Can Zero-shot LLMs save additional work in machine learning prioritised screening for systematic reviews?

Max Callaghan

October 30, 2024

Can LLMs help us to save more work?

- Prioritised screening has saved us a lot of work

Can LLMs help us to save more work?

- Prioritised screening has saved us a lot of work
- But we still spend substantial amounts of time screening documents for systematic reviews

Can LLMs help us to save more work?

- Prioritised screening has saved us a lot of work
- But we still spend substantial amounts of time screening documents for systematic reviews
- LLMs have been proposed as a solution, with

Can LLMs help us to save more work?

- Prioritised screening has saved us a lot of work
- But we still spend substantial amounts of time screening documents for systematic reviews
- LLMs have been proposed as a solution, with
  - Wang et al. (2024) showing impressive results for the zero-shot setting

	Model	Setting	P	R	B-AC	F3	Suc	WSS
CLEF-2017	Bio BERT	Unc	0.06	0.95	0.61	0.30	0.74	0.26
		Cal(0.95)	0.06	0.92	0.64	0.31	0.50*	0.34*
		Cal(1)	0.06	0.97	0.60	0.29	0.82	0.23
	7b-ins	Unc	0.08	0.87	0.72	0.35	0.26	0.56
		Cal(0.95)	0.06*	0.92*	0.69*	0.32	0.52	0.44
		Cal(1)	0.05*	**0.99***	0.60*	0.28	0.96	0.20
	13b-ins	Unc	0.19	0.41	0.66	0.31	0.04	0.91
		Cal(0.95)	0.06*	0.93	0.59*	0.28	0.50*	0.25*
		Cal(1)	0.05*	0.98	0.53*	0.26	0.88*	0.08*
	Ensemb	Unc	**0.31**	0.13	0.56	0.13	0.00	**0.98**
		Cal(0.95)	0.08	0.93*	**0.72**	**0.35***	0.52*	0.50*
		Cal(1)	0.06	0.97*	0.63	0.30	**0.90***	0.29*
CLEF-2018	Bio BERT	Unc	0.06	0.97	0.59	0.29	0.87	0.19
		Cal(0.95)	0.07	0.91*	0.63	0.30	0.57*	0.33*
		Cal(1)	0.06	0.97	0.59	0.29	0.87	0.21
	7b-ins	Unc	0.09	0.88	0.75	0.37	0.27	0.59
		Cal(0.95)	0.08*	0.94*	0.71*	0.35*	0.50	0.46
		Cal(1)	0.06*	**0.99***	0.62*	0.30	**1.00**	0.24
	13b-ins	Unc	0.26	0.36	0.66	0.30	0.00	0.94
		Cal(0.95)	0.06	0.94*	0.59*	0.29	0.47*	0.22*
		Cal(1)	0.05	0.97	0.53*	0.27	0.80*	0.08*
	Ensemb	Unc	**0.35**	0.12	0.54	0.12	0.00	**0.95**
		Cal(0.95)	0.09*	0.94*	**0.75**	**0.38***	0.50*	0.54*
		Cal(1)	0.06	0.99*	0.64	0.32*	0.93*	0.28*
CLEF-2019-dta	Bio BERT	Unc	0.07	0.99	0.58	0.30	0.88	0.18
		Cal(0.95)	0.08	0.89	0.59	0.26	0.50	0.27
		Cal(1)	0.08	0.91	0.59	0.27	0.62	0.25
	7b-ins	Unc	0.09	0.92	0.71	**0.35**	0.62	0.49
		Cal(0.95)	0.10*	0.91*	0.71*	0.34	0.50	0.50
		Cal(1)	0.08*	0.97*	0.66	0.32	0.75	0.34
	13b-ins	Unc	0.19	0.49	0.69	0.32	0.00	0.87
		Cal(0.95)	0.08	0.95	0.56	0.29	0.50*	0.16*
		Cal(1)	0.07	0.99	0.51	0.28	0.88*	0.03*
	Ensemb	Unc	**0.31**	0.21	0.59	0.19	0.00	**0.96**
		Cal(0.95)	0.10	0.91	**0.73**	0.34*	0.50*	0.52*

Can LLMs help us to save more work?

