

Using ChatGPT for Story Sifting in Narrative Generation

Gonzalo Méndez and Pablo Gervás

Facultad de Informática
Universidad Complutense de Madrid
Madrid, Spain
{gmendez,pgervas}@ucm.es

Abstract

The task of selecting a subset of story-worthy events from out of an observed collection of facts—known as *story sifting*—is a useful human ability that has yet to be emulated successfully by computational processes. The emergence of Large Language Models (LLMs) has made it necessary to rethink the way of carrying out many tasks that were previously performed using other tools. This short paper explores how the infamous ChatGPT fares when asked to sift stories from the log of an agent-based simulation featuring romantic relations between characters.

Introduction

In an average day we experience or observe a multitude of events that register in our consciousness, yet at any given point any one of us is capable of isolating a small subset of those events as being appropriate for piecing together into an interesting story to tell about our day. The successful computational modeling of such processes is at this point an open question. This task, known as *story sifting*, is being successfully addressed by other less glamorous AI techniques.

The coverage in the media of the recent triumphs of generative AI based on large language models has created the impression that they can be successful at any task. Therefore it is particularly important that they be evaluated scientifically to establish the extent of their capabilities and, if any, of their limitations. The present paper explores the applicability of one of these models to the task of extracting from a set of events a subset that, when presented in narrative form, results in a story of higher quality than the set of events presented wholesale.

Previous Work

We outline some background on the story sifting task and some basic characteristics of ChatGPT.

Story Sifting

Early work on narrative generation produced literary texts by selecting a subset of lines from an extensive source file (Montfort and Fedorova 2012). A subsequent refinement

on this technique mines sequences of events corresponding to interesting stories from the logs of agent-based simulations. James Ryan’s PhD thesis (Ryan 2018) outlines how, rather than automatically inventing stories, narrative may emerge from the activity of characters set in motion in a simulated story world, and defines the task of curating such narratives out of simulation logs as story sifting. The Felt story sifting and simulation engine (Kreminski, Dickinson, and Wardrip-Fruin 2019) introduced the concept of *story sifting patterns*, which are descriptions of sequences of events that exhibit high potential to be part of interesting narratives. He develops tools for authoring such patterns and applying them to sets of events to implement automated story sifting. This line of research led to the development of Winnow (Kreminski, Dickinson, and Mateas 2021), a domain-specific language for specifying story sifting patterns that can be run on ongoing simulations to identify event sequences with narrative potential.

As it can be seen, it is necessary to consider how well this task can be performed using LLMs, what are their possible limitations and in which cases it is worthwhile to continue using other types of techniques. Although in this contribution we do not intend to answer these questions extensively, we provide our point of view, based on our experience, on how well one of these language models, ChatGPT¹, performs this task.

ChatGPT

ChatGPT is an interactive online system that responds to textual prompts presented by the user with fluent prose that always appears to be an acceptable response to the given prompt. Although there is no recognised scientific publication that describes how ChatGPT operates, it is public knowledge that it is a member of the generative pre-trained transformer (GPT) family of language models (Radford et al. 2018), fine-tuned using reinforcement learning (MacGlashan et al. 2017; Ziegler et al. 2019). By virtue of this, it combines the advantage of a neural representation as semantics (Levy and Goldberg 2014), the linguistic fluency of transformers (Khan et al. 2022) and the knack of finding appropriate responses associated with reinforcement learn-

¹(<https://chat.openai.com>) ChatGPT Free Research Preview, Mar 23 Version (GPT 3.5)

ing. This allows it not only to respond fluently to most user requests, but also to understand clarifications or corrections and respond by adapting its prior outputs as requested by the user. It is this particular functionality that we hoped to explore in this paper for the story sifting task.

Story Sifting with ChatGPT

Story sifting involves processing an input that includes a large set of events pertaining to a specific story world to select a particular subset of those events such that, presented in a narrative fashion, they exhibit narrative qualities beyond those of the whole or any summary of it.

Input Simulation Log

We have tested the performance of ChatGPT in the story sifting task using a simulation log created by an agent-based simulation of characters that interact with each other in a romantic setting. In this simulation, characters can invite others to participate in different events together in order to increase their mutual affinity levels. The aspect of such interactions in the log is the following:

```
Suzette PROPOSE friend_chat Silvy
Silvy ACCEPT-PROPOSAL friend_chat Suzette
Silvy affinity with Suzette 57
Suzette affinity with Silvy 74
[...]
Betty PROPOSE mate_invite_dinner Robert
Robert REJECT-PROPOSAL mate_invite_dinner Betty
Robert affinity with Betty 93
Betty affinity with Robert 79
```

The log we have worked with was 2500 lines long and involved mixed interactions between 15 characters. The following types of events are considered: PROPOSE, ACCEPT-PROPOSAL and REJECT-PROPOSAL. In each case the entry starts with the active agent, and the activity in question and the potential participant follow. The affinity value between characters is not symmetrical and ranges from 0 to 100, which makes the affinity levels take the values of mate, friend, indifferent or foe.

