab.research.google.com/drive/1fek7nCQz7h8A-xdICG_BtXoD1NQNa#scrollTo=tFCh43aAvp . As well as a heatmap with lower benchmarks, when comparing to values of a basemodel without dropout (0.3). I also started the documentation file.	1
Satur ay, 28 June 2025		
Sunda y, 29 June 2025	Today I set off improving the tokenizer by adding a positional encoding. I managed to code a working sinusoidal positional encoder. It showed weaker results then, without a positional encoder, which is very weird. I set off debugging and didn't find any issues, still it's difficult to believe.	6
Monda y, 30 June 2025	I found the problem, there was an error when combining two arrays. I then re-drew some graphs and observed the positional encoder improving the performance, after changing the algorithm. I tried reducing over-fitting with L1 regularization, it didn't help and made the models perform worse.	6
Tuesd ay, 1 July 2025	I confirmed my suspicion that the L2 regression had the same negative effect as L1 and Drop-Out. I decided against trying different batch sizes, because in theory this shouldn't have any impact.	0.5
Wedne sday, 2 July 2025		
Thurs ay, 3 July 2025		
Friday, 4 July 2025		
Satur ay, 5 July 2025		
Sunda y, 6 July 2025		
Monda y, 7 July 2025		
Tuesd ay, 8 July 2025	I properly downloaded and setup the jupyterlab environment locally and trained a first prototype of the recurrent Neural Networks.	3.5
Wedne sday, 9 July 2025	I did some research into the functionality of LSTM and GRU RNNs. I drew the graphs just like I did earlier with the FNN. I also made some minor improvements in the code. I couldn't properly work today, because I was a little sick. (and still am)	3
Thurs ay, 10 July 2025	Looked over how to import variables from other .py scripts. With testing.	1
Friday, 11 July 2025	Implemented the generation of different RNN models with different amounts of neurons and layers. The Data is later also converted into usable np.array. I got as far as to drawing the heatmap, but there is a problem with the array I receive after the long calculations of the many models. I get 30 models, when really I should be expecting 28.	4
Satur ay, 12 July 2025	Fixed the error. The first heatmap of RNNs has been generated. Surprisingly the SimpleRNN doesn't improve performance of arithmetical tasks.	2.5
Sunda y, 13 July 2025	I generated the Heatmaps of LSTMs and researched about attentional LSTMs	3.5
Monda y, 14 July 2025	I generated GRUs Heatmap overnight and later researched about how I can add attention to my LSTMs. Watched a bunch of tutorial videos on youtube and read up on some stuff too. Only research about the next attention with LSTM model.	4
Tuesd ay, 15 July 2025		
Wedne sday, 16 July 2025	Research and first prototype. I copied and slightly modified the code from a website. (geeksforgeeks). I also researched a lot today. The odd thing is that the results of the LSTM with the attention seem to be worse actually.	4
Thurs ay, 17 July 2025	I reorganized and drew the heatmap, but I don't know how to move on. The heatmap was worse, as expected	
Friday, 18 July 2025	I fiddled around with the code of the attention layer, but no progress. In the evening I just wrote a libit about the definition of the RNN and the formulas behind it and stuff.	2.5
Saturd ay		