- Prioritised screening has saved us a lot of work
- But we still spend substantial amounts of time screening documents for systematic reviews
- LLMs have been proposed as a solution, with
  - Wang et al. (2024) showing impressive results for the zero-shot setting
  - and Xia et al. (2024) erroneously claiming that LLMs fix all of the problems faced until now by ml+screening

Table 2
Performance of GPT 4o mini in screening titles and abstracts against a human reviewer's good truth using different k-value

Data set	k-value	Precision	F1-score	Kappa	PABAK	AUC-ROC	Recall
Hall	0.0	0.575	0.672	0.667	0.662	0.900	0.808
Hall	0.5	0.649	0.670	0.666	0.663	0.844	0.692
Hall	1.0	0.653	0.644	0.640	0.636	0.815	0.635
PTSD	0.0	0.129	0.226	0.216	0.210	0.924	0.895
PTSD	0.5	0.161	0.270	0.260	0.255	0.892	0.816
PTSD	1.0	0.176	0.286	0.277	0.271	0.868	0.763
Virus	0.0	0.339	0.485	0.446	0.417	0.882	0.851
Virus	0.5	0.375	0.508	0.473	0.446	0.861	0.789
Virus	1.0	0.436	0.557	0.527	0.502	0.860	0.772

Can LLMs help us to save more work?

- Prioritised screening has saved us a lot of work
- But we still spend substantial amounts of time screening documents for systematic reviews
- LLMs have been proposed as a solution, with
  - Wang et al. (2024) showing impressive results for the zero-shot setting
  - and Xia et al. (2024) erroneously claiming that LLMs fix all of the problems faced until now by ml+screening
- There are a lot of bad papers, which will continue to proliferate, each tweaking some parameters to achieve a high performance that cannot be generalised

Table 2
Performance of GPT 4o mini in screening titles and abstracts against a human reviewer's good truth using different k-value

Data set	k-value	Precision	F1-score	Kappa	PABAK	AUC-ROC	Recall
Hall	0.0	0.575	0.672	0.667	0.662	0.900	0.808
Hall	0.5	0.649	0.670	0.666	0.663	0.844	0.692
Hall	1.0	0.653	0.644	0.640	0.636	0.815	0.635
PTSD	0.0	0.129	0.226	0.216	0.210	0.924	0.895
PTSD	0.5	0.161	0.270	0.260	0.255	0.892	0.816
PTSD	1.0	0.176	0.286	0.277	0.271	0.868	0.763
Virus	0.0	0.339	0.485	0.446	0.417	0.882	0.851
Virus	0.5	0.375	0.508	0.473	0.446	0.861	0.789
Virus	1.0	0.436	0.557	0.527	0.502	0.860	0.772

Realistic and useful evaluations of LLMs for screening

Despite the hype, LLMs might still be helpful, but we need to know

Realistic and useful evaluations of LLMs for screening

Despite the hype, LLMs might still be helpful, but we need to know

- How do LLMs compare against commonly used (and much cheaper!) baselines?

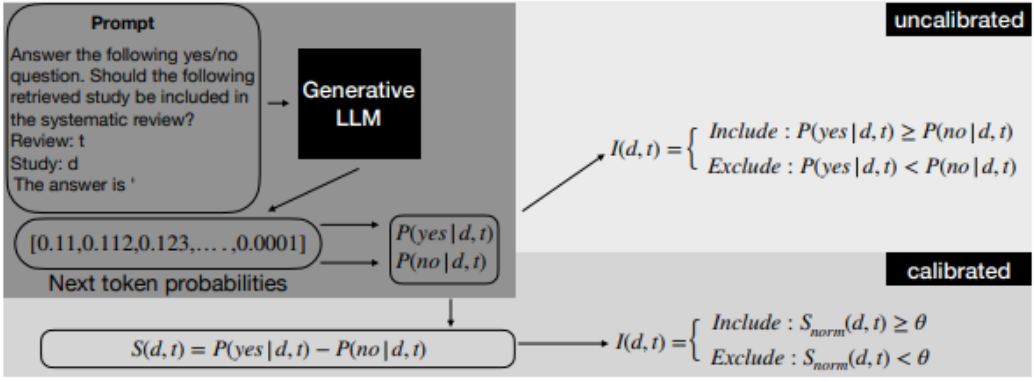
Realistic and useful evaluations of LLMs for screening

