APPLIANCE SIGNATURE SPACE WITH ONE FOR ALL NILM

Anonymous authors
Paper under double-blind review

ABSTRACT

Non-intrusive load monitoring (NILM) estimates the power drawn by individual appliances from a single household meter. Deep NILM models are usually trained one per appliance, so a smart meter that tracks many appliances must store many networks. We study one-for-all NILM (OFAN), in which a single transformer disaggregates whichever appliance it is conditioned on through a token appended to its input sequence, taken either from a learned embedding or from an encoder applied to a representative activation waveform. We add this conditioning to two models: BERT4NILM and NILMformer, and evaluate it on UK-DALE and REFIT over five seeds with paired confidence intervals. For five appliances, one conditioned BERT4NILM model has 1.9M parameters compared to 9.7M for five separate models, and each additional appliance adds a single 256-dimensional embedding. The cost of sharing one model depends on the backbone and the dataset: on UK-DALE, pooled mean absolute error (MAE) rises by 12.8% for NILMformer and by 24.6% for BERT4NILM, while on the REFIT datasets, a single BERT4NILM model lowers MAE by 5.6%. Conditioning on activation waveforms is less accurate than learned embeddings on UK-DALE; however, it transfers what it has learned: applied to the other dataset without retraining and conditioned on the appliance signature it learned in training, a model reaches a mean F1 of 0.43 over 101 sub-metered channels, above a random detector on every channel and above an always-on detector on 88 of them, and 0.20 F1 higher than the same model conditioned on a query built in the target home (95% CI $[0.17, 0.23]$); appliance types absent from training are not detected. Finally, we read the learned embedding table as a proxy for Hart's appliance signature space and find structure in it that only training can explain: in all four model–dataset pairs, the kettle and microwave, the two appliances with similar short high-power cycles, form the closest pair, and in three of them, they trade accuracy under joint training.

1 INTRODUCTION

Global electricity demand has continuously grown over the past two decades and is projected to accelerate further as electrification of transport, heating, and industry expands. Alongside this growth, utilities and regulators have introduced a wide range of demand-side incentives including time-of-use tariffs, dynamic pricing schemes, and demand response programs to shift consumer usage patterns away from peak periods and toward times of renewable surplus (You et al., 2023). These programs share a common requirement: granular visibility into what electricity is being consumed, when, and by which devices.

The widespread deployment of smart meters across residential sectors in the United Kingdom and Europe has created the metering infrastructure necessary to support such visibility. In Great Britain, for example, 71% of the meters in homes and small businesses were smart or advanced meters by the end of 2025, enabling near-real-time whole-home consumption data to be collected at the meter (Department for Energy Security and Net Zero, 2026). Feedback on consumption has been shown to help households reduce their electricity use (Gans et al., 2013; Kim et al., 2024), and this data could extend such feedback to the appliance level, telling householders which devices account for their consumption. Yet whole-home aggregate data, on its own, cannot answer this question.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Non-Intrusive Load Monitoring (NILM), formalized by Hart (1992), addresses this gap. NILM algorithms decompose the aggregate power signal measured at a single metering point, the smart meter, into the constituent consumption signatures of individual household appliances, recovering per-device power traces without the need for dedicated sub-metering hardware at each appliance (Hart, 1992; Zoha et al., 2012). Unlike intrusive monitoring, which requires a sensor at every outlet, NILM operates entirely through signal processing and machine learning applied to the whole-home power trace, making it economically viable at population scale. The progressive availability of large, publicly annotated household datasets, including the Reference Energy Disaggregation Dataset (REDD) (Kolter & Johnson, 2011), the UK Domestic Appliance-Level Electricity dataset (UK-DALE) (Kelly & Knottenbelt, 2015a) and the REFIT Electrical Load Measurements dataset (Murray et al., 2017), has provided the empirical foundation upon which the modern NILM literature has been built, driving steady progress in disaggregation accuracy over the past decade.

Performing disaggregation natively at the edge (such as on the smart meter processor itself) is highly desirable; it keeps sensitive household activity data local and reduces the bandwidth and server storage needed to stream high-rate power data to a remote database. Yet, this deployment context introduces stringent architectural boundaries. Smart meters are highly constrained embedded devices with extremely strict memory allocations, limited compute cycles, and minimal persistent storage capacity, so the models they run must be small. Metering platforms differ widely: a metrology microcontroller cannot run a transformer at all, whereas meters and home gateways built around an application processor can. Our target is the second class, and what we aim to reduce is the number of models that must be stored and maintained, not the computation of a single inference.