ay, 19 July 2025																						
Sunda y, 20 July 2025																						
Monda y, 21 July 2025																						
Tuesd ay, 22 July 2025	I'm writing this at 2am so technically it's also what I did tomorrow. After watching even more tutorials and explanations online, as well as chatting with Gemini about it, I rewrote the LSTM with attention, since last time I wrote the code, I copied most of it from a website, and I wasn't able to properly understand it. Today I fully understand "everything" about this new model. Yet, unsurprisingly it showed the same scores as the ones I got from the website.																					
Wedne sday, 23 July 2025	I reorganized everything and finally found a better workflow for github lol. I also discovered, that I wasn't using the validation data I defined earlier. I fixed that too.																					
Thursd ay, 24 July 2025	I tried out jules. Its really cool. I also did a bit of research on transformers, I'm ready to begin tomorrow.																					
Friday, 25 July 2025	I watched tutorials during the day, and then worked a bit more on the transformer at night roughly 10 - 4am.																					
Saturd ay, 26 July 2025	I did some more research on transformers, had a couple of conversations with gemini, as well as generated the first notebook, which should serve as a sort of introductory transformer model, based on the one from the first paper "attention is all you need". (I also wrote a couple of cells including the positional encoder, (the one the paper used with sin and cos.)																					
Friday, 25 July 2025	This is just for better understanding on my part, since it worked fairly well with RNNs. I also watched tutorials for class type objects in python, ill need that for transformers I'm sure.																					
Sunda y, 27 July 2025	I watched tutorials during the day, and then worked a bit more on the transformer at night roughly 10 - 4am.																					
Saturd ay, 26 July 2025	I FINISHED the transformer (sunglasses emoji). I did A LOT today and I'll skip over a lot. In summary I coded a many to one transformer, modelled after the seq2seq one in the "attention is all you need" paper. Effectively, I just used the Encoder only. Another thing, is that I coded the Multi head attention basically from scratch, which really helped me understand the workings of a transformer.																					
Sunday, 27 July 2025	2 Big problems came up when I first fitted the model: - 1st: it takes about 100x as long to process one epoch of a transformer, then of an attentional RNN. And if you remember, It took my cpu 5.5h of computations to draw the heatmap for the attentional-RNN consisting of 330 model. - 2nd: The transformer isn't learning. It's just guessing around 0 and satisfies itself with that.																					
Sunda y, 28 July 2025	Tried to install tensorflow, but to no avail.																					
Monda y, 29 July 2025	Still no success																					
Tuesd ay, 29 July 2025	Finally was able to install tensorflow with gpu support. Kinda destroyed my jetpack in the process tho. I had some issues with firefox just not opening up. Turns out I had to install an older version of snapd to solve the issue. Afterward I was able to setup the jupyterlab and run my scripts on the jetson nano. This turned out to have 17-times the efficiency.																					
Wedne sday, 30 July 2025	I looked into the issue with encoders/embeddings for previous notebooks. I experimented around a little, but like before the exact same positional encoding, which is used in the transformer(cos/sin) doesn't seem to help, the results are worse even. And the traditional embedding/encoding part isn't necessary, since I am working with number (and only a select few symbols).																					
Thursd ay, 31 July 2025	Basically after some investigation I came to the conclusion to leave it as is, because 1st it should theoretically be enough of an encoder as is, and 2nd because my work aimed to focus on the architecture anyways.																					
Friday, 1 August 2025																						
Saturd ay, 2 August 2025																						
Sunda y, 3 August 2025																						
Monda y, 4 August 2025																						
Tuesd ay, 5 August 2025																						
##### ##### ##### ##### #																						
Thursd ay, 7 August 2025																						
Friday, 8 August 2025																						
Saturd ay, 9 August 2025	Final 2 Days before school: The guests left yesterday, over the course of the last week I did some work here and there on hotel Wan, (I setup jupyterlab weird, and it requires wifi.) I implemented these features into the transformer model, as advised by websites and AIs: - Learning Rate Scheduler with linear warmup and cosine decay.(biggest change I'd say.) - AdamW optimizer - Experimented with different batchsizes - Gradient clipping (learning rate scheduler behaves a little goofy at times) - shuffle in the training data (this is so basic, I think I could/should add this to all the previous models aswell)																					
Sunda y, 10 August 2025	Today I had to reinstall the correct tensorflow again because it updated, but it was much easier since I have the wheel I used the last time. Later I tackled the 0-guessing problem again. I had great success when decreasing the model complexity, this, I think, is the correct solution to the problem. (either that or increasing the dataset size.) I also tweaked and played around with the optimizer's parameters. With all that, as well as some minor adjustments, I was able to reach new mse lows.																					
Monda y, 11 August 2025																						
Tuesd ay, 12 August 2025	I played around with the variables some more, then a LLM suggested I use keras-tuner, and I did. I tried to let the programm do calculations over night, but it shut itself off. -> I wrote transformer2.ipynb Was only able to run 3 trials.																					
##### ##### ##### ##### #																						
Thursd ay, 14 August 2025	Since I worked from school today (without access to the GPU), I decided to implement the keras-tuner for the FNNs, to test its effectivity. I found it performed quite poorly. After 20mins of training the best model it found consisted of 1 layer and 15 neurons. That layer was very noisy and was slightly overfitting. A weird thing happened where the models with small amounts of parameters would start with very small MSEs. Edit: I watched the video provided by my teacher on how to plan and protocol correctly.																					
Friday, 15 August 2025	Today, I firstly made a plan for the remaining time and submitted it. Afterwards I worked on investigating the weird behaviour of the smaller FNNs. It turned out to be due to the glorot_unifir weights initialization. This is weird because earlier I was encountering a similar error with the transformers (transformer0). And I confirmed that when weights are initialized at 0, this is the very lowest point of the local minimum.																					
	Verification: github -> transformer3.ipynb and submitted MaturaPlan.png																					
	Later I came home and tried to use my GPU to train a couple of models with keras-tuner, but the gpu keeps crashing. This is a problem.																					
	Verification: Trial 7 Complete [00h 02m 19s] val_loss: 25.40894882021484 Best val_loss So Far: 17.388446887861328 Total elapsed time: 00h 14m 44s Search: Running Trial #8 <table border="1"><thead><tr><th>Value</th><th>[Best Value So Far]</th><th>Hyperparameter</th></tr></thead><tbody><tr><td>8</td><td>[2]</td><td>num_heads</td></tr><tr><td>128</td><td>[32]</td><td>d_model</td></tr><tr><td>3</td><td>[5]</td><td>num_layers</td></tr><tr><td>256</td><td>[128]</td><td>d_ff</td></tr><tr><td>False</td><td>[False]</td><td>dropout</td></tr><tr><td>3.6841e-06</td><td>[6.3345e-05]</td><td>peak learning rate</td></tr></tbody></table> Epoch 1/25 Trial 7 is the farthest I was able to get.	Value	[Best Value So Far]	Hyperparameter	8	[2]	num_heads	128	[32]	d_model	3	[5]	num_layers	256	[128]	d_ff	False	[False]	dropout	3.6841e-06	[6.3345e-05]	peak learning rate
Value	[Best Value So Far]	Hyperparameter																				
8	[2]	num_heads																				
128	[32]	d_model																				
3	[5]	num_layers																				
256	[128]	d_ff																				
False	[False]	dropout																				
3.6841e-06	[6.3345e-05]	peak learning rate																				
	Today I decided to train the keras-tuner with my laptop instead because it has more RAM, and sure enough after an hour of calculations very promising hyperparameters were discovered.																					

