TTL: Test-time Textual Learning for OOD Detection with Pretrained Vision-Language Models

Supplementary Material

A. Basic statement

A.1. The Use of Large Language Models

Throughout the entire work, we use ChatGPT for language polishing and code assistance.

A.2. Reproducibility Statement

we will make the complete implementation code, including model training configurations and evaluation scripts, publicly available on GitHub upon the acceptance of this paper. Additionally, we will provide detailed documentation specifying dependencies and step-by-step instructions to replicate our experiments. This ensures that researchers can easily verify our findings and build upon our method for future work.

A.3. Detailed experiment setting

Experimental Details The proposed TTOD method was implemented using Python 3.9 and PyTorch 2.3.0, with all experiments conducted on a single NVIDIA GeForce RTX 3090 GPU. Following prior work [24, 29, 37, 49, 52], we adopted ViT-B/16 [11] as the backbone model. The OOD prompt was optimized via AdamW [18] with a learning rate of 0.005 and batch size of 64. We set $N_q = 512$ and $\tau = 1$, and used MCM [29] as the base OOD detector across all experiments.

TTL includes four hyperparameters: (1) Increasing the loss weight generally improves performance. However, very large weights over-emphasize pseudo-labeled OOD samples. From a practical deployment perspective, assuming an extremely unreliable base detector is unrealistic. Therefore, we select a moderate weight (i.e., $\alpha = 0.5$), which emphasizes OOD knowledge learning without overly trusting noisy pseudo-labels—even though this is not the weight that yields the highest peak performance; (2) OKB capacity: TTL is highly robust to the choice of OKB size. To ensure fair comparison with OODD—which maintains a visual memory bank—we adopt the same capacity, setting $K = 2048$; (3) A larger batch size provides more comprehensive pseudo-label signals within each update, enabling more stable adaptation and generally improving detection performance. It also accelerates inference because more samples can be processed in parallel. However, when the batch size becomes excessively large, the number of adaptation steps per test stream decreases dramatically, leading to slower learning of new OOD semantics. This ultimately harms performance in streaming evaluation. To balance adaptation quality, learning efficiency, and inference throughput, we adopt a moderate batch size of $B = 64$.; and (4) The fusion coefficient β differs across datasets mainly because the base detector scores have different magnitudes. On CIFAR-100 and ImageNet-1k, the MCM scores vary by roughly one order of magnitude, and accordingly, the chosen fusion coefficients also differ by about one order of magnitude. Specifically, β is set to 0.006 for CIFAR-100 and 0.0005 for ImageNet-1k.

Algorithm 1 Test-time Textual OOD Learning

Require: test data stream $\{x_i\}_{i=1}^{T}$, text encoder $g(\cdot)$, image encoder $f(\cdot)$, text prompt for ID u_c^{id}, learnable text prompt for OOD u_c^{ood}, batch size B, the priority queue OKB with capacity K

Initialize OOD prompt: $u_c^{ood} = u_c^{id}$
Compute and obtain ID text embeddings: $t_c^{id} = g(u_c^{id})$
for each data sample $x_i \in \mathcal{D}$ **do**
  Calculate base OOD score for x_i using the base detector: $s_{\text{base}}(x_i)$
  Compute adaptive threshold: λ by Equal 2
  Assign pseudo-label to x_i: $\hat{y}$
  Obtain current OOD text embeddings: $t_c^{ood} = g(u_c^{ood})$
  Compute OOD probability p by Equal 3
  Update queue Q: $\mathcal{Q} \leftarrow \mathcal{Q} \cup \{\hat{y}, p\}$
  if $len(Q) = B$ **then**
    train u_c^{ood}:
    Calculate loss $\mathcal{L}_{\text{OMB}}$ loss by Equal 4, input data: Q
    Compute grouping threshold θ by Equal 2
    Calculate loss $\mathcal{L}_{\text{OKP}}$ by Equal 5, input data: Q, θ
    Update u_c^{ood} using $\mathcal{L}_{\text{OMB}}$ and $\mathcal{L}_{\text{OKP}}$
    Obtain updated OOD prompt embeddings: $t_c^{ood} = g(u_c^{ood})$
    Score the updated OOD prompt embeddings by Equal 7
    Store each OOD prompt embedding into the priority queue OKB
    $Q \leftarrow \emptyset$
  end if
  Perform inference correction and prediction:
  Compute final OOD score $S_{final}(x_i)$ by Equal 9
end for
return $S_{final}(x_i)$ for all samples in $\mathcal{D}$