Despite the hype, LLMs might still be helpful, but we need to know

- How do LLMs compare against commonly used (and much cheaper!) baselines?
- How could they safely be used in the real world (Hint: with stopping criteria and prioritised screening)

Zero-shot Generative Large Language Models for Systematic Review Screening Automation

Wang et al. (2024) propose a method to extract probability-like scores from LLMs for inclusion/exclusion decisions in a systematic review



Implementation is relatively straightforward

```python
def binary_probs(tokenizer, model, prompt, no_words=['no'], yes_words=['yes'], return_all=False):
    device = 'cuda' if torch.cuda.is_available() else 'cpu'
    encoded_text = tokenizer(prompt, return_tensors="pt").to(device)
    #1. step to get the logits of the next token
    with torch.inference_mode():
        outputs = model(**encoded_text)

    next_token_logits = outputs.logits[0, -1, :]

    # 2. step to convert the logits to probabilities
    next_token_probs = torch.softmax(next_token_logits, -1)

    topk_next_tokens= torch.topk(next_token_probs, 50)
    tokens = [tokenizer.decode(x).strip().lower() for x in topk_next_tokens.indices]
    p = topk_next_tokens.values

    df = pd.DataFrame.from_dict({'t': tokens,'p': p.cpu()})
    y = df[df['t'].isin(yes_words)]['p'].sum()
    n = df[df['t'].isin(no_words)]['p'].sum()

    if return_all:
        return df.groupby('t').sum().reset_index().sort_values('p', ascending=False).reset_index(drop=True)
    return y-n, y+n
```

Implementation is relatively straightforward

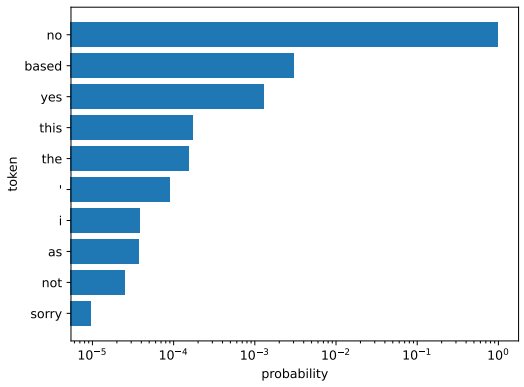
```
prompt = Template('''<s>[INST] <<SYS>>
You are a systematic review helper tasked with finding out whether a study is relevant to
     the review $t

Answer 'yes' if the study is relevant, or 'no' if not
<</SYS>>

Study: $s

Should the study be included? Answer yes or no. [/INST] ''')

prompt.substitute({'t': review, 's': study_title}),
```

Implementation is relatively straightforward

Result: <s>[INST] «SYS» You are a systematic review helper tasked with finding out whether a study is relevant to the review Drug Class Review: Pharmacologic Treatments for Attention Deficit Hyperactivity Disorder
Answer 'yes' if the study is relevant, or 'no' if not «/SYS»
Study: Diuretics and beta-blockers do not have adverse effects at 1 year on plasma lipid and lipoprotein profiles in men with hypertension. Department of Veterans Affairs Cooperative Study Group on Antihypertensive Agents.
Concern based on the reported short-term adverse effects of antihypertensive agents on plasma lipid and lipoprotein profiles (PLPPs) has complicated the therapy for hypertension.
Should the study be included? Answer yes or no. [/INST] No, this study should not be included in the drug class review for Pharmacologic Treatments for Attention Deficit Hyperactivity Disorder. The study is focused on the effects of diuretics and beta-blockers on plasma lipid and lipoprotein profiles in men with hypertension, and does not directly relate to the treatment of Attention Deficit Hyperactivity Disorder (ADHD). The review is focused on pharmacologic treatments for ADHD, and the study does not address any of the included drug classes or outcomes of interest for the review.