	##### Verification: github -> gemma_unsloth.ipynb ##### ##### # Thursd ay, 28 August 2025	I ran the code on the nvidia gpu. It didn't work. I spent the whole time navigating around in the Jetson environment. It was very exhausting and there isn't a lot to say. Firstly I had to install the correct pytorch version. Keeping in mind all the dependencies. I also had to upgrade my jetpack version. And then install a unsloth version that worked. Verification: The tutorial that worked for the pytorch installation. So that it will work with my gpu: https://docs.pytorch.org/TensorRT/getting_started/jetpack.html	5
Friday, 29 August 2025	Saturday, 30 August 2025		
Sunday, 31 August 2025	Finally found some time to work some more. I was able to resolve the dependencies and built up triton from its source. After some small adjustments to the code itself (training dataset import) I encountered another Error, the biggest one I've ever seen. I think it's a problem with the pytorch compiler. I tried deactivating the compiler, it didn't work. So I'm stuck again basically. P.S. I worked for 2 more hours, and I've managed to setup remote access to my jetson. I used ssh tunnel with an -L flag. This allows me to open jupyter lab in my laptops browser from a different network, this means I'll be able to work in school tomorrow. Verification: github -> gemma_unsloth.ipynb	7	
Monday, 1 Septem ber 2025	I tackled the problem from school, and later again from home, I asked for help online, I asked my teachers, my mother and even the official discord. There they told me that the ARM architecture my Jetson is based on isn't supported. This means I can forget using unsloth, I'll switch to Huggingface with accellerate.  BackendCompilerFailed startan [] BackendCompilerFailed:backend.inductor raised: TypeError: cannot pickle '_thread.RLock' object Got this error, when trying to locally fine-tune the gemma3 model. I searched online and couldn't find people having the same problem as me. I also asked AI, who told me to disable the compiler. This didn't work, I wasn't able to select a different backend_compiler. I am using a Jetson nano with Ubuntu 22.04, Cuda 12.6.0, python 3.10.12. details on the error and the notebook in question: https://github.com/WeslakAnton/mistrail/blob/main/pre-trained-transformer/gemma_unsloth.ipynb some important package versions: torch 2.0.0 triton 3.4.0 transformers 4.56.0 unsloth 2025.8.9 unsloth_zoo 2025.8.9 The error is very deep, suggest there might be a problem with the dependencies. Any help will be appreciated. Thanks in advance.	4	
Tuesday, 2 Septem ber 2025	Started work for the new approach - using HuggingFace. I spent most of the time researching the best methods available to do this in my architecture. I encountered some issues. I started setting up the venv with all of it's dependencies. Most of them should be installed correctly and with GPU compatibility. Useful guide to provide an overview of how the finetuning process should look like ideally: https://medium.com/data-science-in-your-pocket/a-practical-guide-to-fine-tuning-gpt3-gemma-3-270m-with-lora-cd03decf2ac1	2	
Wednesday, 3 Septem ber 2025	Verification: github -> gemma_huggingface.ipynb		
Thursday, 4 Septem ber 2025	Today I started working, got a head ache AND FINALLY DECIDED TO START USING JETSON-CONTAINERS. Best decision of my life. It really helps, and once you get it, it makes your life so much simpler. As it turns out Nvidia Jetson devices not only have a Aarch64 architecture, it's like some super special optimized version, meaning it's super exclusive and you'll have plenty of dependency problems if you work normal, like on a x86-64 device. Basically there are internal docker containers available for different popular python packages on github: https://github.com/dusty-nv/jetson-containers I was able to build on top of their containers to fit my needs. All of this obviously took a load of time since I'm a noob at docker and stuff, but now I slowly get it. I generated (with AI) a small script to check whether the bitsandbytes library is working properly. This test has been passed by the container. There was an additional problem where the datasets library I needed for formatting my training data wasn't available on the github. I was able to improvise with a docker command (sounds easy, but took me like 1.5h) Conclusion: ALL REQUIREMENTS (atleast most of them, im not entirely sure if that's all of them) HAVE BEEN INSTALLED AND THE PROGRAMM IS READY "TO BE CODED ON". --- Bitsandbytes GPU Verification Script --- Step 1: Checking for CUDA-enabled GPU... Success: CUDA is available. Found GPU: Orin Step 2: Loading a model with 8-bit quantization ('load_in_8bit=True')... The 'load_in_8bit' and 'load_in_8bit' arguments are deprecated and will be removed in the future versions. PleaseAndBytesConfig' object in 'quantization_config' argument instead. Success: Model loaded in 8-bit without errors. This indicates that bitsandbytes is correctly installed and communicating with the GPU. Step 3: Verifying model properties... Model is on device: cuda0 Model is correctly placed on the CUDA device. Model memory footprint: 165.54 MB Step 4: Performing a simple inference test (forward pass)... Success: Forward pass completed without errors. --- Verification Complete --- All checks passed! Your 'bitsandbytes' installation appears to be working correctly with your GPU.	6	
Friday, 5 Septem ber 2025	I was able to fine-tune gemma3, just like I intended. Amazing! Ran into some problems: - Installing new libraries in a container (temporary solution) - Github and huggingface authentication - Correct hyperparameters and config of the Trainer - System crashed while I was ssh-ed into it. I wasn't able to recover the system and had to stop working. All in all my task for today is completed.	7	
Saturday, 6 Septem ber 2025	Verification: github -> gemma_huggingface.ipynb		
Sunday, 7 Septem ber 2025			
Monday, 8 Septem ber 2025			
Tuesday, 9 Septem ber 2025	I fixed the github authentication problem, I found out how to evaluate the model and I tweaked the hyperparameters (only the learning rate scheduler) for better results, yet I think that it isn't meant to be. The model just doesn't wanna train beyond 0.4 training loss, despite of the different learning rates I tried. I also evaluated the Error on validation data, it was much bigger than expected, the model is overfitting.	5	
Wednesday, 10 Septem ber 2025	print(np.mean(diffs)) 1.240906 Verification: github -> gemma_huggingface_predict.ipynb		
Thursday, 11 Septem ber 2025	Played around some more with different parameters. I defined some eval data, and tried to implement it into the model, but it didn't work for some reason. This also leads to crashes. Exact reason unclear, (probably RAM usage) but why? SFTTrainer is weird, but I have to use it because of the QLoRA. The teacher also let me know to mention that I was "considering" reproducibility of my project. Thankfully he said there are no extra points for that, that would've been a problem for me if there were. Verification: github -> gemma_huggingface.ipynb	3	
Friday, 12 Septem ber 2025	I was finally able to add validation data properly. There now is also a computation of the accuracy of the model, which is the metric I used with the previous model (gemini 2.5). I did this using the compute_metrics function, which I initially generated with a new LLM I found (for coding) (MistralAI). There were a lot of errors, which I had to debug step by step. That sucked. Another problem was the amount of time it took. I just wanted to finish this up, so I could work remotely from school again tomorrow, since for this I need to sometimes restart the container because of memory overload. I should probably also mention that training this model takes 80min, compared to the previous one (without eval_data). Probably because of the inefficient compute_metrics function. [691/954 57:18 < 21:52. 0.20 it/s. Epoch 2.17/3]	6	
Saturday, 13 Septem ber 2025	Step Training Loss Validation Loss Accuracy Entropy Num Tokens Mean Token Accuracy 10 9.809200 6.865591 0.000000 1.418712 1763.000000 0.359184 20 4.320000 2.812313 0.000000 2.478687 3520.000000 0.520952 30 2.039000 1.491469 0.000000 1.725800 5285.000000 0.638003		