Early milestones established by Kelly & Knottenbelt (2015b) successfully demonstrated the efficacy of Denoising Autoencoders (DAEs), Long Short-Term Memory (LSTM) recurrent networks, and convolutional architectures in mapping complex aggregate windows to individual loads. However, these models operate under a rigid system setup: execution of specialized networks optimized exclusively for a single target device class. This dominant methodology in the literature commits the full capacity of a trained neural network graph to a single appliance's signature: an approach we call *one-for-one NILM (OFON)*. Recent architectures in this line include BERT4NILM (Yue et al., 2020), which adapts the bidirectional context-capturing power of language transformers with a composite loss function penalizing state errors, and NILMFormer (Petralia et al., 2025), which introduced subsequence stationarization and Time-Aware Relational Positional Encoding (TimeRPE) to mitigate signal non-stationarity for large utility networks like Électricité De France (EDF).

One-for-one NILM training gives accurate disaggregation for each targeted load, but it does not scale with the number of appliances. Under this paradigm, the storage and memory footprint of the deployed firmware scales linearly ($O(|\mathbb{A}|)$) with $\mathbb{A}$ the set of appliances being tracked. House 1 of UK-DALE alone has more than 50 sub-metered channels, and tracking 50 appliances with separate BERT4NILM models would take 97M parameters (388 MB in 32-bit floats), compared to 1.95M for one model conditioned on the appliance. The savings are in stored parameters; disaggregating a given appliance still costs one forward pass of the backbone.

To mitigate this computational redundancy, alternative research has explored multi-appliance shared backbones. These configurations force early structural layers to learn global aggregate abstractions, which are subsequently split into separate streams by either passing data to a predetermined appliance matrix or attaching individual, appliance-specific sub-network output heads, such as the compact one-to-many architecture proposed in MA-CNN (Li et al., 2023). While this shares a multi-task constraint across known device classes, it remains structurally rigid. [1]. These networks can only disaggregate the specific devices included in the original training matrix. Incorporating a newly released home appliance requires changing the network topology, adding output layers and retraining on an expanded dataset.

To resolve this rigidity, intermediate strategies have turned to transfer learning and signal conditioning. D'Incecco et al. (2019) demonstrated meaningful cross-appliance feature transferability using Appliance Transfer Learning (ATL) and Cross-Domain Transfer Learning (CTL) extensions on

[1]This rigidity is also why a multi-head model is not a controlled baseline here: it requires simultaneous targets for every appliance at each timestep, while our appliance channels are metered in different houses, so training one would mean either zero-filling absent appliances or filtering the training houses. We therefore compare against QueryNILM (Yang et al., 2025), a query-based model that conditions on the appliance in the same way OFAN does (§4)

sequence-to-point topologies. Similarly, Garcia-Marrero et al. (2026) introduced RefQuery, which learns lightweight appliance embeddings during a brief adaptation phase while freezing the central backbone to support plug-and-play expansion.

Furthermore, García et al. (2025) bridged the multi-head gap by enhancing a Fully Convolutional Denoising AutoEncoder (FCNdAE) with Feature-wise Linear Modulation (FiLM) layers (Perez et al., 2018). This design used a one-hot indicator vector to condition a single model's execution path, demonstrating that a multi-target task could be handled without expanding output heads.

Concurrently within the broader machine learning landscape, conditional generative architectures and few-shot learning systems have established methods for managing open-ended classes using dense vector conditioning. In natural language processing, transformers commonly receive extra information at the input rather than through separate branches: BERT (Devlin et al., 2018) adds token, position and segment embeddings, and prompt tuning prepends learned tokens that condition a frozen language model (Lester et al., 2021).

This formulation is echoed in QueryNILM (Yang et al., 2025), which frames energy disaggregation as a query task. By feeding a whole-home trace together with a target appliance query vector into a stack of dilated convolutional and residual layers, QueryNILM isolates individual device traces from a single model graph, and studies two signaling schemes: one-hot encoded identifier vectors and appliance activation vectors. This establishes the form of the conditioning signal as a design variable in its own right rather than an implementation detail and raises a question it does not set out to answer: how such a query behaves once it is injected into a backbone whose receptive field is global rather than convolutional, whether the choice of signaling scheme interacts with that backbone, and whether a learned query outperforms one derived from a recorded waveform.

1.1 Proposed Work: One For All NILM (OFAN)

To address these limitations, we introduce **OFAN** (**O**ne **F**or **A**ll **N**ILM), a conditional framework that appends an appliance embedding to the input sequence of a transformer backbone. The embedding is either looked up from a table indexed by appliance or computed by an encoder from a representative activation waveform, and it is appended to the power features and positional information before the attention layers. Every attention layer can therefore read the appliance token, so one model disaggregates each appliance it is conditioned on, and the number of stored models no longer grows with the number of appliances. We add this conditioning to BERT4NILM and NILMformer, train on UK-DALE and REFIT with five seeds, and compare configurations with paired confidence intervals over seeds.

