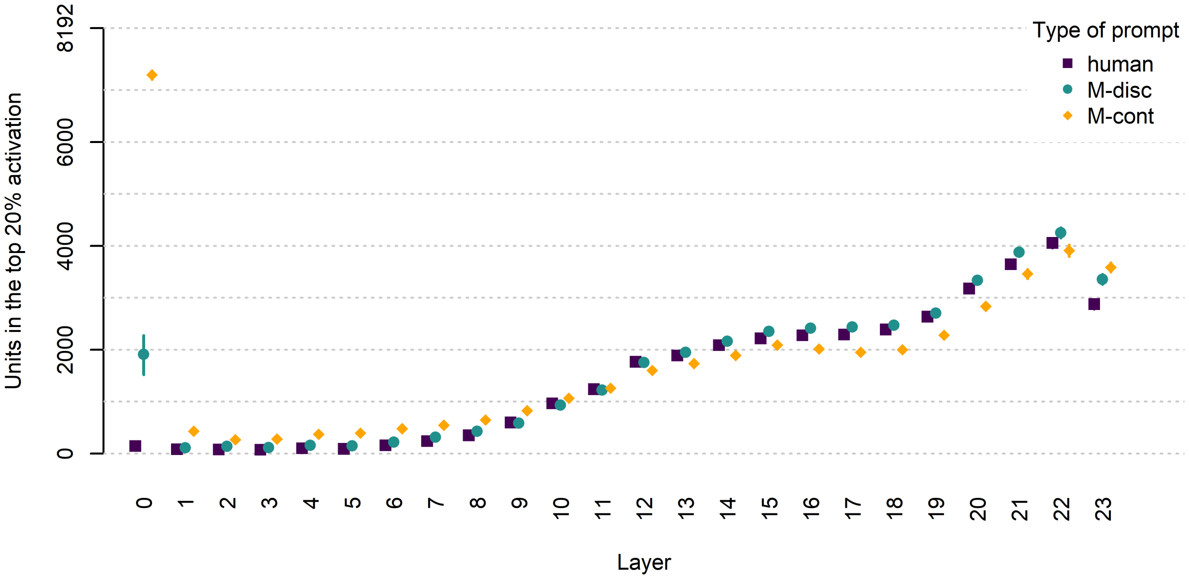

Figure 3: For each prompt type, we plot the number of units belonging to the top 20% most activated units (overall across prompt types). *M-cont* and *M-disc* have significantly more highly activated units than *human* on the first layer, with the effect particularly strong for *M-cont*. There is also a weaker tendency for the machine prompts to activate more units on the last layer compared to the *human* ones. Data from OPT-1.3b.

rons). We define the typical units for prompt A as those that are among the top 20% most frequently activated by this prompt type, while at the same time being among the bottom 20% least frequently activated by prompt types B and C.[5] This filtering procedure leaves 14 *human* (resp. 6), 4 *M-disc* (resp. 4) and 58 *M-cont* (resp. 238) units for OPT-350m (resp. OPT-1.3b). As a sanity check, we also repeated the analysis with laxer thresholds involving more units, and the results were similar to the ones we report here.

Next, we associate each selected unit to a set of items from the LM vocabulary that strongly trigger its activation. Using the Wikipedia corpus,[6] for each item in the vocabulary we save the average unit activation in a forward pass. We sort the resulting matrix to get, for each unit, the top 500 vocabulary items leading to the strongest activation. We extract both input items, recording unit activation when an item is in the input sequence, and output items, recording activation when an item is predicted by the LM. We apply lower-casing and initial-space stripping on the resulting vocabulary set. More details are provided in Appendix C. This method has been chosen for its simplicity. However, it is also noisy and sensitive to rare but “exciting” tokens (e.g., *magikarp*, see Appendix C Table 4). Improving unit-item association extraction is left to future work.