Hypothesis for reason of the error:

- There is a previous model that is being loaded, instead of the most recent one. (or not the most recent epoch)
- All the previous runs have been deleted, there is only one left. The specific checkpoint has been specified (the most recent one).
- There is an error with generating the data, like using different hyperparameters (temperature and stuff.) The pytorch .generate() and the huggingface provided generator pipeline both yielded similar results. And the pipeline uses hyperparameters from the fine-tune.
- There might be an issue with the calculation during the evaluation steps
- I thoroughly checked the compute_metrics function, and made sure the validation dataset is being used and not the train dataset. The function itself was fair and there is no error. So confirmed the multithreading.

	Step	Training Loss	Validation Loss	Accuracy	Entropy	Num Tokens	Mean Token Accuracy	
	10	9.809200	6.865591	0.000000	1.418712	1763.000000	0.359184	
	20	4.320000	2.812313	0.000000	2.478687	3520.000000	0.520952	
	30	2.039000	1.491469	0.000000	1.725600	5285.000000	0.638003	
	40	1.216000	1.120224	0.160000	1.125232	7039.000000	0.678528	
	50	0.974200	0.948166	0.130000	0.976329	8806.000000	0.694357	
	60	0.808300	0.742455	0.240000	0.889345	10566.000000	0.748972	
	Verification: github: -> gemma_huggingface.ipynb, https://huggingface.co/AntonBOOM/output							
Friday, 12 Septe mber 2025								
Saturday, 13 Septe mber 2025								
Sunday, 14 Septe mber 2025								
Monday, 15 Septe mber 2025								
Tuesday, 16 Septe mber 2025		After trying to evaluate my model on the data used before for gemini 2.5 I encountered a weird logical inconsistency where the accuracy calculated during the training process was much higher than if I run a test on the exact same data after the fine-tuning process now. This was very weird and I was suspicious of a generation error on my part, but I couldn't find one. I tried generating using the pytorch lower-end function, as well as the pipeline put forth on huggingface tutorials, both yielded similar results. After finetuning Gemma3 1b (a very similar, but larger LLM) I encountered a similar problem when evaluating. This time there was less of a difference between the two, which leads me to believe, that this wasn't an error on my part afterall. 0.63 Despite the expected accuracy of 1.00 I get something like this (0.63). This is a pretty big difference Verification github -> matura/pretrained/gemma-huggingface-predict, big-gemma-huggingface (This is an entry is an accumulation of multiple, from the past couple of days)						
Wednesday, 17 Septe mber 2025		Finished up the final touches for this workload. I finished up predictions with gemma3 1b. Everything with the gemma3 1b was actually done today, I just wrote the entry for yesterday today, that's why I mentioned it. The results are pretty good, there are reasons behind them, everything is good. Verification -> matura/pre-trained-transformers						
Thursday, 18 Septe mber 2025		I was just organizing today, and I tried to debug the before mentioned problem where the models, after being finetuned, showed many times worse performance than in training for whatever reason. I did the thing on the right as told by mr. Schneider. I can't think of any other hypothesis for the error. Al can't help me here. I don't know what to do I'm pretty stuck here. (I'll move on in a couple of days if the logical inaccuracy persists.) # The base model ID, as indicated by the error message base_model_id = "google/gemma-3-270m-it" # Your fine-tuned adapter adapter_id = "./output" # 1. Load the base model model = AutoModelForCausalLM.from_pretrained(base_model_id, dtype=torch.bfloat16, device_map="auto", trust_remote_code=True) # 2. Load the Peft model and merge it with the base model model = PeftModel.from_pretrained(model, adapter_id) model = model.merge_and_unload() # The 'model' variable now holds your fully fine-tuned model Verification: The table to the right						
Friday, 19 Septe mber 2025		Only a little bit of adjusting						
Saturday, 20 Septe mber 2025		Today I fixed a little error and reran the training of the gemma1b model. It wasn't a lot of work really, nothing to write home about.						
Sunday, 21 Septe mber 2025								
Monday, 22 Septe mber 2025								
Tuesday, 23 Septe mber 2025								
Wednesday, 24 Septe mber 2025								
Thursday, 25 Septe mber 2025								
Friday, 26 Septe mber 2025								
Saturday, 27 Septe mber 2025								
Sunday, 28 Septe mber 2025								
Monday, 29 Septe mber 2025								
Tuesday, 30 Septe mber 2025		Today I finally started working on the project again, I've been quite busy in the past few days. I revisited the issue of the pre-trained models giving different accuracies, then expected. I used one of the newest AI's for help with my problem and it suggested to add the line model.eval() line to ensure the model is in evaluation mode. I implemented this as well as some other few adjustments. But this didn't help much. Afterwards I had all the evaluation data needed for pre-trained fine-tuned models. I wrote it down in an excel file. (I also started this excel file a couple of days earlier) Verification: https://kb42puvydrk5w.ok.kimi.link and github						
Wednesday, 1 October		Today I sorted out the github issue and finally committed my thingy. This took long because I'm not very familiar with github. I also got a bill from google today for 83 CHF.						

All the previous runs have been deleted, there is only one left. The specific checkpoint has been specified (the most recent one).

