Investigating Comparison Reasoning Skills in a Large Language Model

Chrisanna Kate Cornish
ccor@itu.dk

Abstract

This study investigates the comparison reasoning capabilities of a large language model (LLM), specifically OLMo_7B_0724_Instruct, by evaluating its responses to counterfactual comparison questions, and comparing saliency scores across tokens that the model should, and shouldn't find important, following the work of Ray Choudhury et al. (2022). Our results indicate that the model frequently fails to answer both counterfactual questions correctly, suggesting a reliance on heuristics, rather than the expected reasoning skill. The saliency analysis shows that the model often does not consider the same tokens important as a human would. These findings indicate that this method of investigating reasoning abilities has some limitations, but the model likely relies on shortcuts rather than human-type reasoning.

1 Introduction

Large Language Models are rapidly becoming ubiquitous in modern life. As they are becoming more integrated into our everyday lives, understanding their reasoning process is increasingly important, especially as decision-making abilities are turned over to them.

This study follows the work of Ray Choudhury et al. (2022) who looked at how BERT-style NLP models comprehend language, including defining expected reasoning steps for specific tasks, and analysing models behaviour with regards to this.

We used two tests designed to identify whether the model was reasoning in line with this definition. Firstly, a counterfactual explanation was used which found a difference in ability with original and perturbed questions, suggesting the model was not reasoning correctly, but rather relying on something else to answer the questions. Secondly, a saliency-based method, integrated gradients was used to establish whether the model would find the same tokens important as a human reader, and this showed no evidence that this was occurring. We also considered whether there was any alignment between the two methods, and here we found that there was generally very little alignment. An example is shown in Figure 1, which does show the expected reasoning although this was not a general trend.

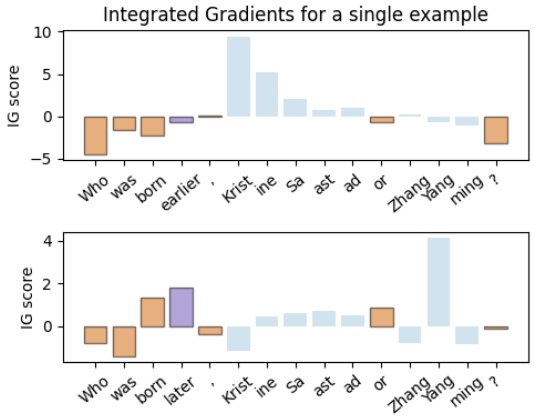

Figure 1: Example of saliency scores attributed to an example question and its perturbed counterpart. The model output for the first was '*Kristine Saastad*' and the second '*Zhang Yangming*', which aligned exactly with the expected answers. For the first question, the positive partition score was: -0.68, whilst the negative was -2.06, and for the second, 1.82 against -0.08. The interpretation of this example is that the model is showing the expected reasoning.

Code relating to this project can be found at its Github repository.

2 Related Work

Ray Choudhury et al. (2022) looked at how BERT-style NLP models comprehend language, including defining expected reasoning steps for specific tasks, and analysing models behaviour with regard to this. Nikankin et al. (2024) looked at mathematical reasoning of LLMs and concluded they were using a 'bag of heuristics', rather than specific rea-

soning or relying on memorisation. Huang et al. (2023) took a different approach and considered whether LLMs could explain themselves and concluded that the evaluations were different but as faithful as traditional methods, however Chen et al. (2023) found that LLMs generated misleading explanations that lead to wrong mental models. This follows the work of Jacovi and Goldberg (2020). Their work developed guidelines on the difference between a plausible explanation, and a faithful one; the first being one that is believable to us as humans, for example a convincing chain-of-thought explanation, and the second being one that accurately represents the model's reasoning process, which is harder to be certain of. Atanasova et al. (2020) found gradient-based explanations such as saliency performed well across various tasks and model architectures, although they did not look at generative models. Zhou et al. (2024) however found that SHAP-based explanations were more faithful than gradient-based ones in a smaller scale study. This is an evolving field and no single agreed on 'best' method exists. Not only that, there is little agreement on what is meant by reasoning, or faithfulness, making comparison across works difficult. This paper will rely on the definition of reasoning provided by Ray Choudhury et al. (2022) where they consider that a system has human-level competence with respect to its task iff:

a. *It is able to correctly perform the task X (identify the target information in QA, correctly classify texts, generate an appropriate translation etc.);*

b. *It does so by relying predominantly on information that a competent human speaker would also find relevant;*

c. *It does so consistently under distribution shifts that do not pose challenges to competent human speakers.*

Using this, we will test parts b. and c., to investigate an LLM as to whether it is able to demonstrate human-level reasoning in a QA task.