Calibration Strategies. Several calibration strategies were

Table 9. Comparison of different calibration strategies.

Function	Formula	FPR95↓	AUROC↑
MaxSim	$-\max_{j\in\{1,\dots,K\}}\cos(\mathbf{z}_x,\mathbf{t}_j^{\text{ood}})$	20.52	95.57
ExpSum	$-\sum_{j=1}^{K}\exp\left(\cos(\mathbf{z},\mathbf{t}_j^{\text{ood}})/\tau\right)$	19.61	95.44
IDR	$\frac{\sum_{j=1}^{N}\exp\left(\cos(\mathbf{z},\mathbf{t}_j^{\text{id}})/\tau\right)}{\sum_{j=1}^{N}\exp\left(\cos(\mathbf{z},\mathbf{t}_j^{\text{id}})/\tau\right)+\sum_{j=1}^{K}\exp\left(\cos(\mathbf{z},\mathbf{t}_j^{\text{ood}})/\tau\right)}$	22.85	94.74
Ours	$S_{\text{base}}(\mathbf{x})-\beta\cdot S_{\text{cal}}(\mathbf{x})$	**12.46**	**97.29**

adopted, and the specific formulas are shown in Table 9.

More Details about Adaptive Threshold. OWTTT [23] searches for the optimal parameter λ using a fixed step size of 0.01 between 0 and 1, which is actually unsuitable for different OOD scores (e.g., using ImageNet as ID data and the SUN dataset as OOD data, a step size of 0.01 could span the entire range of MCM Scores across all samples). Here, we propose using the minimum score encountered during testing as the lower bound for the search and the maximum score as the upper bound. Within this range, we uniformly divide the interval into segments matching OWTTT's approach to search for the optimal parameter λ.

B. Algorithms

This section provides a detailed breakdown of Algorithm 1 (Test-time Textual OOD Learning), complementing the core description in the main text. The algorithm aims to dynamically adapt OOD text prompts during testing, leveraging textual information to enhance OOD detection performance without relying on external pre-defined OOD categories.

C. Full Results of Ablation Studies

Sensitivity of Basic OOD Detector. Please refer to Table 8. We can see that we have achieved a positive improvement in performance for different basic detectors. And it's for all datasets, not just the average results. The study shows the insensitivity to the selection of base OOD detetcor for TTL.

D. Additional Results

Designs of scoring function with OOD prompts. Given the learned OOD prompts, Four strategies for generating final calibrated prediction S_{final} using discovered OOD knowledge are evaluated on ImageNet-1k benchmark: (1) Max OOD Similarity (MaxSim); (2) Exponentiated Sum OOD (ExpSum); and (3) ID-OOD Softmax Ratio (IDR) [17]; As shown in Table 9, the most effective strategy is ours, which directly subtracts OKB similarity from the base detector score. This show that OOD knowledge functions as a complementary corrective signal, with effectiveness depending primarily on knowledge quality rather than sophisticated scoring mechanisms.

Different Backbone Architectures. Please refer to Table 10. As shown in the results, TTOD still achieved the best performance under different backbones.