Implementation is relatively straightforward

Result: <s>[INST] «SYS» You are a systematic review helper tasked with finding out whether a study is relevant to the review Drug Class Review: Pharmacologic Treatments for Attention Deficit Hyperactivity Disorder
Answer 'yes' if the study is relevant, or 'no' if not «/SYS»
Study: Diuretics and beta-blockers do not have adverse effects at 1 year on plasma lipid and lipoprotein profiles in men with hypertension. Department of Veterans Affairs Cooperative Study Group on Antihypertensive Agents.
Concern based on the reported short-term adverse effects of antihypertensive agents on plasma lipid and lipoprotein profiles (PLPPs) has complicated the therapy for hypertension.
Should the study be included? Answer yes or no. [/INST] No, this study should not be included in the drug class review for Pharmacologic Treatments for Attention Deficit Hyperactivity Disorder. The study is focused on the effects of diuretics and beta-blockers on plasma lipid and lipoprotein profiles in men with hypertension, and does not directly relate to the treatment of Attention Deficit Hyperactivity Disorder (ADHD). The review is focused on pharmacologic treatments for ADHD, and the study does not address any of the included drug classes or outcomes of interest for the review.



Comparing priorisation approaches

- We evaluate using the Synergy dataset De Bruin et al. (2023) of 26 systematic reviews (Tim, Diana, Sergio, Lena, and James, will make a much bigger one!)

Comparing priorisation approaches

- We evaluate using the Synergy dataset De Bruin et al. (2023) of 26 systematic reviews (Tim, Diana, Sergio, Lena, and James, will make a much bigger one!)
- For each review, we generate 100 different initial random samples of 10%, and do normal prioritised screening in batches of 10% with an SVM 100 times

Comparing priorisation approaches

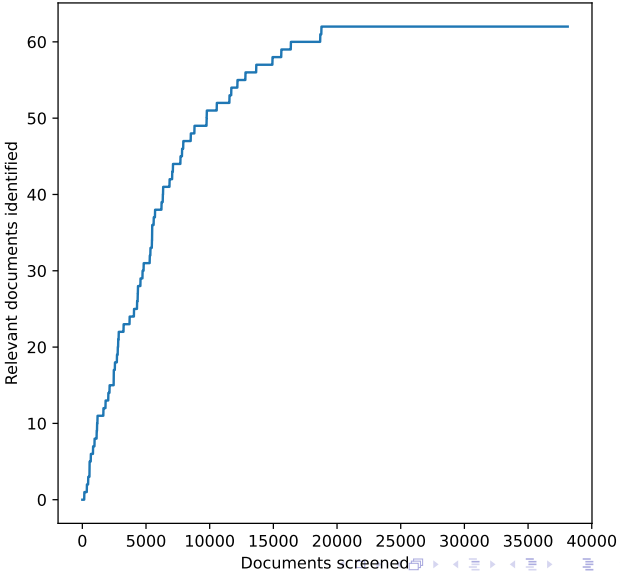
- We evaluate using the Synergy dataset De Bruin et al. (2023) of 26 systematic reviews (Tim, Diana, Sergio, Lena, and James, will make a much bigger one!)
- For each review, we generate 100 different initial random samples of 10%, and do normal prioritised screening in batches of 10% with an SVM 100 times
- We also make a prediction for each document with 4 Llama models of different sizes and ages

Comparing priorisation approaches

- We evaluate using the Synergy dataset De Bruin et al. (2023) of 26 systematic reviews (Tim, Diana, Sergio, Lena, and James, will make a much bigger one!)
- For each review, we generate 100 different initial random samples of 10%, and do normal prioritised screening in batches of 10% with an SVM 100 times
- We also make a prediction for each document with 4 Llama models of different sizes and ages
- We turn these predictions into a prioritised list of documents to be screened

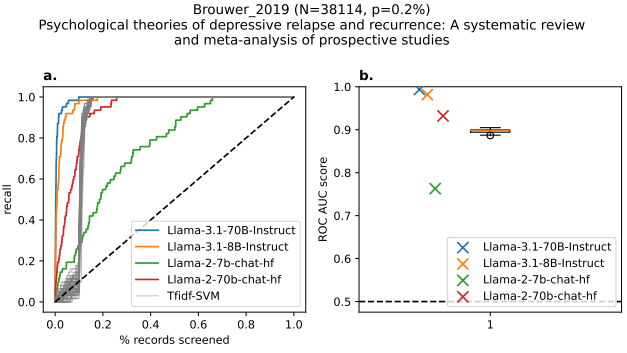
Result #1: It works!