3 Methodology

3.1 Model

This work will use the OLMo 7B 0724 Instruct model[1] (Groeneveld et al., 2024) released in July 2024. This is built on their 7B model as a base and then fine-tuned for question-answering problems.

[1]https://huggingface.co/allenai/OLMo-7B-0724-Instruct-hf

3.2 Data

Following the procedure from Ray Choudhury et al. (2022), 23K examples were extracted from HotpotQA(Yang et al., 2018) and 2wikimultihopQA(Ho et al., 2020) datasets. Of these, Named Entity Recognition (NER) was used to extract questions with 2 and only 2 entities mentioned in the question text as well as one of 6 comparison terms (see Table 1), leaving 9.6K examples that could be perturbed to create counterfactual questions, by exchanging the comparison term for its partner, and swapping the expected answer between the two identified entities. This allows the question to still make sense and be solvable for a competent human reader.

Comparison term	→	Counterpart
first	→	last
earlier	→	later
later	→	earlier
more recently	→	earlier
younger	→	older
older	→	younger

Table 1: Comparison terms and their counterparts, as identified in Ray Choudhury et al. (2022)

3.3 Counterfactual Explanations

We create paired questions with original and perturbed comparisons, Figure 2 shows an example of an original and its paired question. This is designed to test part c. of the definition in Section 1, as this shouldn't pose a challenge to a competent human speaker. If the model is relying on a shortcut or some incorrect reasoning, it will not be able to answer both questions correctly.

3.4 Integrated Gradients-based Explanations

Integrated Gradients (IG) (Sundararajan et al., 2017) is a way to assign an importance value to each input token, via estimating the integral of the outputs gradients with respect to the inputs as they vary along a line to a neutral baseline. This should test part b. of the definition from Section 1, that it should rely on information that humans would find relevant. Using Captum's (Miglani et al., 2023) implementation, to extract a score for each token in the input, we followed Ray Choudhury et al.

Context: Ainhoa Artolazábal Royo(born 6 March 1972) is... Allen Holden(18 April 1911 – 12 December 1980) was...

Original
Question: Who was born earlier, Allen Holden or Ainhoa Artolazábal?
Answer: Allen Holden
Perturbed
Question: Who was born later, Allen Holden or Ainhoa Artolazábal?
Answer: Ainhoa Artolazábal

Figure 2: Perturbed question example. Question entities are marked in blue, comparison terms in orange.

(2022)'s method to identify two partitions in the **Question** text according to the rules in Table 2.

Partition	Rule
Positive	Comparison token
Negative	Everything that wasn't part of the comparison term, or the entities being compared

Table 2: Partition rules

The positive partition identifies what comparison needs to be made and is the only difference between the two questions. If the model is reasoning as expected, we expect this partition to have a higher score than the negative partition. The entities themselves do not form part of either partition. An example is shown in Figure 3.

Who was born earlier , Kristine Saastad or Zhang Yangming ?

Figure 3: Partitions example. The positive partition is marked in blue, the negative partition in orange.

3.5 Baselines

The baseline for the counterfactual explanations will be the score on the original questions. The baseline for the IG explanations will be the scores from a random selection of tokens.

If the model is reasoning as expected, we would expect that the model will perform as well on the perturbed questions as for the original ones. We would also expect the model to have a higher saliency score for the positive partition compared to the negative. If the methods are faithful, we would also expect to see some alignment, so the model will perform well on both questions and have positive partition > negative partition.

4 Results

4.1 Counterfactual Explanations

The model was considered correct when its answer contained the expected answer, and not the second entity, for both the original and the perturbed question. It was considered to have failed if the answer contained the same entity (and only that entity) for both questions. Other possible answer combinations were to have given the opposite answer to both questions or to have given both or neither answers to one or both questions. Table 3 shows that we see similar proportions of correct and failed question pairs.

Performance	Count	Percentage
Correct	3127	32.6%
Fail	3027	31.3%
Reversed	217	2.2%
Doubled/Neither*	3276	33.9%

Table 3: Response accuracy for paired questions. *Doubled/Neither includes 2633 pairs where the response was empty for one or both questions.

Examination of the failed cases shows that in 78%, the model gave a response appropriate to the original question for both, rather than the perturbed one.

We also considered the accuracy[2], the Exact Match[3] and the F1 score[4] for the model. Table 4 shows that for all three metrics, the model performed better on the original questions over the perturbed ones.