Getting ChatGPT to Accept the Log

The first goal was to let ChatGPT read the log file and interpret it in order to start carrying out the story sifting task, and this is where we encountered the first obstacle, as the version we used only allows interaction by means of the chat, which involved pasting the content of the log file directly in the chat. Following ChatGPT's instructions, we started by pasting the full log in the chat, which resulted in an error. After several attempts reducing the size of the pasted text we ended up working with batches of 100 lines, as bigger sizes produced errors or involved a very poor performance.

The first batch with this size was correctly read and interpreted, and ChatGPT provided the following output

```
Based on the log you provided, it seems to be a record of social interactions between different individuals. The log includes information about proposals, acceptances, and rejections of social activities such as going out, having dinner, and going to the cinema. It also includes information about the affinity level between the individuals involved in each activity. Additionally, it includes some
```

```
information about negative interactions between some individuals, such as insults and ignoring. It is possible that this log is part of a simulation or game that simulates social interactions between individuals.
```

However, we encountered a new obstacle after we entered the next few batches, as ChatGPT seems to forget its own instructions and provides the following output:

```
It seems like a log of interactions between several people. Do you have any specific question or request related to this log?
```

A couple of inputs later, it decided we wanted a summary of the events in the log. We tried to circumvent these limitations in different ways (e.g trying to tell ChatGPT the number of batches we were going to paste and numbering all of them) but ChatGPT seems to forget the instructions after very few interactions (usually three). Eventually, we were forced to work with only one set of 100 lines, as ChatGPT not only forgot the instructions, but also ignored sequential inputs and tended to work with the last batch we pasted. At this point, we identified this issue as a limitation in the extent of our experiment and selected the first 100 lines of the log in order to start with the story sifting process.

Story Sifting Tests

Having reached the point where the ChatGPT system might be considered aware of the content of the log to be processed, we presented the following prompt to request that it attempt to generate a story from the log, so we could have a baseline of what we could expect next:

```
Can you write a story using this log information?
```

ChatGPT's response is reproduced below:

```
Once upon a time, in the year 2022, a group of friends had a complex social dynamic that was closely monitored by an AI system that tracked their proposals and affinities. Megan proposed to have lunch with Meredith, and the latter accepted. Meanwhile, Lester proposed to chat with Robert, but Robert rejected the proposal. Suzette proposed to chat with Silvy, and the latter accepted. Betty proposed to go out on a weekend with Clark, but Clark rejected the proposal. Meredith proposed to watch TV with Lester, but he rejected the proposal.
```

```
(The story expands for 5 similar paragraphs)
```

The story, as it can be observed, is only a little more than a direct translation from the log format into text, almost lacking any narrative quality.

Next, we asked ChatGPT to write a story using only the most significant events (which would be a general description of what story sifting consists in). The result in this case was somehow similar to the previous one, a story that omitted some potentially relevant events, specifically all the breakups that appear in the log, as they result in drastic changes in the affinities between characters. It first claimed that there were no breakups in the log, but later acknowledged that it had "missed" the events.

When asked why and how it had decided which were the most relevant events, the answer was:

```
I determined that these events were the most relevant based on the frequency and variety of the actions that took place, as well as the different relationships between the characters that they reveal. [...] The events I mentioned are some of the most frequent and varied actions that occur in the world and are therefore important to be aware of.
```

Further attempts to get more elaborated and specific responses did not shed any light to the previous explanation. Notably, the changes in the affinities between characters were deliberately ignored.

After this, we tried to highlight the importance of the affinity values by asking ChatGPT to write a story using the pairs of characters that experienced larger variations in their mutual affinity values, but the result was similar to the previous case, obtaining a story that omitted all breakups between couples.

Watching these results, we decided to be more specific about the intended output, so the following input directed the generated result towards one of the couples who brake up:

Can you write the story of a couple that breaks up?

ChatGPT's response is partially reproduced below:

Once upon a time, Megan and Lester were a happy couple. They had been together for several years and had built a life together. They enjoyed spending time together, going on adventures, and making memories. However, as time went on, they began to grow apart. They stopped communicating effectively and stopped making an effort in the relationship. Lester began to withdraw and stopped showing Megan affection. He would spend more time at work or with his friends, leaving Megan feeling alone and neglected.