- The method produces a ranking that identifies all relevant documents before all documents have been screened



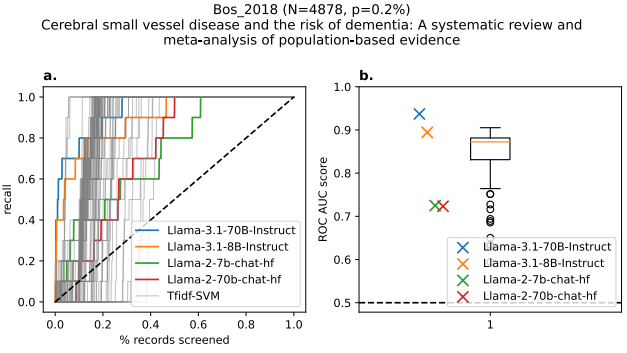
Result #1: It works!

- The method produces a ranking that identifies all relevant documents before all documents have been screened
- The first model we tried was much worse than our SVM, but newer and bigger models were much more promising!



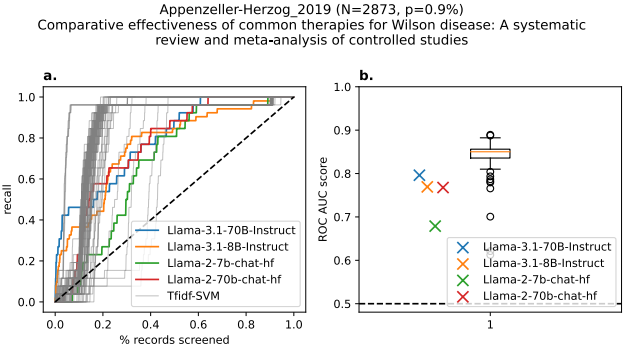
Result #1: It works!

- The method produces a ranking that identifies all relevant documents before all documents have been screened
- The first model we tried was much worse than our SVM, but newer and bigger models were much more promising!
- But performance varied across datasets



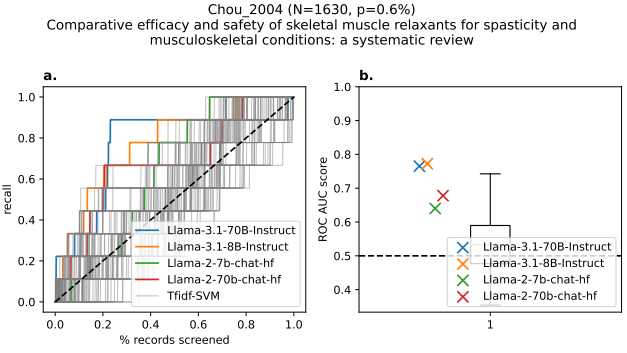
Result #1: It works!

- The method produces a ranking that identifies all relevant documents before all documents have been screened
- The first model we tried was much worse than our SVM, but newer and bigger models were much more promising!
- But performance varied across datasets
- ROC AUC scores offer one way to compare rankings



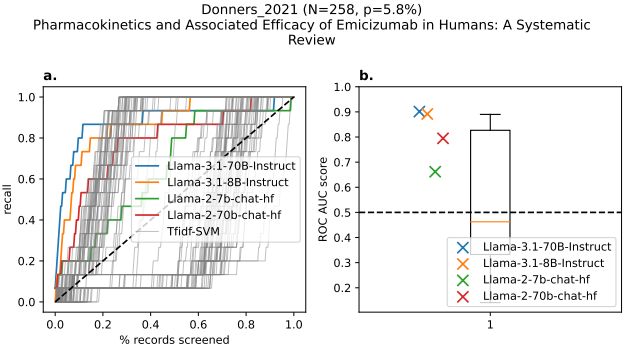
Result #1: It works!

- The method produces a ranking that identifies all relevant documents before all documents have been screened
- The first model we tried was much worse than our SVM, but newer and bigger models were much more promising!
- But performance varied across datasets
- ROC AUC scores offer one way to compare rankings



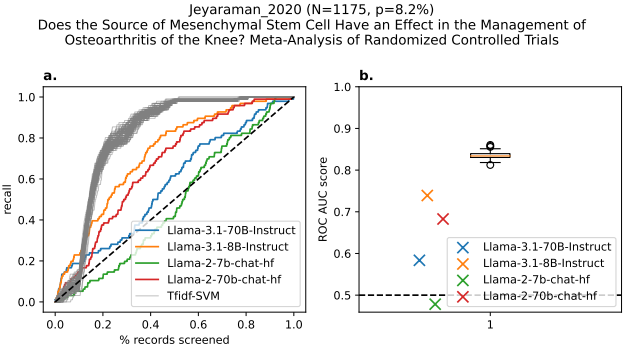
Result #1: It works!