The distribution of comparison terms was uneven, as shown in Table 5. 'First'/'last' only appear in original/perturbed sentences respectively, and are by far the most common terms whilst other terms appear in both sentences. We also see differences in questions with specific comparison words,

[2]1 if expected answer in the response, else 0
[3]1 if the response matches expected answer exactly, else 0
[4]calculated by proportions of shared words in response and expected answer

Question	Accuracy	Exact Match	F1
Both	32.6%	4.2%	0.42
Original	68.4%	27.7%	0.52
Perturbed	46.7%	10.2%	0.31

Table 4: Average Accuracy, Exact Match and F1 scores for Both questions, Original only, and Perturbed only.

for example, if the original comparison term is 'earlier', the model tended to perform better, with 45.0% of examples being correct, compared to the general average of 32.6%, whereas examples with 'first' as the original comparison term were correct only 24.5% of the time. A breakdown for all terms can be found in Appendix A.

Term	Original %	Perturbed %
First	46.6%	0%
Last	0%	46.6%
Earlier	21.3%	14.1%
Later	10.7%	21.3%
More recently	3.4%	0%
Older	10.1%	7.9%
Younger	7.9%	10.1%

Table 5: Proportion of comparison terms in the original and perturbed sentences.

We also see difference in the distribution of F1 scores for the original and perturbed questions. Figure 4 shows that the F1 scores have peaks at 0 and 1, representing no match at all and perfect matches, and otherwise the scores cluster around the means for each. Whilst the very high scores correspond to the exact match where the original sentences are around 3 times higher than the perturbed, the opposite end of total failure is also very different, with 2672 for the original sentences against 4593 for the perturbed.

If the model were reasoning as expected, its performance on both the original and the perturbed questions should be similar. As such, these results do not provide evidence that the model is following the expected reasoning strategy, but is relying on some other heuristics.

4.2 Integrated Gradients

Both the original and the perturbed questions were passed to the IG algorithm, and scores were returned for 1271 of them, representing approx 13% of the samples. The IG scores relating to the question part of the text were isolated, and divided into the positive and negative partitions. To create the baseline, a random sample of tokens were taken, matching the length of the positive and negative partitions for each example. As shown in Figure 5, the average difference for each positive-negative partition pair clustered around 0, with little difference in the median and IQR, however many values lay outside the IQR, especially for partitions focussed on the original questions. Interestingly, the original partitions had extreme values greater than 0, whilst the random partitions had extreme values sitting far below 0.

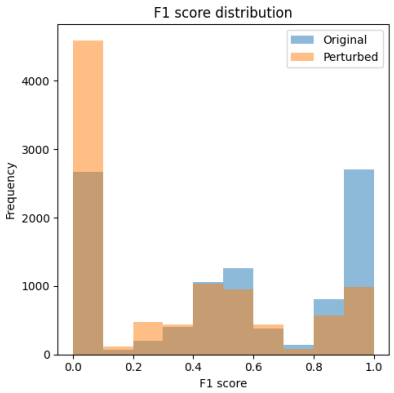

Figure 4: Distribution of F1 scores for original and perturbed questions

We also considered individually if a sample's positive partition was significantly higher than its negative partition, using a one-tailed t_test with the null hypothesis that the positive partition did not have a higher average score than the negative partition, and $p = 0.05$. The results are shown in Table 6, along with Benjamini-Hochberg corrections to control the False Discovery Rate due to the number of tests carried out. This shows a small increase of 5.2% in the original questions compared to their random baseline, and no difference for the perturbed questions.

Section	Significant%	Corrected%
Original	45.9%	38.9%
Perturbed	41.1%	39.9%
Orig. Random	41.4%	33.7%
Pert. Random	41.3%	39.9%

Table 6: Proportion of samples where the positive partition is significantly higher than the negative partition

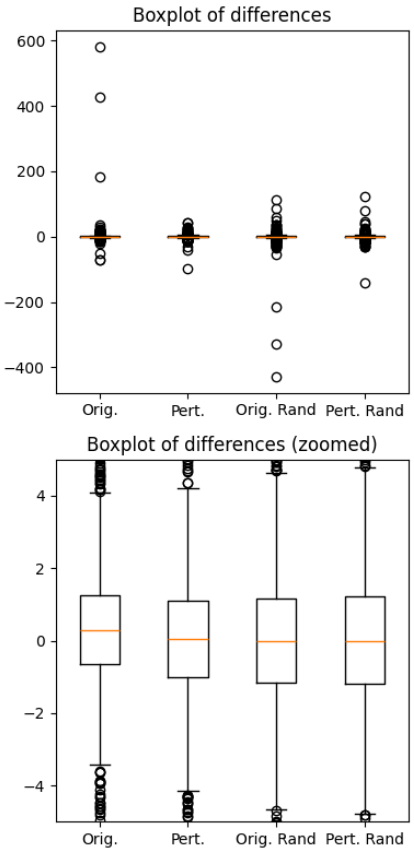

Figure 5: Positive-Negative partition differences.