- There is an error with generating the data, like using different hyperparameters (temperature and stuff.) The pytorch generate() and the huggingface provided generator pipeline both yielded similar results. And the pipeline uses hyperparameters from the fine-tune.
- There might be an issue with the calculation during the evaluation steps
I thoroughly checked the compute_metrics function, and made sure the validation dataset is being used and not the train dataset. The function itself is very basic and there is no error, as confirmed by multiple LLMs.
- The model is overfitted
No, predicting on train data yielded the same (slightly better) results

After trying out different checkpoints, all of a sudden the accuracy of the 270m model dropped from 40% to 15%

- This lead me to believe maybe it has something to do with the way the models are calculating everything, apart from the temperature and other hyperparameters, it might be because of performance optimization software calculation on GPU or CPU with work offload to CPU might give different results.
- Transformers is imported the same way, and is working on GPU exclusively both times.
- Ran the code on the exact same dataset used in the validation part during training, no differences.
- There is a slight difference with the calculation. In the compute metrics function the prediction is only counted if the model uses a turnseparator. This could explain why the seemingly normal gap in accuracies occurred.
After re-running the training with the correction, the accuracy dropped a little (from 100% to 97%) at the end.
- The pre-trained model obviously has a drop out and layer norms. The issue might be that the model is not in eval mode and is dropping out a lot of valuable data, this would explain the slightly worse performance. This was not the case, because after setting model.eval() this only increased the accuracy by around half a percent for the Gemma3 270M model, nothing really changed. + This should probably be on by default anyways.
- A perfectly reasonable explanation would be, the model's weights aren't loaded in to the model properly, like it happened with FNN2.py "see 4th of Oct Arbeitsprozess".
This cannot be the case because loading in the hyperparameters is completely different to loading in the weights. A model with loaded-in weights is already trained (hence the name - pre-trained model). Additionally the model is showing signs of having learned from the fine-tuning process for example in the way it responds with the trained int.0 instead of the traditional int.

r 2025

Thursday, 2 Octobe
ay, 2
October 2025 I started working with extracting the infos from the transformers notebooks. I will be using the 2 I got by using keras tuner to optimize hyperparameters. I quickly realized that using bootstraps is basically like cheating. And not fair, because I sometimes used them, sometimes I didn't: very sketchy. I decided to define: NO bootstraps.

5

I just shut down the billing account on google. It cost me way too much money.

I filled the excel table with the evaluation data from the both transformers as well as RNNN.

Total Parameters:	26595	RNNN: RNN2 SimpleRNN	transformer4: transformer
Architecture Parameters:	2951	493241	30259
Optimizer Parameters:	2734	291	76429
MAE in Range:	0.2977996	0.0381983	0.0440775
MRE in Range:	0.1297235	0.0151896	0.0180047
MAE out Range:	0.3737702	0.2301562	0.3221214
MRE out Range:	0.3856819	0.5415157	0.5001459
MAE long Expressions:	3.7057795	6.0690327	6.1303090
Benchmark scores:	0.4496330	5.2896360	2.8802627

Pretained fine-tuned Model: Gemma 2.5 Pro, Gemma 3.1B, Gemma 5.270M

Accuracy in Range:	95.17	97.87	54.22
Accuracy out Range:	95.31	79.74	54.22
Accuracy Long Range:	90	41	26
Parameter Info:	Not disclosed	1.00E+09	2.70E+08

Verification: github -> slightly updated notebooks transformer4, transformer5, RNNN

Friday, 3 October 2025 I included a attention based LSTM into the excel sheet. Afterwards I got to work on FNN2, which was fairly easy to make and didn't take long at all. The keras-tuner took 3h to determine the optimal number of neurons and layers is: 2 layers each with 9 neurons. The closer I etched to this process being finished the less RAM memory did the system have left, so the final couple took very long.

Trial 98 Complete [0h 0m 54s]

val_loss: 0.0318362616187171936

Best val_loss So Far: 0.0012506949715316296

Total elapsed time: 0:3h 0:6m 18s

Search: Running Trial #99

Value	Best Value So Far Hyperparameter
53	9 num_neurons
16	2 num_layers
False	False dropout/F
Epoch 1/100	
60/60	748s 11s/step - loss: 19.3370 - val_loss: 12.8697
Epoch 2/100	
60/60	22s 314ms/step - loss: 7.1973 - val_loss: 2.5593
Epoch 3/100	

Verification: github -> FNN/FNN2.pybn

Saturday, 4 October 2025 Today I was able to finally finish up work with the FNN2 notebook and thus the excel spreadsheet, and thus the evaluation part of my project!!!!!! This is verry good news. The results vary slightly from what I expected for some reason. From my hypothesis. Which since the zwischen produkt has been: smaller model with fewer tweakable parameters should perform better on this easy task. Because of the concept of overkill. This is from my lit. review.

To be more precise on what I did today: I ran the keras-tuner, now leaving more ram space for it by not using the gnome environment. I also made a bit of debugging: The FNN2 first yielded much worse results after the keras-tuner, after like 15 mins I finally realised I had forgotten to train the model, after loading it with the required hyperparameters.

FNNs:	FNN2
Total Parameters:	6,211
Architecture Parameters:	6,211
Optimizer Parameters:	0
MAE in Range:	0.0389406
MRE in Range:	0.0155504
MAE out Range:	2.2301364
MRE out Range:	0.2509270
MAE long Expressions:	6.2929916
Benchmark score:	6.6343632

Verification: github -> FNN2

Sunday, 5 October 2025 Today I just reorganized the github again, especially gave proper names to the files in pre-trained-transformers. I also tried using google jules to generate a readme file. This didn't work, because jules started hilucinating very drastically.

Monday, 6 October 2025 Today I downloaded LaTeX, aryan and simon did. When I was writing the lit. review overleaf had notified me that the document was soon going to be too large, and that I'll need to subscribe.

I am now using Visual Studio Code in combination with git as an editor. After setting everything up I was able to start the document and stuff. I decided to make two documents btw. One about the methodology and a second about the findings and theoretical background.

I started with the introduction to the methodology.

I struggled a lot with the copilot being ANNOYING. I was able to just completely deactivate him. I'm not dealing with that.



