A Measuring the overlap of activated knowledge neurons

In Section 4, we measure and compare which knowledge neurons, that is, units found in the intermediate layers of the feed-forward Transformer blocks (Dai et al., 2022; Geva et al., 2022), are activated by different prompt types. In particular, we measure the *knowledge neuron activation overlap* (abbreviated as activation overlap). The following algorithms in pseudo-code detail how this measure is obtained.

First, we construct a Boolean matrix recording which units are activated by a template. As shown in the pseudo-code below, a unit is said to be activated if its value is greater than 0 on more than $k = 20\%$ of cases when instantiated with each of the subjects associated to the template relation. In the pseudo-code, `Relations` is the set of relations available in our task set; given a relation, `Subjects` provides the list of relevant subjects and `template(s)` instantiates the template with subject `s`. `Model` is the LM being used.

```
def get_act(template, relation):
    cpt = 0
    k = 0.2
    # iterate across subject
    for s in Subjects(relation):
        inpt = template(s)
        for u in Model(inpt).units:
            cpt[u] += (u>0)
    # u is activated if >0 for
    # more than k% of inputs
    act = cpt > k*len(Subjects(relation))
    return act
```

Then, for each pair of templates (in `Templates`), we compute the intersection over union of their respective activation matrices:

```
overlap = {}
for r in Relations:
    for t_A in Templates(r):
        for t_B in Templates(r):
            act_A = get_act(t_A,r)
            act_B = get_act(t_B,r)
            i = act_A & act_B
            u = act_A | act_B
            overlap[(t_A, t_B)] = i/u
```

Finally, we average these pairwise overlaps while filtering by prompt type (e.g., only averaging the activation overlap for pairs containing one *human* and one *M-cont*).

Layer 0, unit 248		
“(	¿¿¿¿¿¿¿	.....
[_	.....	=”/
::::::::	{:	,,,,
Layer 10, unit 674		
antidepress	debian	frieza
magikarp	minecraft	xperia
oneplus	awakens	bitcoin
Layer 16, unit 2126		
filename	windows	drm
sshd	misunder	rm
folder	pkg	vm ‘
Layer 22, unit 3617		
everal	huge	every
risome	some	any
these	crappy	nifty

Table 4: For each intermediate unit in the OPT's feed-forward layers we extract the set of items leading to the strongest activation on the Wikipedia corpus. As an illustration, we selected four different units with distinct profiles and displayed them in this table, along with their top 9 most associated input items, for OPT-350m. We observe varying degrees of consistency and naturalness across units.

B Diagnostic classifiers

To complement the activation-overlap-based analysis presented in Section 4.1 of the main paper, we run a set of shallow linear “diagnostic” classifiers (Giulianelli et al., 2018) of the activations generated by the models on each layer in response to inputs from each prompt type. As usual, we focus on the activation of knowledge neurons.

Data As in the main paper, we use all templates with LAMA accuracy $>= 10\%$, filtering out a random subset of the LAMA P176 relation *human* templates, as this relation is greatly over-represented. We are left with 21 and 24 tasks for OPT-350m and OPT-1.3b, respectively. As we are interested in units inherently distinguishing prompt types independently of lexico-semantic aspects associated to specific templates or tasks, we partition the data so that the training and test data contain disjoint tasks (and, *a fortiori*, disjoint templates). We consider 4 such partitions, each time using data from 16 (OPT-350m)/18 (OPT-1.3b) tasks for training and 5 (OPT-350m) /6 (OPT-1.3b) for testing, such that

there is no test task overlap across the partitions.[9] We instantiate each template with 10 randomly selected subjects from the corresponding LAMA lists. For each pairwise classification task, we balance the test instances by downsampling the larger class, so that chance/majority/minority accuracies ("baselines" in Figure 4) are at 50%.

Classifier We use a logistic regression classifier with L1 regularization (to encourage sparseness), with the L1 term coefficient fixed at $\alpha = 0.01$. We fit the classifier with stochastic gradient descent, using the Scikit-learn toolkit (Pedregosa et al., 2011). For each of the 4 training partitions, we repeat the experiment with 5 different seeds, resulting in a total of 20 runs for each layer.

Results. Figure 4 reports average per-layer classification accuracies for each pairwise prompt type comparison, with standard deviations across the 20 runs. In all cases, accuracy values are well above baseline level, and typically very high. We conclude that each single layer contains units whose activation is sufficiently discriminative for each prompt type to successfully train the classifiers, despite the challenging setup in which training and test tasks (and consequently templates) are completely disjoint. Interestingly, few units on each layer suffice to discriminate prompt types (the average classifier weight sparsity across all experiments is at 99.5% with 0.2% standard deviation for OPT-350m and 99.6% with 0.4% s.d. for OPT-1.3b). The easiest distinctions involve *M-cont* as one of the classes, confirming that out-of-vocabulary embeddings make continuous prompts particularly different from natural language. Indeed, it's remarkable that distinguishing *M-disc* from *M-cont* is generally easier than distinguishing between *M-disc* and *human* prompts.

To conclude, the classification experiments bring strong convergent evidence that genuinely different pathways characterize different prompt types across all layers of the network.

C Unit/vocabulary item association

Implementation details When extracting unit/vocabulary-item association, we empirically set the window size to 15. This value is reasonable close to prompt size (5 in average), while containing a sufficient amount of tokens to get a meaningful context. In addition, we set the window stride to 15 to save computation time. Furthermore, due to our limited computation resources, and given the size of the Wikipedia corpus (6B), we only used 66% of the data for OPT-1.3b.