- The method produces a ranking that identifies all relevant documents before all documents have been screened
- The first model we tried was much worse than our SVM, but newer and bigger models were much more promising!
- But performance varied across datasets
- ROC AUC scores offer one way to compare rankings



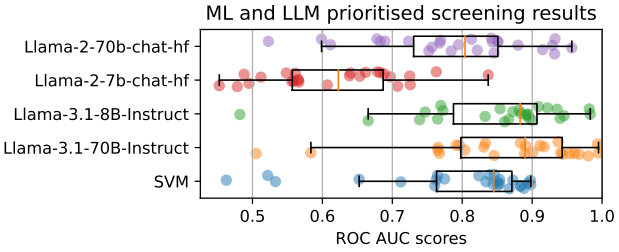
Result #1: It works!

- The method produces a ranking that identifies all relevant documents before all documents have been screened
- The first model we tried was much worse than our SVM, but newer and bigger models were much more promising!
- But performance varied across datasets
- ROC AUC scores offer one way to compare rankings



Result #1: It works!

- The method produces a ranking that identifies all relevant documents before all documents have been screened
- The first model we tried was much worse than our SVM, but newer and bigger models were much more promising!
- But performance varied across datasets
- ROC AUC scores offer one way to compare rankings



Stopping criteria

- Next, we calculate scores for our stopping criteria with a range of different recall targets

Stopping criteria

- Next, we calculate scores for our stopping criteria with a range of different recall targets
- Stopping when $p < \alpha$ gives a recall $>$ target in the α^{th} percentile, in 97/100 settings

r target	model confidence	SVM	Llama-2-7b	Llama-2-70b	Llama-3.1-8B	Llama-3.1-70B
0.80	0.50	0.938	**0.758**	0.893	0.911	0.911
	0.80	0.924	**0.797**	0.866	0.845	0.889
	0.90	0.900	0.818	0.872	0.836	0.876
	0.95	0.867	0.806	0.888	0.878	0.878
	0.99	0.811	0.838	0.865	0.876	0.886
0.90	0.50	0.963	0.905	0.936	0.957	0.954
	0.80	0.957	0.900	0.958	0.961	0.933
	0.90	0.947	0.937	0.959	0.953	0.939
	0.95	0.933	0.943	0.944	0.947	0.939
	0.99	0.928	0.941	0.947	0.926	0.958
0.95	0.50	0.979	**0.947**	1.000	0.981	0.987
	0.80	0.974	0.973	0.976	0.975	0.979
	0.90	0.973	0.968	0.981	0.978	0.980
	0.95	0.964	0.973	0.982	0.976	0.968
	0.99	0.973	0.975	0.986	0.983	0.975
0.99	0.50	1.000	1.000	1.000	1.000	1.000
	0.80	0.994	1.000	1.000	1.000	0.991
	0.90	0.993	0.996	1.000	0.999	0.996
	0.95	0.993	0.997	1.000	0.999	0.996
	0.99	0.996	0.992	0.999	0.997	0.996

Stopping criteria

- Next, we calculate scores for our stopping criteria with a range of different recall targets
- Stopping when $p < \alpha$ gives a recall $>$ target in the α^{th} percentile, in 97/100 settings
- In most settings, total work savings were greater for SVMs

r target	model confidence	SVM	Llama-2-7b	Llama-2-70b	Llama-3.1-8B	Llama-3.1-70B
0.80	0.50	126,898	100,248	121,777	127,401	**132,808**
	0.80	113,708	64,568	97,147	111,411	**114,748**
	0.90	105,053	57,678	90,327	101,871	**106,668**
	0.95	98,503	48,608	83,338	94,921	**101,288**
	0.99	**87,803**	37,838	71,397	79,471	85,058
0.90	0.50	114,988	65,778	99,217	112,011	**115,498**
	0.80	93,923	42,768	79,807	83,661	**97,438**
	0.90	**84,658**	34,158	65,007	76,701	82,268
	0.95	**76,483**	28,748	59,447	69,251	73,108
	0.99	**62,350**	23,118	49,790	55,091	59,088
0.95	0.50	98,258	44,408	82,907	94,731	**101,318**
	0.80	**74,098**	27,238	58,587	64,281	68,678
	0.90	**63,245**	21,738	44,470	53,091	57,388
	0.95	**54,860**	18,781	37,370	45,954	48,671
	0.99	**41,676**	13,545	27,171	34,855	35,592
0.99	0.50	**63,573**	22,638	42,897	54,811	55,378
	0.80	**31,323**	11,158	20,597	29,191	30,398
	0.90	**21,160**	5,825	13,480	13,641	18,478
	0.95	**16,229**	4,065	8,837	9,435	11,938
	0.99	**7,980**	1,985	4,355	4,934	6,625