Verification: github synchronized /documentation

Tuesday, 7 October 2025 I began with the "real" writing part today. I wrote the sections: train-test, FNN. I also set up the bibliography and a whole bunch of other stuff. I was able to work with VSC sync'd with git between Desktop and laptop. Oh and I fixed the problem with the FNN1 notebook.

For more details: just check the commits.

Verification: github -> methodology.tex

Wednesday, 8 October 2025 I started Today with writing the part about the benchmark. I'm still not sure wether to include theory in methodology or findings. I then did all of RNNs including simple RNN and attentional RNN. I rethought Dropout a little. I implemented the stuff I wrote about RNNs earlier, but I think it's AI generated, so I'll have to re write it. I'll keep the formula probably.

Ohhh right and I also trained a new model which I called RNN2. It is a copy from the FNN2 notebook. It just implements keras-tuner for SimpleRNN, to give it a more fair chance in the excel table. It's still the worst model by far:-)

Contents	
0.1 Brief Summary of the Project	1
0.2 Introduction to this document	1
1 Feed-forward Neural Networks (FNN)	4
1.1 Train and Test Data	4
1.2 Training a Neural Network Using TensorFlow	4
1.3 FNN1 and FNN2 Notebooks	6
1.4 The Benchmark	6
1.5 Drop-out	7
2 Recurrent Neural Network (RNN)	7
2.1 Basic Visualization of an RNN	7
2.2 RNN1 and RNN2	8
3 Attention and Transformers	9
3.1 Attentional RNNs	9
References	10

Above you can see how the Table of contents looks right now for verification.

Verification: github -> RNN2.ipynb and methodology.tex

Thursday, 9 October 2025 Today I did a lot. I reworked documentation directory. I added requirements.txt I wrote A LOT OF LATEX. I was able to finish the methodology document. Well, not finish... I finished writing everything in there. There's still stuff for me to correct and stuff.

I also added AI_assistance directory, which contains a latex file where I list the major requests I had with AI. I didn't really have any. Only one. It's the only reason I included this directory.

Afterwards I shared my progress with aryan and he did likewise. He suggested, I should add a gittignore file. He helped me through this, but I mainly used the google jules AI for this. It was helpful, but I could've done it myself much quicker, I just wanted to use it for fun.

So yeah, I cleaned up my repo and am currently trying to use jules again to write a readme file for the repo, after this I'll go to bed.

Edit: this took longer than expected.

Verification: check github commits

Friday, 10 October 2025 I started writing findings. There I wrote a whoole bunch. 2 pages to be exact with lots of research on the side.

Yesterday, I was able to employ a great technique for explaining how a RNN architecture works: I explain it a little, I show the equations, I explain the equations. I think that should be enough.

3.5

r 2025

$$\begin{aligned} f_t &= \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) & (1) \\ i_t &= \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) & (2) \\ \tilde{C}_t &= \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) & (3) \\ o_t &= \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) & (4) \\ C_t &= f_t \odot C_{t-1} + i_t \odot \tilde{C}_t & (5) \\ h_t &= o_t \odot \tanh(C_t) & (6) \end{aligned}$$

Source for the equations: GeeksforGeeks, 2025a

Some beautiful equations describing the LSTM architecture above.

I also decided to put all the theory into the findings document

Verification: [github/documentation/findings](#)

Saturday, 11 October 2025

The writing went smooth

Contents

0.1 Abstract	1
0.2 Introduction	1
0.3 Feed-forward Neural Networks (FNNs)	1
1 RNN	3
1.1 Numerical Visualization of a RNN	3
1.2 Relevant Takeaway	3
2 Other Types of RNNs	3
2.1 Long Short-Term Memory (LSTM)	4
2.2 Gated Recurrent Unit (GRU)	4
2.3 Bidirectional LSTM with Attention	5
3 Transformers	6
3.1 Multi-Head Self-Attention	6
3.2 The Encoder Layer	7
3.3 Point-wise Feed-Forward Network	7
3.4 Positional Encoding	7
References	8

Verification: [github/documentation/findings](#)

Sunday, 12 October 2025

Check it out!! I finished 2 pages on the transformer architecture. What's left now are the pre-trained finetuned LLMs. This should be shorter, but the next section, is the most important one: the real findings. I already have a lot to say about my data and I have explanations for the way the benchmarks shaped themselves the way they did.

6.5

Monday, 13 October 2025

Verification: [check github commits](#)

Tuesday, 14 October 2025

Wednesday, 15 October 2025

Thursday, 16 October 2025

Friday, 17 October 2025

Saturday, 18 October 2025

Sunday, 19 October 2025

I really wanted to finish the findings today, but I couldn't and I think it's for the best. I wrote a section on the pre-trained transformers, which felt kinda dry, because it was just a big block of text. Later in the day I also wrote the section on final evaluation and conclusion. The problem is I didn't really have a conclusion LOL I wrote down some notes and was able to add a latex table of the findings, but not more.

6

Verification: [github](#)

Monday, 20 October 2025

I worked on the conclusion again and pretty much finished it up. I also added FNN3 infos to methodology and findings and Aliusage.

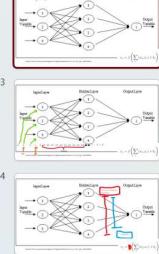
3

I set up the date for an interview with a buddy. He fits the Zielgruppe perfectly.

Tuesday, 21 October 2025

I started working on the presentation for marton today. I don't think it will be much most of the talk will be very improvised and it will be more of a discussion, than a presentation. I put together some graphs to help me explain the different architectures today. Oh, and I also wrote a thing or two about further works.

3

Verification: [Look my image](#)

A sneak peek into the presentation.

Wednesday, 22 October 2025

I finished up the documentation today. Well for the most part. I added some details, I ran an AI model through my pdfs and let it correct my text for grammar errors. It did a surprisingly well job, I still had to mow through the pages after it and clean everything up, but I was impressed. I also moved and rewrote the dropout section.

Thursday, 23 October 2025

I had a meeting today, in which we talked with Herr Schneider, he helped me out a lot with my questions.