Samples As illustrated in Table 4, we came up with a large diversity of unit profiles, some of them being associated to more or less linguistically valid items, and with varying degree of semantic consistency.

D Profiling typical units by output vocabulary analysis

Table 3 in the main paper reports the 30 *input* vocabulary items with the largest LMI with respect to each prompt type. Table 5 here reports the top-30 *output* items. We largely confirm the same trends, although we do notice an overall tendency for the units triggered by machine prompts to be associated to more "language-like" output material, which makes sense as ultimately these prompts do produce well-formed task-relevant outputs.

[9] As we have 21 total tasks meeting our conditions for OPT-350m, one of the 4 partitions used for this model consists of 15 training and 6 test tasks

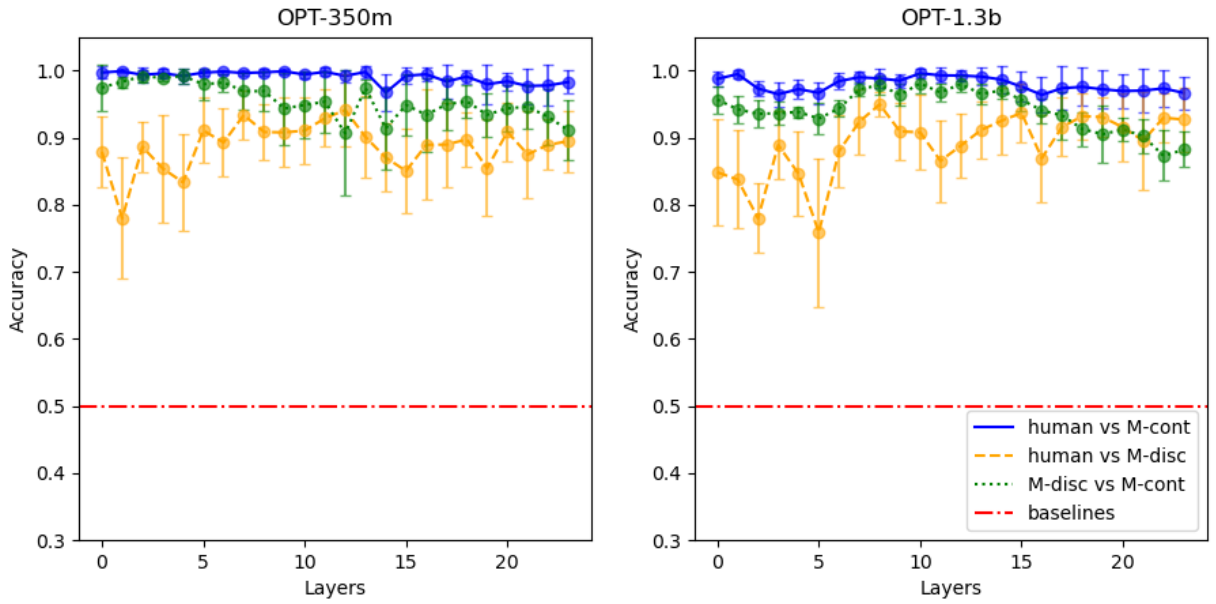

Figure 4: Average accuracies and standard deviations for 20 runs of the pairwise prompt-type classification experiments across the 24 layers of the networks.

human			*M-disc*			*M-cont*		
			OPT-350m					
nobody	undone	tonight	incompet	fanc	{	ÔøΩ	,ñà	compare
yesterday	before	scrambled	alluded	ridic	\’	ÔøΩÔøΩ	”,	pref
okay	anybody	reasonably	\\	782	¨%	tracker	});	also
really	plugged	messed	):	xcom	blat	;;	moreover	comple
awoken	captcha	pinpoint	juxtap	”:[”	forge	checking	[’	attempt
bother	interchangeable	right	”\	-—	physic	supported	therefore	’\);
corrobor	glanced	authenticated	revert	698	tyr	meanwhile	text	0000
parsed	glean	snowball	insin	({	dst	avg	—–	((
bothering	tweaked	earlier	faintly	invoke	772	insert	¿¿¿	currently
nailed	figured	tasted	irresist	mysql	ceremon	header	},”	prev
			OPT-1.3b					
accumulating	snug	peeled	reconc	oddly	localhost	âkǵ	â¯	âłġâłġâł
elic	attributable	distinctly	monstrous	vaugh	filib	âłâł	â·â	ǒł
overpower	solely	ooz	trembling	prett	though	âĺı	ãń ǐ	inst
substantially	unmatched	waging	check	outlandish	recip	ìſ	âłġâł	}{
fundamentally	overloaded	incred	adolesc	mpg	disag	kinnikuman	say	âĩij
reinvest	obligatory	impecc	âī¼	pause	acquies	%%%%	+=	âĺĩ
unparalleled	infused	deprecated	understanding	independ	murky	===	any	*****
anonym	constrained	inherently	collaps	bicy	budgetary	services	sample	ï¸ı
overseen	sandwic	surg	disbel	scram	game	provider	â¶	ãĥĩ
ideally	achievable	tempted	unpop	foundational	billboards	âłâłâłâłâłâł	url	^^^^

Table 5: Top 30 output items associated to each prompt type ranked by LMI. Some strings have been abbreviated to fit column width.