The primary technical contributions achieved in this work are summarized as follows:

- **Architectural decoupling via learned appliance embeddings:** A conditioning mechanism appends a trainable appliance embedding to the input of a transformer backbone, so that a single model disaggregates the appliance it is conditioned on without structural modification.
- **Scalable unified disaggregation framework:** OFAN reduces model storage relative to one-for-one approaches from $O(|\mathbb{A}|)$ independent models to a single shared model with a compact embedding table of size $|\mathbb{A}| \times d$, where d is the embedding dimensionality, or a shared appliance encoder (§2.2); for five appliances this reduces BERT4NILM from 9.7M to 1.9M parameters.
- **Embedding analysis:** We use the learned embedding table as a proxy for the appliance signature space of Hart (1992) and analyze its pairwise cosine similarities against random and seed-agreement baselines; the kettle and microwave form the most similar pair in all four backbone–dataset pairs, and in three of them these two appliances trade accuracy under joint training.
- **Controlled evaluation on standard benchmarks:** On four model–dataset pairs (UK-DALE and REFIT; BERT4NILM and NILMformer) we separate the effect of the token, of sharing one model and of the query type, with paired confidence intervals over five seeds: the token alone changes no appliance's MAE significantly, and sharing one model raises pooled MAE by 12.8–24.6% on UK-DALE and lowers it by 5.6% for BERT4NILM on

REFIT. We also compare against QueryNILM (Yang et al., 2025) under the same protocol, query type for query type, with neither model dominating (§4).

- **Transfer to a new dataset without retraining:** Conditioned on the appliance signature it learned in training, a model detects the appliance types it knows in 101 sub-metered channels of the other dataset (mean F1 0.43, above chance on every channel), which a query built in the target home does not match; appliance types absent from training are not detected (Appendix C).

2 METHODOLOGY

2.1 PROBLEM FORMULATION

Let the aggregate power signal measured at a single point of consumption be denoted as a sequence $\boldsymbol{x} = \{x_1, x_2, \ldots, x_w\} \in \mathbb{R}^w$, where W represents the temporal sequence (window) length. The objective of Non-Intrusive Load Monitoring (NILM) is to reconstruct the individual power consumption sequence of a specific target appliance a from the compound sequence $\boldsymbol{x}$.

In the conventional OFON paradigm, a distinct neural network model f_{θ_a} must be trained independently for each appliance category $a \in \mathbb{A}$:

$$\hat{\boldsymbol{y}}_a = f_{\theta_a}(\mathbf{x}) \tag{1}$$

This baseline approach scales poorly in real-world deployments, as it necessitates the storage and maintenance of $|\mathbb{A}|$ independent parameter sets $\{\theta_a\}$.

Conversely, the proposed One For All NILM (OFAN) framework reformulates disaggregation as a unified, conditional regression task. We employ a single shared neural network architecture f_θ that is parameterized by a global weights vector θ and conditioned on a learned continuous appliance embedding $\boldsymbol{e}_a \in \mathbb{R}^d$:

$$\hat{\mathbf{y}}_a = f_\theta(\mathbf{x}, \mathbf{e}_a) \tag{2}$$

The set of embedding vectors $\{\boldsymbol{e}_a : a \in \mathbb{A}\}$ is optimization-driven and learned jointly end-to-end alongside the global model parameters θ during the training phase. During inference, the target appliance is disaggregated simply by querying the unified model with the corresponding embedding $\boldsymbol{e}_a$, so the number of stored models stays constant as $|\mathbb{A}|$ grows; only the $|\mathbb{A}| \times d$ embedding table grows with it.

2.2 APPLIANCE EMBEDDING

A central design question in OFAN is how to represent the target appliance to the model. Unlike previous OFON approaches where the model implicitly encodes appliance identity through its weights, OFAN requires an explicit conditioning signal. We investigate two approaches.

Index Embedding. Each appliance $a \in \mathbb{A}$ is assigned a trainable dense vector $\boldsymbol{e}_a$ stored in a lookup table of size $|\mathbb{A}| \times d$. The table is randomly initialized and optimized jointly with the backbone during training. At inference, the appropriate row is retrieved by appliance index. This adds a negligible number of parameters ($|\mathbb{A}| \times d \ll |\theta|$) and requires no preprocessing: for five appliances and $d = 256$ the table holds 1,280 parameters, so one BERT4NILM model has 1.94M parameters against 9.69M for five per-appliance models (NILMformer: 2.71M against 13.52M). The limitation is that the embedding carries no information about the appliance prior to training: discrimination must emerge entirely from the disaggregation objective. However, the lack of this inductive bias gives the model the freedom to explore the embedding space for a flexible appliance representation.