I later wrote to him about him correcting my documentation, I'm not sure if he's going to because altogether it's 27 pages. That might scare him off. But in reality there's relatively little real content in there.

Structure for guided discussion with marton:

- Introduction
- Consent for recording
- Question about past experience with neural networks
- Open documentation lead by me.
- Closing remarks and stuff
- Camera off
- Show rest of the project if requested
- Give the feedback questionnaire to Marton

Friday, 24 October 2025

I finished work for the discussion with marton. We will meet tomorrow at 11 o'clock. I prepared a short powerpoint. And a questionnaire, in which I ask him all the relevant stuff, as well as also see if he understood what I taught him.

I worked till late today, because I still had lots to do and mr.Schneider gave me some great ideas on what to do with marton, to squeeze as much information out of him as possible.

BTW I decided to only explain the FNN and transformer models. And skip over the RNNs entirely. All in all the presentation itself isn't long. But I want it to be this way, so that Marton gets the ability to constantly barrage me with questions.

Saturday, 25 October 2025

I finished work for the discussion with marton. We will meet tomorrow at 11 o'clock. I prepared a short powerpoint. And a questionnaire, in which I ask him all the relevant

6

stuff, as well as also see if he understood what I taught him.

I worked till late today, because I still had lots to do and mr.Schneider gave me some great ideas on what to do with marton, to squeeze as much information out of him as possible.

BTW I decided to only explain the FNN and transformer models. And skip over the RNNs entirely. All in all the presentation itself isn't long. But I want it to be this way, so that Marton gets the ability to constantly barrage me with questions.

Guided Discussion

When you submit this form, it will not automatically collect your details like name and email address unless you provide it yourself.

Kontrollfragen zum Thema

1. Das Transformer model besitzt "positional awareness". Daten die ins Model hineingefüllt werden, laufen also zuerst durch keinen

The discussion will be in german.