Ordering of Testing Data. Test-time adaptation methods are inevitably affected by the order in which the test samples arrive. To rigorously test this aspect, we randomly shuffled the order of the test data using three distinct seeds. We observed that our method exhibits robustness to changes in the ordering of test data. Specifically, across three experiments conducted on the ImageNet dataset, the AUROC scores were 97.34%, 97.2%, and 97.34%, respectively, demonstrating fluctuations of less than 0.2%. We report the average results from three random runs in our paper.

Effect of $\mathcal{L}_{\text{OKP}}$. As shown in Figure 7, with $\mathcal{L}_{\text{OKP}}$, the collected pseudo-labeled OOD set progressively splits into a much clearer bimodal distribution over time. This confirms that $\mathcal{L}_{\text{OKP}}$ successfully separates OOD samples from ID boundary samples, the source of contamination, ultimately enabling more reliable test-time adaptation for OOD detection.

Variants of $\mathcal{L}_{\text{OMB}}$. $\mathcal{L}_{\text{OMB}}$ consistently outperforms standard cross-entropy $\mathcal{L}_{\text{CE}}$ (Table 11) on ImageNet-1k benchmark. By balancing the weight of ID/OOD sample weights, $\mathcal{L}_{\text{OMB}}$ effectively preserving the learning signal of minority-class samples. This yields more balanced updates and improved OOD detection robustness.

Effectiveness on OpenOOD Benchmark. We use four popular ID datasets CIFAR-10/100 [19], ImageNet-200/1K [10]. Following the OpenOOD benchmark [47], the OOD testing datasets are categorized into two groups: Near OOD and Far OOD. Specifically, for CIFAR-10/100 benchmarks, the Far OOD group includes MNIST [20], SVHN [33], Texture [9], Places365 [55], and the Near OOD group comprises CIFAR-100/10 and Tiny ImageNet-200 [10]. For ImageNet-200/1K, the Near OOD group includes SSB-hard [39], NINCO [3], and the Far OOD group comprises iNaturalist [16], Texture [9], and OpenImage-O [40].

The base detectors employed by OODD, AdaND, and our method TTL are all MCM, while AdaNeg utilizes the more powerful Neglabel as its base detector. As shown in Table 12 and Table 13 on the OpenOOD benchmark [47], under the far-out-of-distribution (far-OOD) setting, our method consistently achieves the best results. Under the near-OOD setting, when using CIFAR-10, CIFAR-100, and ImageNet-200 as in-distribution (ID) datasets, our method's performance rivals that of AdaNeg using external datasets and stronger base detectors. This demonstrates the effectiveness of our method across diverse experimental settings.

In the most challenging near OOD experimental setting using ImageNet-1k as the ID dataset, all methods exhibit limited performance. Since the base detector is fundamentally incapable of effectively distinguishing be-

Table 8. Complementarity to other OOD detectors with the ID dataset of ImageNet-1k. Green indicates an improvement, while red indicates the opposite.

Method	iNaturalist		SUN		Places		Texture		**Average**	
	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑
MCM	30.92	94.61	37.59	92.57	44.71	89.77	57.85	86.11	42.77	90.76
+ Ours	**0.42**	**99.87**	**7.18**	**98.45**	**15.86**	**96.22**	**26.39**	**94.60**	**12.46**	**97.29**
Improve	-30.50	+5.26	-30.41	+5.88	-28.85	+6.45	-31.46	+8.49	-30.31	+6.53
GL-MCM	15.09	96.72	29.08	93.41	37.07	90.37	58.94	83.11	35.04	90.90
+ Ours	**0.42**	**99.88**	**7.71**	**98.37**	**16.14**	**96.04**	**33.67**	**92.10**	**14.49**	**96.6**
Improve	-14.67	+3.16	-21.37	+5.67	-20.93	+5.67	-25.27	+8.99	-20.55	+5.70
Neglabel	2.00	99.47	20.95	95.47	36.48	91.56	45.00	90.02	26.10	94.13
+ Ours	**0.44**	**99.86**	**6.49**	**98.68**	**15.26**	**96.81**	**10.74**	**98.00**	**8.23**	**98.34**
Improve	-1.56	+0.39	-14.46	+3.21	-21.22	+5.25	-34.26	+7.98	-17.87	+4.21
LoCoOp	23.24	95.27	31.56	93.76	38.55	91.19	43.43	90.28	34.19	92.62
+ Ours	**0.22**	**99.93**	**4.92**	**98.88**	**13.91**	**96.51**	**16.12**	**96.44**	**8.79**	**97.94**
Improve	-23.02	+4.66	-26.64	+5.32	-24.64	+5.32	-27.31	+6.16	-25.40	+5.32
FA	13.37	96.80	28.83	93.12	30.30	92.54	30.50	92.66	25.75	93.78
+ Ours	**0.28**	**99.90**	**5.87**	**98.68**	**10.20**	**97.81**	**7.15**	**98.66**	**5.88**	**98.76**
Improve	-13.09	+3.10	-22.96	+5.56	-20.10	+5.27	-23.35	+6.00	-19.87	+4.98