Having obtained the list of vocabulary items associated to each unit, we count how many times each vocabulary item occurrs in association with any typical unit of each prompt type (as defined at the beginning of this Methodology paragraph), obtaining 3 frequency lists, one for each prompt type. We compare the relative frequencies of each vocabulary item in each list to determine which vocabulary items are most distinctively associated to (the set of typical units of) each type. In particular, using a standard method from corpus linguistics, we compute the *local Mutual Information* score (Evert, 2005) between each vocabulary item v and each prompt type t:

$$\mathrm{LMI} = |v,t| \log \frac{P(v,t)}{P(v)P(t)}$$

where $|v,t|$ counts the occurrences of v in the t list, the joint probability $P(v,t)$ is estimated based on $|v,t|$; $P(v)$ is estimated using the cumulative occurrence count of v in all lists, and $P(t)$ is the total number of occurrences of any item in the t list. Table 3 reports the top-30 *input* vocabulary items ranked by LMI for each prompt type and both LMs.

Machine prompts recruit “non-linguistic” units Looking at the OPT-350m results first, nearly all characteristic *human* items are well-formed words,

[5] A unit is activated when its value is greater than 0.

[6] Subset “`20220301.en`” from HuggingFace

human			*M-disc*			*M-cont*		
				OPT-350m				
whats	gazed	ful	handler	(&	361	ÔøΩ	stat	////
name	nifty	darn	expr	avascript	ancel	ÔøΩÔøΩ	page	pwr
why	devs	freaking	iterator	cpp	yout	});	{\	sts
fuck	much	these	terness	addons	risome	());	0000	]}
noticed	like	have	hillary	\-	frameworks	....	+=	table
really	daddy	wanna	filename	lication	ithub	crossref	};	interstit.
thats	likes	what	easy	702	ÔøΩ	println	stats	/**
does	honestly	crappy	disabled	502	errors	warn	));	({
thing	workaround	relent	rc	601	poons	})	crip	/////////
goddamn	bothers	been	json	sacrific	inline	————	–	debug
				OPT-1.3b				
undermines	severe	conducive	â©	äĥ®	attle	âķg̀	leilan	looph
curls	dictates	frowned	appalling	extreme	cram	âłg̀âłg̀âł	âłâłâłâł	^^^^
makes	tempt	optimized	mal	eff	dop	endif	everal	âκĵ
will	unfold	remain	early	monitor	egregious	canaver	archdemon	xff
does	bounces	reap	complex	schizophren	rep	citiz	marketable	../
manifests	persist	stroll	hou	rece	pass	%%%%	//	0000
prevail	outweigh	shines	gres	fail	insanely	ó	âħ¢:	aeper
haunt	haun	hangs	crazy	kinda	ãĥ¨	âķ	dilig	nanto
meshes	smokes	governs	delay	shitty	prototype	%%	....	cryst
wipes	poised	fills	capital	///	devices	#####	â·â·	leban

Table 3: Top 30 vocabulary items associated to each prompt type ranked by LMI. Machine-generated prompts respond to less language-like items than those triggered by human prompts. Nearly all *human* items are well-formed words. Many *M-disc* items, on the other hand, are non-English diacritics, special symbols or code-related terms. *M-cont* items are entirely "non-linguistic". Some strings have been abbreviated to fit column width.

and include a high number of forms cuing syntactic processing, such as function elements (*whats, why, does*...), inflected verbs (*noticed, gazed, liked*...) and modifiers (*really, much, honestly*...). A remarkable amount of *M-disc* items are coding-related terms (*handler, expr, iterator*...). Numbers and punctuation sequences that could be coding-related or web-page boilerplate (*(&,* \-) also appear, as well as a few regular words or word fragments (*Hillary, easy, sacrific*...). Finally, for *M-cont*, the items are entirely "non-linguistic", being composed of sequences of non-Latin characters or punctuation marks, as well as code fragments.

Concerning OPT-1.3b, we observe pretty much the same patterns for *human* and *M-cont*. For *M-disc*, on the other hand, together with a number of non-English diacritics and special symbols, there is a strong increase in regular words and word fragments, although the latter still clearly differ from those associated to *human* prompts, in that syntax-related items, such as function words or inflected verb forms, are largely missing. In line with what we observed in Figure 1, we thus observe a cline on which, at least for *M-disc*, the difference in processing human and machine-generating prompts decreases with model size.

We tentatively conclude that machine prompts are not only triggering different activation pathways, but that the units involved in these pathways tend to respond to less language-like items than those triggered by human prompts. Note that these units are spread across the layers of the network, so that we are not only recording low-level differences in processing the input strings or vectors.[7] Moreover, the results are largely mirrored by those obtained when associating units with output instead of input vocabulary items (Table 5 in App. D).