One day, Megan met Meredith and they hit it off right away. They began spending more time together, going on lunch dates and watching TV together. Lester became jealous of the time Megan was spending with Meredith and rejected her proposal to watch TV together.

Meanwhile, Clark rejected Betty's proposal to go out for the weekend, leaving Betty feeling rejected and unwanted. Suzette and Silvy hit it off and began chatting regularly. Robert rejected Lester's proposal to chat, causing Lester to feel rejected and unwanted.

[...]

As it can be seen, ChatGPT makes a big effort to provide a background to the story, but it takes too short to start rambling and including events about characters that do not take part in the requested narrative thread, therefore producing a garbled result.

An even more focused attempt to obtain a specific output for the story sifting process consisted in specifying the characters that were to be considered in the narrative, with the following result:

Can you tell a story of Betty and Clark using the information in this log?

ChatGPT's response is partially reproduced below:

Betty and Clark had been friends for a while, and Betty had been hoping to spend some more time with him. One day, she proposed that they go out together for the weekend. Unfortunately, Clark rejected her proposal. Although she was disappointed, Betty tried to move on.

Meanwhile, Meredith proposed to Lester that they watch TV together as mates, but he rejected her proposal. Violet also proposed to Megan that they chat, but Megan rejected her proposal.

Megan proposed to Tony that she give him a massage, but he also rejected her proposal. However, they still had a high affinity score, indicating that they still liked each other as friends.

[...]

Again, ChatGPT decided to include other events that are present in the log instead of complying with our request. At this point, instead of trying to force a specific output, we tried to give ChatGPT a little more freedom by only restricting the number of characters that should be considered:

Can you select the two most interesting characters from this log and tell their story?

ChatGPT's response is partially reproduced below:

Meredith and Violet were once best friends. They did everything together, and everyone knew that they were inseparable. However, things took a turn when Meredith proposed to have lunch with Megan, and Violet proposed to chat with her. Meredith accepted Megan's proposal, but rejected Violet's proposal. This made Violet upset, and she proposed to hug Megan instead. However, Meredith rejected Violet's proposal.

As time passed, Meredith and Violet's relationship deteriorated. They had a falling out, and Meredith even informed others that she despised Violet. She also accused Violet of cheating on her, which further strained their relationship. Meanwhile, Meredith developed an affinity for Lester, while Violet's affinity with Meredith dwindled.

[...]

Apart from a small inconsistency in the fact that Meredith proposes Megan to have lunch but then it is Meredith, and not Megan, who accepts the proposal, reading the log it is remarkable to see that the facts in the first paragraph do not appear in it or appear in a different way. Additionally, when asked why Meredith and Violet are the most interesting characters in the log the answer is:

Overall, the interactions and dynamics between these characters have the potential to create a compelling story.

This means that it has selected the characters not only for what they have done, but for what they might do in the rest of the log according to what ChatGPT expects from what it has seen. Another plausible interpretation is that, provided that ChatGPT basically relies on a LLM, the explanation provided by ChatGPT is a carefully crafted text that does not really reflect what is happening underneath.

At this point, we decided to stop our experiments using ChatGPT for story sifting, as it was already patent that ChatGPT has a high potential to create unrestricted narratives, but it lacks ability to perform tasks such as story sifting, which require sticking to a plan in a more consistent way.

Observed Limitations

The following limitations have been observed during the experiments carried out:

- ChatGPT seems to forget the instructions after very few interactions, which limits the possibility of working with a big amount of data
- ChatGPT ignores sequential inputs and tends to work with the last batch received
- ChatGPT's response to a set of events appears to be restricted to summarising it, very neatly but with no option of selecting a particular subset based on its potential narrative qualities
- ChatGPT's output is quite biased towards adding its own content, thus making it difficult to restrict the result to the data provided as input
- ChatGPT's criteria to perform story sifting operations are vaguely explainable, generic and difficult to influence in order to obtain results that follow different criteria or to adjust the to a specific domain

As a result, our impression is that there is still room for improvement in relation to LLM-based story sifting, and consequently there is still need to keep on using and researching on other techniques for this task.