Table 10. Performance comparison on the ImageNet-1k benchmark with ResNet50 backbone.

Method	iNaturalist		SUN		Places		Texture		**Average**	
	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑	FPR95↓	AUROC↑
MCM	32.92	93.73	47.73	90.43	60.67	85.71	61.65	85.24	50.74	88.78
Neglabel	2.60	99.29	22.62	95.05	47.71	90.0	42.85	89.80	28.95	93.53
FA	68.97	82.02	55.84	86.93	62.58	82.68	34.96	91.74	55.59	85.84
OODD	3.72	99.05	30.80	93.66	60.06	83.76	48.58	88.55	35.79	91.25
AdaND	14.92	96.96	40.47	91.19	44.54	88.26	34.22	90.98	33.53	91.84
AdaNeg	1.07	99.63	13.22	96.78	35.11	93.44	**25.61**	**94.67**	18.75	96.13
TTL(ours)	**0.73**	**99.75**	**9.39**	**98.16**	**18.49**	**95.25**	26.88	94.21	**13.87**	**96.84**

Table 11. Ablation study of the L_{OMB}.

Function	FPR95↓	AUROC↑
$\mathcal{L}_{\text{CE}}$	14.23	96.98
$\mathcal{L}_{\text{OMB}}$	**12.46**	**97.29**

tween ID and OOD samples, the OODD, AdaND, and TTL methods—which employ weaker base detectors—failed to achieve effective adaptation. Their performance was, to varying degrees, inferior to that of the base detector. Conversely, AdaNeg, utilizing a stronger base detector, still achieves performance improvements. For a fairer comparison, we uniformly replaced the base detectors of all adaptation methods with the stronger FA detector for testing. Results indicate that when the base detector possesses sufficient discrimination capability, our method achieves the best adaptation performance.

Analysis of Learned OOD Textual Knowledge. To interpret the textual knowledge learned and stored in the OKB