5.5

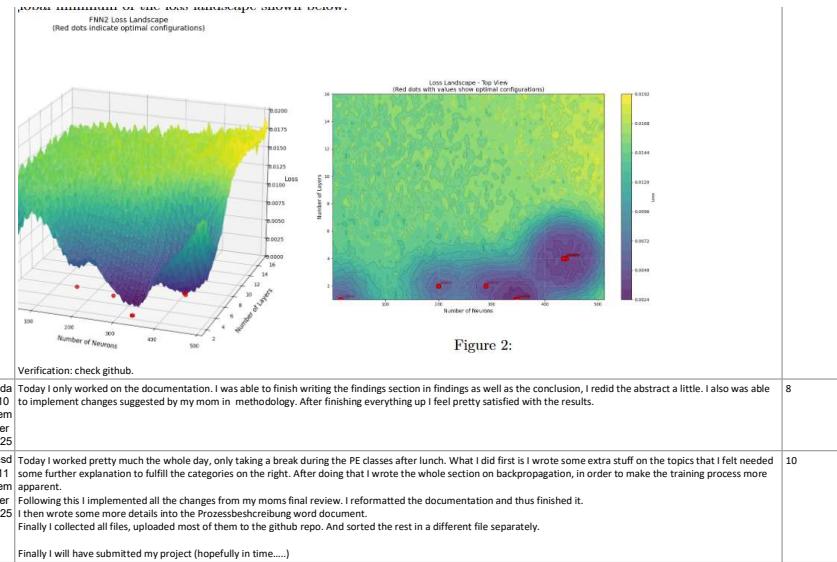
Saturday, 25 October 2025	<p>Verification: PPP done. Microsoft forms done.</p> <p>The discussion went great, no awkward questions came up, which I couldn't answer. I did find that I stuttered a little too much, this is also something I'll have to work on for the presentation at the end.</p> <p>All in all we had a great talk, the recording went well, except that at some point during the recording I decided to move my mic away from my mouth which made me sound very quiet.</p> <p>The feedback was collected, everything went smoothly as you'd expect.</p> <p>Verification: Video and Excel table with questionnaire results.</p>	1.5
Sunday, 26 October 2025	<p>First I reviewed all the responses collected in the questionnaire and left my own comment on them.</p> <p>A conclusion was reached.</p> <p>Later I also wrote them into the methodology and findings.</p>	3.5
Monda y, 27 October 2025	<p>5.1 Guided Discussion</p> <p>After conducting an open presentation for a colleague, his feedback was collected and his knowledge of the presented topics graded.</p> <p>He asked me what the reason for conducting this discussion was to test if regression models (like the ones in this project) are easier to teach and understand for people interested in neural networks, who already possess some basic knowledge.</p> <p>The Results from the questionnaire yield that the discussion partner didn't fully understand either topic but was able to answer basic questions. The topic of FNNs was easier to grasp than transformers.</p> <p>According to the subject, the direct examples from this project which were delivered in accordance to theory, were the best help. The reason for this is that they can be easily understood by anyone who has studied the basics of regression models.</p> <p>The subject of Regression models is about sequence to sequence (seq2seq) models also helped. The reasons listed by the subject were the similarities between the regression in the output layer and linear regression as taught in school, as well as the usage of mostly numbers throughout the whole model architecture, from input to output. The subject also stated that the latter helped with the understanding of tokens or vectors and how they are processed. Prior to this discussion that was not apparent to the subject.</p> <p>The main criticism found in the feedback includes the lack of information received was the lack of explanation for how the optimization process works. This is understandable, because the focus for this project was placed elsewhere.</p> <p>In conclusion, though only evaluated on a very small dataset, it can be said that it is beneficial to explain the basic workings of FNNs and transformers on the basis of models for regression, because of the wider use of numbers, as well as better connections to topics like linear regression, discussed as a part of the standard curriculum in school.</p>	3.5
	<p><small>*The FNN and transformer architectures of regression models were taught.</small></p> <p>The new subsection in findings as it is, with spelling mistakes.</p> <p>Verification: Github and Excel file with questionnaire results.</p>	
Tuesday, 28 October 2025	<p>I was able to rewrite the positional Encoder Section with more context. Then I later also implemented some changes suggested by my mom for the methodology. There really isn't much else to say.</p> <p>Verification: Github -> documentation</p>	3
Wednesday, 29 October 2025		
Thursday, 30 October 2025		
Friday, 31 October 2025		
Saturday, 1 November 2025		
Sunday, 2 November 2025		
Monday, 3 November 2025	<p>I finished implementing the good suggested improvements from my mom's review.</p> <p>Verification: github.cation: github</p>	3
Tuesday, 4 November 2025		
Wednesday, 5 November 2025	<p>All improvement suggestions from mr. Schneider were implemented. After going through his list I also let gemini grammatically correct both documents. After some other minor adjustments the documentation files were submitted to both of my parents for a final review and feedback round.</p> <p>I also wrote down some questions for Schneider on the right</p>	6
Thursday, 6 November 2025	<p>I had a meeting with mr. Schneider where I got the chance to ask the last couple of questions. There I got even more useful comments, and help. After the meeting I got to work on writing a document about the process, as suggested by mr. Schneider. I used a mermaidchart.com website to generate some pretty charts and diagrams to make my point.</p> <p>I also investigated how I could prove the significance of my results. I reached the conclusion, that I should probably just calculate the p-value for the benchmark values of all the regression models, by rerunning them around 5 times. The null-hypothesis would then be that I expect results to be the same across all models.</p> <pre>3.754284362408516 0.2685873463700985 0.16808440109152306 0.1470259778828506 Benchmark: 4.337985707335067</pre> <p>The benchmark is visibly, even worse with the new replaced values in the tokenizer (0.25, 0.75)</p> <p>Verification: github: multiple commits.</p>	3
Friday, 7 November 2025	<p>In school I was able to write a script NPC-final, where the user can quickly access a trained model, and the model responds to user requests to calculate different expressions. (I also fiddled around with the findings.)</p> <p>I later worked on the issue with the p-values, how to prove the significance? As I don't have lots of experience with stats, I first tried the shapiro-wilk test for normality. That wasn't what I wanted, then I tried the Kolmogorov-smirnov test with a uniform distribution, which turned out to be once again not what I wanted.</p> <p>Back at home, while working on this I noticed a huge error in the calculation of deviation for long expressions. Just a typo. But this meant that I will have to correct it for all regression models and rerun them. The keras-tuner doesn't have to be rerun, since it uses validation data, there wasn't a bug there.</p> <p>I reran all models except the transformer.</p> <p>My mom helped me with the statistics, she showed me that I can do a one-sample t-test with 1 to show a p-value of a list of benchmarks, with which the benchmarks perform better than 1.</p> <p>Verifications: lots of commits on github</p>	7
Saturday, 8 November 2025	<p>Yesterday I found a minor inconvenience in the code. A GIANT ASS BUG. It basically meant that all the calculations for long expressions were wrong. Which means that lucky old me had to recalculate everything.</p> <p>This time around I was much more careful, I was scanning everything for bugs. I also formulated a more firm early stopping and found a couple of bugs. I actually had to do everything twice, because there was a mistake with my correction. While the program was calculating new stuff, I was working on the documentation and the excel table. I wasn't able to finish everything for the documentation, but I got pretty far, like halfway.</p> <p>Oh, and I probably should mention that I finally implemented a p-value, hopefully correctly now. I used the two-sided one-sample t-test value, as suggested (correctly I think) by my mom (statistician).</p> <p>I might have forgotten to mention something, I worked the whole day, and its late now.</p> <pre>diffLongRange = [] for i in range(200, 300): diffLongRange.append(np.array(np.abs(long_y_test[i]) - np.array(predsLongRange[i)))) ✓ NEEDdiffLongRange = [] for i in range(len(long_y_test)): NEEDdiffLongRange.append(np.array(np.abs(long_y_test[i]) - np.array(predsLongRange[i)))) ✓ print("MAE longer Expressions: ", np.mean(NEEDdiffLongRange)) diffOutRange = []</pre> <p>Verification: like 10 commits on github, because I was working on like 3 devices simultaneously.</p>	9
Sunday, 9 November 2025	<p>I started the day by noticing a minor inconvenience with the calculation of the benchmark test pvalue. There needs to be a possibility for the benchmark to be negative. I solved this by applying a logarithmic transformation to the benchmarks. Then I wrote a new script to calculate the benchmarks correctly. I then wrote the whole section about the findings table in methodology, this took the most amount of time. I did sooo much more stuff that I won't even have the time to mention. But basically I also worked on the findings. I was re-writing the findings section. I also generated the script to draw the loss landscape shown below. It's drawn for the tuning summary of FNN. Generated by claude-opus-4-1-20250805-thinking-16k.</p>	10

Questions for schneider:
Figure4 something wrong with the citation???

Does he think I should combine the two papers into one? Or rename the findings document perhaps?

Emph dashes?

* 1.1: Wie viele Datensätze von was haben Sie?
* 3.2: Loshchilov ... müsste in Klammern. -> Das haben sie jetzt nur bei diesem zitat bemerkten. Wollen sie vielleicht dass ich gleich alle zitate in klammern mache?



Verification: check github.

Monday, 10 November 2025
Today I only worked on the documentation. I was able to finish writing the findings section in findings as well as the conclusion, I redid the abstract a little. I also was able to implement changes suggested by my mom in methodology. After finishing everything up I feel pretty satisfied with the results.

8

Tuesday, 11 November 2025
Today I worked pretty much the whole day, only taking a break during the PE classes after lunch. What I did first is I wrote some extra stuff on the topics that I felt needed some further explanation to fulfill the categories on the right. After doing that I wrote the whole section on backpropagation, in order to make the training process more apparent.
Following this I implemented all the changes from my moms final review. I reformatted the documentation and thus finished it.

10

I then wrote some more details into the Prozessbeschreibung word document.
Finally I collected all files, uploaded most of them to the github repo. And sorted the rest in a different file separately.
Finally I will have submitted my project (hopefully in time....)