This does not provide evidence that the model follows the expected reasoning pattern, and gives importance to tokens that humans would find relevant.

4.3 Alignment

As a final check, we also considered the alignment between the two methods. If the model is reasoning, then we would expect to see a correct answer for both questions, as well as a significantly higher positive partition compared to its negative partition. These results are shown in Table 7, and we see generally there is little alignment between the two methods.

We also considered the cases where the higher positive partition was significant. Amongst the original questions with a higher positive partition, 85.5% answered the original question correctly, compared to 82.7% correct without a higher positive partition. Also for the perturbed questions with a higher positive partition, 59.1% answered the perturbed question correctly, compared to 56.4% without. This does not show support that the model is generally reasoning as expected.

Question	Alignment
Both	6.8%
Original	33.3%
Perturbed	23.6%

Table 7: Alignment between methods. Both: higher positive partitions for both questions and both answered correctly. Original: higher positive partition and answered correctly in the original question, regardless of the behaviour for the perturbed, and Perturbed: higher positive partition and answered correctly in the perturbed question, regardless of the behaviour for the original.

5 Discussion

This investigation has not supported the idea that the OLMo model is reasoning in a human-like way in a comparison task. This could be because the model is simply not following the expected reasoning, or because the methods are insufficient to detect the reasoning skills. It could be that the model is capable of higher level reasoning skills but the comparison questions were too similar to some other task that it had learnt a shortcut to solve during training. This would be a failure of the task, rather than the model itself. Further testing with different types of tasks may help to discover this. Another approach would be to fine-tune the model with the expected reasoning, to teach it to use its reasoning capacities (if they exist) rather than relying on shortcuts.

Analysis of the training data could give some insight into specific failings, for example why it tends to perform better on the original questions rather than the perturbed ones. Certain comparison terms were more frequent in the original questions than the perturbed ones, and if this is also the case in English in general, and reflected in its training data, then it would follow that it would be more capable with the more frequent terms. The two explainability methods used in this work were adapted from Ray Choudhury et al. (2022)'s work, which found that BERT-style models did appear to use the expected reasoning for comparison tasks, however this work fails to replicate that result for the LLM. This is in line with Jacovi and Goldberg (2020)'s paper, which considered that a method that was faithful to one model or task was not necessarily going to be as faithful when used to evaluate another.

The average accuracy for the questions that we were able to obtain saliency scores for was higher

than the general case. This is due to the model returning empty responses, which were considered incorrect in the general analysis, but not able to be included in the saliency analysis.

5.1 Limitations

A limitation of this work is that using the model 'out-of-the-box' made it difficult to analyse the results easily. Rather than simply answering the question given, it tends to give a longer style answer. In some cases it returns an empty response which causes failures with the Captum IG implementation. Fine-tuning could improve this but could also allow the model to learn specific shortcuts for this dataset rather than allowing it to show its reasoning skills. Constrained decoding (Beurer-Kellner et al., 2024) was also considered, but not implemented in the end, rather the model was allowed to generate a restricted number of tokens, which were then processed for analysis.

Of 23K examples, only 9.6K were identified via the NER method to be able to automatically create the perturbed questions. It is possible that if some other method could be developed in order to have a more varied dataset, it would yield more comprehensive results. This method of pattern matching can also fail, either by not identifying the whole of an entity, or failing to identify one at all. Another problem was with the IG implementation, which failed when the model returned an empty response. This meant the number of steps between the input token and the baseline had to be reduced in order to allow some results to be obtained. This makes the method less reliable. Ye et al. (2021) also looked at similar methods and concluded that token-based attribution was less useful than pairwise interactions methods for analysing reasoning in a NLP model.

We could also consider that the model's better performance on the original questions could be an indicator that the model has seen this data, or something more similar to the original questions during its training. Neither dataset is listed as having been included directly, so further investigation into the similarity to training data may provide some answers into this.