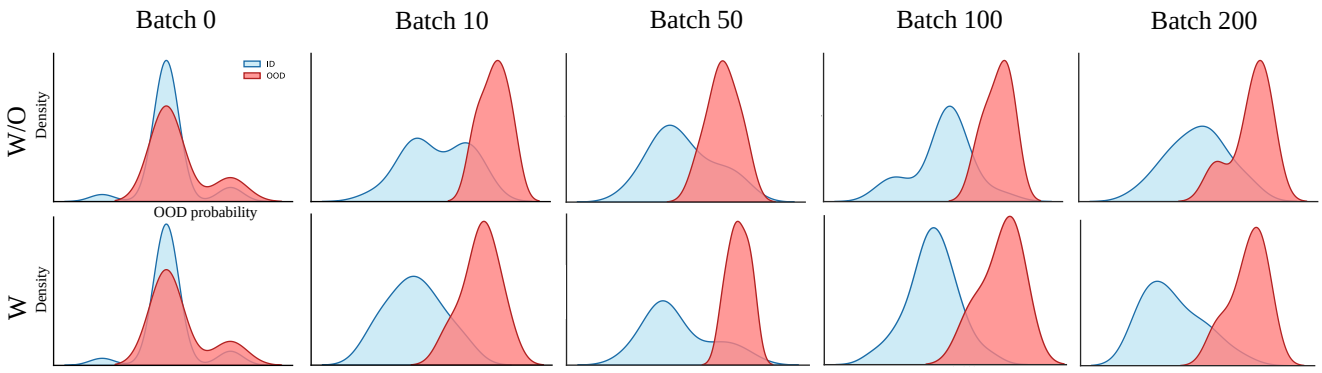

Figure 7. Effect of $\mathcal{L}_{\text{OKP}}$ on separating ID-boundary and OOD samples, 64 samples/batch. Top row: OOD probability density without $\mathcal{L}_{\text{OKP}}$; bottom row: OOD probability density with $\mathcal{L}_{\text{OKP}}$.

Table 12. Performance Comparison on OpenOOD benchmark across three ID Datasets. Lower FPR95 and higher AUROC are better. Best results are in **bold**, and the second-best results are underlined.

ID Dataset	Method	Extra Resources	Near OOD		Far OOD	
			FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑
CIFAR 10	MCM	×	35.00	91.00	12.57	96.77
	Neglabel	√	35.32	92.96	15.74	96.29
	OODD	×	48.61	89.19	14.11	96.70
	AdaND	×	35.06	90.97	2.02	99.40
	AdaNeg	√	32.38	**94.01**	7.31	98.28
	TTL	×	**30.25**	93.60	**1.07**	**99.75**
CIFAR 100	MCM	×	91.01	70.53	73.27	79.66
	Neglabel	√	77.54	71.90	59.66	79.84
	OODD	×	91.52	70.23	69.48	79.88
	AdaND	×	78.45	70.27	22.33	90.90
	AdaNeg	√	71.62	77.56	40.81	88.41
	TTL	×	**52.34**	**82.33**	**0.21**	**99.64**
ImageNet-200	MCM	×	63.66	83.66	17.97	96.13
	Neglabel	√	49.83	87.61	9.36	97.87
	OODD	×	48.39	83.82	14.82	96.13
	AdaND	×	53.77	83.76	14.85	96.00
	AdaNeg	√	**41.48**	**88.76**	9.79	98.05
	TTL	×	48.16	87.75	**7.89**	**98.50**

during test-time OOD detection, we draw inspiration from CoOp's analytical framework [56]. Specifically, we search the vocabulary of the VLM for tokens whose embeddings are closest to the learned vector, based on Euclidean distance. Note that CLIP [35] uses the BPE [36] representation for tokenization, and thus its vocabulary contains frequent subword units, such as "fore" (shared by words like "forest" and "foreland"). In our experiments, the learnable prompt consists of four tokens. Since these prompts vectors are optimized in a continuous space, we visualize their semantic meanings by listing the ten nearest words to each learned vector. The Table 15 shows results on the ImageNet-1k benchmark. The "None" entry indicates the default prompt initialized with "a photo of a", while two representative entries from the OKB are shown for illustration. On the *iNaturalist* dataset, the nearest tokens such as *plants*, *wildflowers*, *alpaca*, and *arugula* demonstrate that the model autonomously learns ecological and biological semantics from unseen natural images. On the *SUN* dataset, dominated by diverse outdoor scenes, the discovered tokens—*landscape*, *seascape*, *campsite*, and *cruise*—indicate that the model captures spatial composition and environ-

Table 13. Performance Comparison on OpenOOD benchmark with ImageNet-1k as ID dataset. Lower FPR95 and higher AUROC are better. Best results are in **bold**, and the second-best results are underlined.