Recall that our analysis is based on units that are not only highly typical of a prompt type across relations, but also in the top gradient quartile, suggesting that they significantly contribute to the model's output distribution. It is puzzling that units that mostly respond to coding fragments or unusual characters could lead the network to produce the correct next token in the semantic tasks we are studying. We conjecture that distributed activation from such units might nudge the network towards the right output semantic fields through connectiv-

[7] The selected typical *human* units occur in layers 4th to last of OPT-350m and layers 2 to 14 of OPT-1.3b (counting from 0). The *M-disc* units range from layers 3 to 10 of OPT-350m and 2 to 13 of OPT-1.3b. The *M-cont* units are the only ones where, as expected given the distribution illustrated in Figure 3, a significant proportion occurs on the first (0-th) layer (about one third for OPT-350m and one fourth for OPT-1.3b), but the remaining two thirds/three fourths range from layers 2 to 21 and 1 to 10, respectively.

ity pathways that fortuitously arose during network training. This is an important topic for future work.

6 Discussion

We have studied the phenomenon of linguistically and semantically opaque machine-generated prompts from the perspective of how LMs process them, compared to human-crafted ones. Our study has important **Limitations**, that are discussed in the relevant section below. However, at least for the prompt generation methods, LMs and tasks we explored, we can draw some general conclusions.

More than a "happy accident" Our evidence suggests that the differences between human and machine-generated prompts are not just superficial, but affect all levels of network processing, and result in the activations of qualitatively different units. Some of these units are stable across semantic tasks, suggesting that they are more generally recruited to process any "unnatural" input. Moreover, contrary to what one could reasonably predict, there is some evidence that machine prompts are more robust than human ones, in the sense that they achieve better output calibration.

It's unlikely that the LM has been exposed to anything like *M-disc* prompts during its initial training, and definitely it could not have seen out-of-vocabulary *M-cont* prompts, so we can only assume that the special pathways triggered by these prompts arose through unforeseen side effects of pre-training.[8] However, they seem to be more than just lucky connectivity accidents exploited by specific prompts to solve specific tasks, or else it would be difficult to explain the overall low entropy of machine prompt predictions and the commonalities in the units they activate. Moreover, there is evidence that *M-disc* prompts can transfer across Transformer-based LMs (Rakotonirina et al., 2023; Zou et al., 2023), suggesting that unnatural language pathways might arise from the interaction of general characteristics of the Transformer architecture with Web-derived training data that are partially shared across many current pre-trained LMs. We must defer a better understanding of the nature of these unnatural pathways to future work. In particular, we plan to zoom further in into the processing of specific templates, tracking their processing throughout the network with methods such as the vocabulary-based unit analysis of Section 5.

On investigating unnatural language We believe that investigating "unnatural language" as we did here (see also Khashabi et al., 2022; Ishibashi et al., 2023; Rakotonirina et al., 2023) should be a central concern to NLP for at least three reasons.

First, *understanding* why LMs work as well as they do, and what are their failure modes, is one of the questions with the broadest scientific and societal implications we can ask today. It would however be dangerously limiting to narrow our investigation to how LMs process *natural* language only, ignoring their behaviour when presented inputs outside their training distribution.

Second, unnatural language can be exploited for negative purposes, as shown by Wallace et al. (2019) and Zou et al. (2023), who derived apparently nonsensical prompts that could steer multiple LMs' responses towards harmful behaviour, such as generating racist language.

Finally, there is recent interest in letting LMs directly communicate with each other to jointly solve tasks or to build a community (Park et al., 2023; Zeng et al., 2023). Based on our evidence, it might be pointless to insist that LM-to-LM communication takes place in natural language, given that LMs might share information more efficiently through unnatural prompts. Conversely, if being able to decode the communication flow is important (e.g., for safety reasons), care must be taken to stop LMs from drifting into unnatural language.

For all these reasons, we hope that our preliminary contribution will encourage our community to pay more attention to the phenomenon of unnatural language processing.

[8] We experimentally verified that model training is necessary for effective unnatural prompts to arise. We ran both Autoprompt and Optiprompt on 3 distinct random initializations of OPT-1.3b with the same hyperparameters as in our main experiments, and found that the resulting *M-disc* prompts achieved 0% accuracy in nearly all cases, whereas *M-cont* where at best able to retrieve the majority output of a task.