By the time of writing, the OLMo2 model suite has been released, which includes larger models and claim better performance across a range of metrics. Applying these same methods to the newer models would give an interesting point of comparison, given Ray Choudhury et al. (2022) found there to be variation in different versions of BERT models.

5.2 Broader Impacts

Models such as LLMs are considered 'black-box' models; they are opaque in that the path from input to output isn't straightforward, making it difficult to analyse exactly how they come to their specific outputs. According to Nvidia [5], LLMs are used in a wide variety of applications, including in life-science research, finance, retail and law. Being able to examine the model is important if models are to be use in automatic decision making, in order to comply with GDPR requirements[6], as well as in order to examine them for bias in their decision making. This study shows it is still a challenge to reliably explain LLM outputs.

6 Conclusion

In conclusion, this paper used two methods to examine the reasoning skills of a LLM. We found little to support the idea that the model was reasoning in a manner similar to humans. Counterfactual questions were used that relied on the model reasoning correctly with regards to the comparison term, in order to answer both questions. We found that this occurred in around 32% of question pairs. In a similar number of pairs, the model gave the same answer to both, usually the answer to the original question, implying that it was using some other method to come to its decision. We also considered a gradient-based saliency explanation, with the idea that we could identify the tokens that the model 'should' pay attention to, and those that it shouldn't, in order to determine whether it was right for the right reasons. We also found that there was no evidence to support that the model was doing so in most cases. Further research with different explainability methods is needed to find a method that is reliable and faithful to this type of model.

Acknowledgments

I would like to thank Anna Rogers for her guidance and patience in supervising this project.

[5] https://blogs.nvidia.com/blog/what-are-large-language-models-used-for/

[6] 71): https://eur-lex.europa.eu/eli/reg/2016/679/oj

References

Pepa Atanasova, Jakob Grue Simonsen, Christina Lioma, and Isabelle Augenstein. 2020. A Diagnostic Study of Explainability Techniques for Text Classification. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3256–3274, Online. Association for Computational Linguistics.

Luca Beurer-Kellner, Marc Fischer, and Martin Vechev. 2024. Guiding LLMs The Right Way: Fast, Non-Invasive Constrained Generation. *arXiv preprint*. ArXiv:2403.06988 [cs].

Yanda Chen, Ruiqi Zhong, Narutatsu Ri, Chen Zhao, He He, Jacob Steinhardt, Zhou Yu, and Kathleen McKeown. 2023. Do Models Explain Themselves? Counterfactual Simulatability of Natural Language Explanations. *arXiv preprint*.

Dirk Groeneveld, Iz Beltagy, Pete Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Harsh Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, Shane Arora, David Atkinson, Russell Authur, Khyathi Raghavi Chandu, Arman Cohan, Jennifer Dumas, Yanai Elazar, Yuling Gu, Jack Hessel, Tushar Khot, William Merrill, Jacob Morrison, Niklas Muennighoff, Aakanksha Naik, Crystal Nam, Matthew E. Peters, Valentina Pyatkin, Abhilasha Ravichander, Dustin Schwenk, Saurabh Shah, Will Smith, Emma Strubell, Nishant Subramani, Mitchell Wortsman, Pradeep Dasigi, Nathan Lambert, Kyle Richardson, Luke Zettlemoyer, Jesse Dodge, Kyle Lo, Luca Soldaini, Noah A. Smith, and Hannaneh Hajishirzi. 2024. OLMo: Accelerating the Science of Language Models. *arXiv preprint*.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020. Constructing A Multi-hop QA Dataset for Comprehensive Evaluation of Reasoning Steps. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6609–6625, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Shiyuan Huang, Siddarth Mamidanna, Shreedhar Jangam, Yilun Zhou, and Leilani H. Gilpin. 2023. Can Large Language Models Explain Themselves? A Study of LLM-Generated Self-Explanations. *arXiv preprint*.

Alon Jacovi and Yoav Goldberg. 2020. Towards Faithfully Interpretable NLP Systems: How Should We Define and Evaluate Faithfulness? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4198–4205, Online. Association for Computational Linguistics.

Vivek Miglani, Aobo Yang, Aram Markosyan, Diego Garcia-Olano, and Narine Kokhlikyan. 2023. Using Captum to Explain Generative Language Models. In *Proceedings of the 3rd Workshop for Natural Language Processing Open Source Software (NLP-OSS 2023)*, pages 165–173, Singapore. Association for Computational Linguistics.

Yaniv Nikankin, Anja Reusch, Aaron Mueller, and Yonatan Belinkov. 2024. Arithmetic Without Algorithms: Language Models Solve Math With a Bag of Heuristics. *arXiv preprint*.