ID Dataset	Method	Extra Resources	Near OOD		Far OOD	
			FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑
ImageNet-1k	MCM	×	84.17	69.22	44.39	90.61
	Neglabel	√	69.27	75.38	23.23	94.94
	LoCoOp	×	82.51	68.03	33.42	92.12
	IDLike	×	86.23	59.51	36.11	92.16
	LocalPrompt	×	77.91	73.44	28.6	93.71
	FA	×	69.21	77.97	25.78	93.67
	OODD	×	73.83	67.08	24.81	91.87
	OODD + FA	×	60.06	77.31	23.52	92.21
	AdaND	×	80.78	68.01	27.76	92.19
	AdaND + FA	×	75.40	74.28	23.05	93.53
	AdaNeg	√	67.35	76.01	20.9	95.44
	AdaNeg + FA	√	76.15	73.79	19.75	95.38
	TTL	×	89.44	56.3	19.88	95.59
	Ours + FA	×	**56.53**	**83.33**	**13.62**	**97.05**

Table 14. Effect of TTL with base detectors of different quality.

Method	FPR95↓	AUROC↑	Method	FPR95↓	AUROC↑
MCM	42.77	90.76	MaxLogit	67.13	83.92
+ Ours	**12.46**	**97.29**	**+ Ours**	**33.34**	**91.52**
MCM-Var	77.60	73.80	MCM-Entropy	**84.84**	**68.64**
Ours	**75.74**	**74.29**	**+ Ours**	85.19	58.6

mental context as OOD semantic cues. The emergence of verbs such as *enjoying*, *exploring*, and *commuting* further suggests that the model identifies human–scene interaction semantics absent from the ID training domain, demonstrating its capacity to extract high-level contextual shifts for better detection. In the *Places* dataset, which represents complex scene-level OOD distributions, the discovered tokens (e.g., *footprints*, *oceans*, *terracotta*, *surrealism*) reflect both material and stylistic semantics that differ from the in-distribution visual domain. By learning these distinctive scene-level attributes without supervision, the model effectively captures the semantic shift between familiar and unseen environments. For the texture-dominant *Texture* dataset, the nearest tokens—*formed*, *abstractart*, *homeitems*, *refurbi*, and *flavo*—show that the model adapts to surface- and pattern-related semantics. These learned tokens correspond to appearance-level variations such as color, gloss, and regularity, indicating that the model learns texture-sensitive knowledge that enhances its ability to detect OOD samples exhibiting distributional drift at the visual level. Overall, these results demonstrate that even without explicit retraining, our method can **discover OOD-relevant textual knowledge** from unlabeled test-time data streams. The learned semantics capture and describe distributional discrepancies between ID and OOD domains in an interpretable way, thereby improving both the model's adaptability and the robustness of OOD detection under distribution shift.

Robustness to Base Detector Quality. Pseudo-label noise and error reinforcement are an inherent challenge in test-time adaptation, yet our evaluation across varying signal-to-noise ratios (Table 14) demonstrates that TTL acts as a robust signal amplifier rather than relying on a strong oracle. Specifically, TTL yields consistent gains even when the base detector is weak (e.g., MCM-Var [29]), confirming that our purification mechanism successfully extracts structural information from noisy pseudo-labels. It is only when the base signal's signal approaches a near-random regime (e.g., MCM-Entropy [29]) that adaptation naturally diminishes. These results confirm that TTL does not require high-quality supervision but effectively improves performance provided the base detector offers a weak-but-structured separation signal.