Table: absolute work savings across datasets (N records=169,288)

Some things still to explore

- Prompting strategies (inclusion criteria)

Some things still to explore

- Prompting strategies (inclusion criteria)
- Combining LLMs with traditional approaches

Some things still to explore

- Prompting strategies (inclusion criteria)
- Combining LLMs with traditional approaches
- Updating prompts based on user feedback

Some things still to explore

- Prompting strategies (inclusion criteria)
- Combining LLMs with traditional approaches
- Updating prompts based on user feedback
- Better baselines (finetuning BERT)

Some things still to explore

- Prompting strategies (inclusion criteria)
- Combining LLMs with traditional approaches
- Updating prompts based on user feedback
- Better baselines (finetuning BERT)

Note that although the LLM implementation is simplistic, so is the comparator

Conclusions

- LLMs are neither a quick fix or a silver bullet

Conclusions

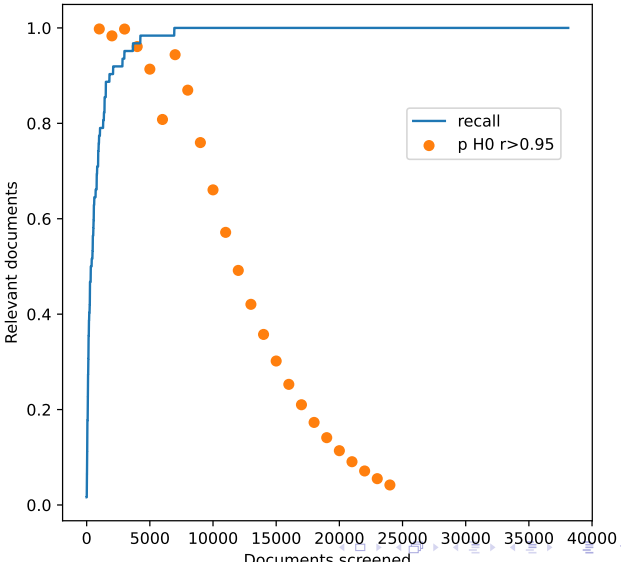
- LLMs are neither a quick fix or a silver bullet
- They do not have magic properties that fix problems of traditional ML, they are simply likely to make them worse

Conclusions

- LLMs are neither a quick fix or a silver bullet
- They do not have magic properties that fix problems of traditional ML, they are simply likely to make them worse
- We need to be on our guard!

Conclusions

- LLMs are neither a quick fix or a silver bullet
- They do not have magic properties that fix problems of traditional ML, they are simply likely to make them worse
- We need to be on our guard!
- Do not forget stopping criteria!



Conclusions

- LLMs are neither a quick fix or a silver bullet
- They do not have magic properties that fix problems of traditional ML, they are simply likely to make them worse
- We need to be on our guard!
- Do not forget stopping criteria!
- But they have some advantages, such as the fact that we only need to calculate scores once!

Callaghan, M. and Müller-Hansen, F. (2020). Statistical Stopping Criteria for Automated Screening in Systematic Reviews. *Systematic Reviews*.

De Bruin, J., Ma, Y., Ferdinands, G., Teijema, J., and Van de Schoot, R. (2023). SYNERGY - Open machine learning dataset on study selection in systematic reviews.

Wang, S., Scells, H., Zhuang, S., Potthast, M., Koopman, B., and Zuccon, G. (2024). Zero-shot Generative Large Language Models for Systematic Review Screening Automation.

Xia, Z., Ye, J., Hu, B., Qiang, Q., and Debnath, R. (2024). LLMscreen: A Python Package for Systematic Review Screening of Scientific Texts Using Prompt Engineering.