Comparison with an Evolutionary Sifting Method

In order to make our position stronger and show that it is possible to outperform ChatGPT in the story sifting task, we have used the original log file containing 2500 lines as an input for a story sifting algorithm specifically crafted for this task. It has been developed using a genetic algorithm that makes use of heuristics based on the affinity values in order to optimize the story sifting process. The output of this algorithm is partially reproduced below:

```
ProposeActivity-544 mate_go_to_cinema Lester Meredith
ActivityRejected-545 mate_go_to_cinema Lester Meredith
ProposeActivity-629 mate_dinner_with_candles Lester Meredith
ActivityRejected-631 mate_dinner_with_candles Lester Meredith
ProposeActivity-747 mate_hug Lester Meredith
ActivityRejected-748 mate_hug Lester Meredith
ProposeActivity-975 mate_watch_tv Lester Meredith
ActivityAccepted-976 mate_watch_tv Lester Meredith
```

In this case, we acknowledge the superior quality of the texts generated by Large Language Models, and more specifically ChatGPT, over other techniques, so we have used it in order to generate the final (partial) story, as shown below:

Lester had been trying to plan the perfect date for weeks. He wanted something special, something that would show Meredith just how much he cared about her. But every idea he proposed seemed to fall flat.

First, he suggested they go to the cinema to see a romantic movie. Meredith wasn't interested and rejected the idea. Then, he proposed a dinner with candles, hoping to create a romantic atmosphere. But again, Meredith rejected the idea.

Feeling dejected, Lester decided to try something simpler. He proposed they just spend some quality time together watching TV. Meredith accepted, and they settled in on the couch with some snacks and drinks.

As they watched their favorite show, Lester put his arm around Meredith and gave her a hug. She leaned into him, and they both felt the warmth and comfort of each other's embrace.

In that moment, Lester realized that it wasn't about the grand gestures or elaborate plans. It was about the simple moments of connection that made all the difference. And as they sat there together, watching TV and cuddling, he knew that he had found something special with Meredith.

It is impressive to see how ChatGPT has managed to expand the bare sketch of events provided in the log into a moving story. The elements sifted from the log as pertaining to a potentially interesting story have been expanded into paragraphs that enrich the story with detail and nuance. Even so, the example provided shows an ideal output generated by ChatGPT, as previous attempts also included events that were kindly provided by it, but which were not present in the output generated by the genetic algorithm. It is clear that solutions like ChatGPT have a significant potential in terms of what they contribute to story generation. However, they still have room for improvement in terms of specific tasks such as story sifting.

In order to compare both models in similar conditions, in Table 1 we show the selection made by each of them over a log of 100 lines.

Potential Applications

It is clear from the reported results that the evolutionary story sifting process achieves better results in terms of identifying a relevant subset of events from a given log but presents a poor textual rendition. At the same time, it is clear that ChatGPT has difficulty in grasping the concept of "selecting a relevant subset of events" and inserts events that were not present in the log, but generates a text that is impressive by its fluency. These two observations suggest that the optimal solution might be found in an informed combination of the two approaches. In such a combination the evolutionary sifting algorithm might be relied upon to carry out the selection of the relevant events and ChatGPT to render the resulting selection as text. Any such combination may need to be refined to ensure that any events hallucinated by the neural solutions are identified and filtered out.

Potential applications of such models would include automated pre-processing of system logs or surveillance records to highlight relevant sequences of events, or automated generation of relevant narrative threads in video games.

Conclusions

The experiments carried out show that it is at present difficult to get ChatGPT to carry out processes of story sifting from simulation logs, as this is a knowledge related task more than a simply language related one. Although the prose of the responses is fluent and sounds natural, the concept of focusing on narrative threads that restrict the narrative to particular subsets of the characters appears to be beyond the current capabilities of the system. It is possible that fine-tuning the underlying language model with a story-sifting specific dataset may improve the results produced by these models, but this is something that still needs to be researched.

Successive efforts of prompt engineering angling for the appropriate responses have not found the expected result. This is in spite of the impressive ability of the system to come up with appropriate responses to most requests.

The results obtained generally constitute valid summaries of the material provided, often presented in reasonable narrative form and in fluent prose. However, they tend not to satisfy the requirements for a valid process of story sifting, which are different from those of simple summarisation. Whereas summarisation involves finding a significantly shorter rendering of pretty much the same material in the input, story sifting should involve a process of deciding to focus on a subset of the material such that the narrative quality of the result is significantly higher than the original input or any summary of it. This concept appears to present difficulties to ChatGPT.

We nevertheless believe that there is significant potential in the idea of applying neural solutions to the story sifting task.

As further work, we intend to explore the use of alternative neural language models to check if they present similar limitations and propose approaches to overcome them.