Table 15. The nearest words for each of the 4 context tokens discovered on different OOD datasets. Each row corresponds to one context vector, and the ten nearest words are listed in order of similarity. Words that are in **bold** indicate those directly related to the OOD dataset's domain semantics. N/A indicates non-Latin tokens.

dataset	entry	token	1	2	3	4	5	6	7	8	9	10
None	1	1	a	an	0	1	6	7	5	4	2	3
		2	photo	photos	pic	7	3	2	4	5	0	1
		3	of	1	3	6	7	0	2	5	8	4
		4	a	an	0	1	6	7	5	4	2	3
iNaturalist	1	1	femme	**plants**	charms	**nutrients**	**wildflowers**	adilla	investments	things	resu	teammates
		2	selfies	alpaca	alpac	!'	N/A	affirm	dancers	N/A	footage	pictures
		3	of	opposite	N/A	behalf	whos	past	N/A	keyword	illustrious	N/A
		4	behindthe	bankno	chuckle	playo	residente	N/A	besides	expla	caled	starwar
	2	1	**arugula**	adilla	charms	teammates	femme	dementi	hearts	teammate	cosplayer	**wildflowers**
		2	alpaca	airdrop	alpac	extras	N/A	supportindiefilm	confirms	broadcasts	whilst	wholesale
		3	N/A	N/A	N/A	N/A	N/A	.!!	mediation	N/A	past	++
		4	chuckle	N/A	nrl	behindthe	bankno	playo	condomin	whist	**fore**	isai
SUN	1	1	mustache	**stay**	a	attire	braces	**cruise**	signature	autograph	scones	**azure**
		2	photo	**campsite**	**landscape**	**seascape**	activation	**chillout**	**landscapes**	letstalk	!'	**landscapephotography**
		3	of	**enjoying**	**enjoys**	actions	aton	**enjoyed**	installing	activation	calling	**exploring**
		4	weal	**evapor**	metaph	qur	poon	mpg	meand	syl	refr	**wooden**
	2	1	mustache	piday	commissioning	autograph	**stay**	date	**cruise**	attire	signature	**typography**
		2	photo	**campsite**	offseason	lineups	gameplay	commitment	**naturephotography**	**commuting**	**chillout**	letstalk
		3	**enjoys**	**enjoying**	**exploring**	**enjoyed**	activation	installing	applied	medallion	**explores**	**enjoy**
		4	syl	qur	earn	wfa	metaph	ellu	edou	prosp	margare	benevol
places	1	1	a	**roma**	**footprints**	N/A	forza	N/A	an	N/A	iftar	etres
		2	photo	-'	addresses	painting	[@	N/A	addressing	semester	**photobook**	(@
		3	**seine**	movements	sem	functionality	optimize	**homeowner**	adn	**morn**	prompted	of
		4	unex	**terracotta**	unemployment	invent	economists	jailbreak	econ	exclu	dispro	ameli
	2	1	**ulster**	**slovenia**	**cryst**	**oceans**	N/A	outline	**surrealism**	rgv	suffra	tracklist
		2	hdr	—	N/A	redesigned	installations	N/A	printable	':	——	**styled**
		3	wkend	**contempor**	**morn**	beginner	lovehim	arrog	seine	consult	wks	societal
		4	unex	bluebird	spie	dispro	**statu**	baff	pinst	economists	jailbreak	econ
Texture	1	1	xboxone	N/A	N/A	N/A	N/A	entirety	gianni	trojans	hardware	gmail
		2	olympians	ballet	paralympic	sawards	warhol	loool	N/A	fineartamerica	soviet	N/A
		3	**kaleido**	**formed**	**abstractart**	rof	elector	atomic	houghton	appointed	sitcom	-$
		4	exem	womancrush	**homeitems**	**viny**	**plicity**	**refurbi**	**ergon**	**flavo**	rahulg	fianc
	2	1	a	N/A	every	etres	accessing	momento	ramapho	equip	hahahha	websites
		2	photo	**uniforms**	twitart	N/A	followparty	N/A	mortgages	railwayana	arrested	N/A
		3	of	on	from	to	and	of	about	with	in	&
		4	N/A	recap	ksu	N/A	N/A	ksa	**sculpt**	**pendants**	dessert	**nailart**