Sagnik Ray Choudhury, Anna Rogers, and Isabelle Augenstein. 2022. Machine Reading, Fast and Slow: When Do Models "Understand" Language? In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 78–93, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic Attribution for Deep Networks. *arXiv preprint*.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering. *arXiv preprint*. ArXiv:1809.09600 [cs].

Xi Ye, Rohan Nair, and Greg Durrett. 2021. Connecting Attributions and QA Model Behavior on Realistic Counterfactuals. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5496–5512, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Wei Zhou, Heike Adel, Hendrik Schuff, and Ngoc Thang Vu. 2024. Explaining Pre-Trained Language Models with Attribution Scores: An Analysis in Low-Resource Settings. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 6867–6875, Torino, Italia. ELRA and ICCL.

A Comparison terms individual breakdown

These tables show the break down of results per comparison term, as well as whether the term was part of the original or the perturbed question. A count of 0 means the term never appeared in this context.

First	*as original*	*as perturbed*
F1 orig.	0.58	nan
F1 pert.	0.23	nan
Exact Match orig.	0.35	nan
Exact Match pert.	0.08	nan
Accuracy orig.	0.74	nan
Accuracy pert.	0.34	nan
counts		
Correct	1104 (24.5%)	0
Fail	1976 (43.8%)	0
Reversed	101 (2.2%)	0
Double/Neither	1332 (29.5%)	0

Last	*as original*	*as perturbed*
F1 orig.	nan	0.58
F1 pert.	nan	0.23
Exact Match orig.	nan	0.35
Exact Match pert.	nan	0.08
Accuracy orig.	nan	0.74
Accuracy pert.	nan	0.34
counts		
Correct	0	1104 (24.5%)
Fail	0	1976 (43.8%)
Reversed	0	101 (2.2%)
Double/Neither	0	1332 (29.5%)

Earlier	*as original*	*as perturbed*
F1 orig.	0.59	0.40
F1 pert.	0.39	0.46
Exact Match orig.	0.35	0.16
Exact Match pert.	0.15	0.17
Accuracy orig.	0.74	0.57
Accuracy pert.	0.57	0.65
counts		
Correct	928 (45.0%)	560 (41.1%)
Fail	479 (23.2%)	308 (22.6%)
Reversed	52 (2.5%)	46 (3.4%)
Double/Neither	604 (29.3%)	450 (33.0%)

Later	*as original*	*as perturbed*
F1 orig.	0.37	0.59
F1 pert.	0.45	0.39
Exact Match orig.	0.16	0.35
Exact Match pert.	0.17	0.15
Accuracy orig.	0.53	0.74
Accuracy pert.	0.64	0.57
counts		
Correct	383 (36.9%)	928 (45.0%)
Fail	231 (22.3%)	479 (23.2%)
Reversed	32 (3.1%)	52 (2.5%)
Double/Neither	392 (37.8%)	604 (29.3%)

Younger	*as original*	*as perturbed*
F1 orig.	0.31	0.41
F1 pert.	0.34	0.30
Exact Match orig.	0.06	0.15
Exact Match pert.	0.08	0.05
Accuracy orig.	0.50	0.62
Accuracy pert.	0.58	0.48
counts		
Correct	251 (33.0%)	314 (32.2%)
Fail	97 (12.7%)	167 (17.1%)
Reversed	5 (0.7%)	13 (1.3%)
Double/Neither	408 (53.6%)	482 (49.4%)

Older	*as original*	*as perturbed*
F1 orig.	0.41	0.31
F1 pert.	0.30	0.34
Exact Match orig.	0.15	0.06
Exact Match pert.	0.05	0.08
Accuracy orig.	0.62	0.50
Accuracy pert.	0.48	0.58
counts		
Correct	314 (32.2%)	251 (33.0%)
Fail	167 (17.1%)	97 (12.7%)
Reversed	13 (1.3%)	5 (0.7%)
Double/Neither	482 (49.4%)	408 (53.6%)

More Recently	*as original*	*as perturbed*
F1 orig.	0.51	nan
F1 pert.	0.49	nan
Exact Match orig.	0.16	nan
Exact Match pert.	0.16	nan
Accuracy orig.	0.72	nan
Accuracy pert.	0.71	nan
counts		
Correct	177 (54.3%)	0
Fail	77 (23.6%)	0
Reversed	14 (17.8%)	0
Double/Neither	58 (4.3%)	0