Action	Evo	Chat
Megan PROPOSE have lunch Meredith		X
Lester PROPOSE chat Robert		X
Suzette PROPOSE chat Silvy		X
Betty PROPOSE weekend out Clark	X	X
Meredith PROPOSE watch tv Lester		X
Clark REJECT weekend out Betty	X	X
Lester REJECT watch tv Meredith		X
Meredith ACCEPT have lunch Megan		X
Violet PROPOSE chat Megan	X	X
Robert REJECT chat Lester		X
Silvy ACCEPT chat Suzette		X
Clark PROPOSE hug Betty		X
Betty PROPOSE invite dinner Robert		
Betty REJECT hug Clark		X
Robert REJECT invite dinner Betty		
Megan PROPOSE give massage Tony		X
Lester PROPOSE sleep together Meredith		
Mary PROPOSE play tennis Megan		X
Silvy ACCEPT chat Suzette		X
Meredith INFORM despise Violet		X
Tony REJECT give massage Megan		X
Megan REJECT chat Violet	X	X
Violet PROPOSE hug Megan		
Meredith INFORM cheat Violet		X
Megan PROPOSE go out John		
Betty INFORM break up Robert		
Mary PROPOSE help Silvy		
Suzette INFORM slander Mary		
Clark PROPOSE hug Ray		
Ray ACCEPT hug Clark		
Meredith REJECT sleep together Lester		
Betty PROPOSE have lunch Clark	X	
Clark ACCEPT have lunch Betty	X	
Suzette PROPOSE hug Silvy		
Silvy PROPOSE weekend together Drew		
Silvy ACCEPT hug Suzette		
Drew PROPOSE help Simon		
Silvy INFORM cheat Suzette		
Mary PROPOSE help Silvy		
Silvy REJECT help Mary		
Suzette PROPOSE have coffe Silvy		
Silvy PROPOSE talk Drew		
Silvy ACCEPT have coffe Suzette		
Drew ACCEPT talk Silvy		
Mary PROPOSE chat Megan		

Table 1: Comparison between a story sifting operation performed by the evolutionary algorithm and ChatGPT. The columns marked with X show the actions selected by each of them over a log of 100 lines (lines with changes in affinity levels have been removed for clarity and space reasons). The evolutionary algorithm tends to take into account all the events, while ChatGPT tends to focus more on the initial events

Acknowledgments

This paper has been partially funded by the projects CAN-TOR: Automated Composition of Personal Narratives as an aid for Occupational Therapy based on Reminiscence, Grant No. PID2019-108927RB-I00 (Spanish Ministry of Science and Innovation); and ADARVE: Análisis de Datos de Realidad Virtual para Emergencias Radiológicas, Grant No. SUBV-20/2021, funded by the Spanish Consejo de Seguridad Nuclear (CSN).

References

- Khan, S.; Naseer, M.; Hayat, M.; Zamir, S. W.; Khan, F. S.; and Shah, M. 2022. Transformers in vision: A survey. *ACM computing surveys (CSUR)* 54(10s):1–41.
- Kreminski, M.; Dickinson, M.; and Mateas, M. 2021. Winnow: A domain-specific language for incremental story sifting. *Proc. of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment* 17(1):156–163.
- Kreminski, M.; Dickinson, M.; and Wardrip-Fruin, N. 2019. Felt: A simple story sifter. In *12th International Conference on Interactive Digital Storytelling, ICIDS 2019*, 267–281. Berlin, Heidelberg: Springer-Verlag.
- Levy, O., and Goldberg, Y. 2014. Dependency-based word embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 302–308.
- MacGlashan, J.; Ho, M. K.; Loftin, R.; Peng, B.; Wang, G.; Roberts, D. L.; Taylor, M. E.; and Littman, M. L. 2017. Interactive learning from policy-dependent human feedback. In *International Conference on Machine Learning*, 2285–2294. PMLR.
- Montfort, N., and Fedorova, N. 2012. Small-scale systems and computational creativity. In *International conference on computational creativity*, volume 82.
- Radford, A.; Narasimhan, K.; Salimans, T.; Sutskever, I.; et al. 2018. Improving language understanding by generative pre-training. *Journal of*.
- Ryan, J. 2018. *Curating Simulated Storyworlds*. Ph.D. Dissertation, University of California Santa Cruz, CA, USA.
- Ziegler, D. M.; Stiennon, N.; Wu, J.; Brown, T. B.; Radford, A.; Amodei, D.; Christiano, P.; and Irving, G. 2019. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*.

Author Contributions

Gonzalo Méndez carried out the experiments with ChatGPT and wrote the initial draft of the paper. Pablo Gervás revised the draft and elaborated on previous work on neural models. Both authors revised the paper several times and jointly developed the discussion and the conclusions.