Activation Embedding. Motivated by Yang et al. (2025), we additionally explore deriving the embedding from a representative appliance activation, a recorded power trace covering a complete operational cycle. A set of activations is collected for each appliance, each activation is transformed to the frequency domain via the FFT amplitude spectrum, and the resulting feature vectors are clustered by applying K-means ($k = 5$) to identify the dominant operating mode. The activation nearest

to the centroid of the largest cluster is retained as the representative. The full procedure is given in Figure 3.

At training time, the representative activation is passed through a learned *appliance_encoder*, a five-layer dilated convolutional network, to produce $\boldsymbol{e}_a$, trained jointly with the backbone. The encoder adds 245k parameters, shared by all appliances. This approach is designed to encode the spectral and temporal characteristics of the appliance's power signature into the embedding, and allows the model to be conditioned at test time on a support activation from a home or appliance not seen in training, which we evaluate in Section 4 and Appendix C.

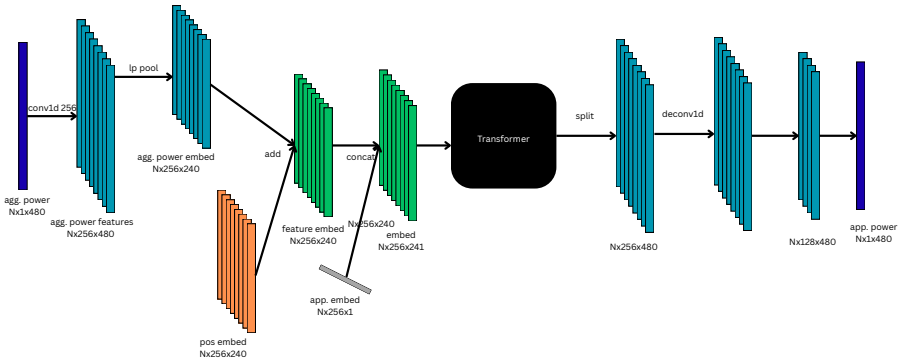


Figure 1: OFAN with the BERT4NILM backbone: the appliance embedding is concatenated to the power embedding sequence before the transformer and removed after it

2.3 Backbone Instantiations

OFAN is backbone-agnostic: the conditioning module described in §2.2 requires only that the backbone operates over a sequence of embeddings before its core attention stack, so that the appliance token can be appended at that position. We instantiate OFAN on two transformer backbones that represent complementary points in the design space.

BERT4NILM. The aggregate power window of length $W = 480$ (sampling period 6 s on UK-DALE, 8 s on REFIT) is first mapped to an embedding sequence by a 1-D convolutional feature embedding followed by L^2 pooling with stride 2, halving the sequence to length $W/2 = 240$. Learnable positional embeddings are added element-wise. In the OFAN variant, the appliance embedding $\mathbf{e}_a$ is appended as position $W/2 + 1$, expanding the sequence seen by the transformer to length 241. After the two-layer bidirectional attention stack ($d = 256$, $H = 2$ heads, dropout $p = 0.1$), the appended token is removed and the sequence is restored to length W via a transposed convolution before the output head. The model is trained with BERT-style masking: 25% of input positions are masked before the forward pass (80% of them set to -1, 10% given Gaussian noise and 10% left unchanged), the MSE loss is computed over the masked positions, and this masking scheme is preserved in the OFAN variant.

NILMformer. The aggregate power window of length $W = 256$ at 60 s resolution is processed by a dilated feature embedding (a bank of four parallel 1-D convolutional branches with dilation rates $\{1, 2, 4, 8\}$) which captures local power dynamics at multiple temporal scales and projects to $\frac{3}{4}d = 192$ dimensions. Timestamp features (month, day-of-week, hour, minute), encoded as sinusoidal pairs, are projected by a pointwise convolution to the remaining $\frac{1}{4}d = 64$ dimensions and concatenated, yielding a full $d = 256$ representation at each position. To address the non-stationarity of aggregate power signals, a stationarisation token (the projected mean and standard deviation of the input window) is appended to the sequence before the transformer. In the OFAN variant, the appliance token $\mathbf{e}_a$ is further appended after the stationarisation token; the three-layer attention stack ($d = 256$, $H = 8$ heads, dropout $p = 0.2$, diagonal attention mask) therefore attends jointly over power positions, the stationarisation context, and the appliance query. After the transformer, both the appliance token and the stationarisation token are removed, and the power sequence is decoded by a pointwise `Conv1d` output head. NILMformer is trained as a pure regression model: no token masking is applied, and the loss is MSE over all positions.

The two instantiations share an identical conditioning interface: the appliance token is always the final element appended to the pre-transformer sequence and the removed afterwards, Fig. 1, while differing in sampling rate, sequence length, and positional structure. Because the two backbones also follow different training protocols, we compare configurations within each backbone rather than across backbones.

3 Experiments Setup

3.1 Datasets

UK-DALE. The UK Domestic Appliance-Level Electricity dataset (Kelly & Knottenbelt, 2015a) provides whole-home and per-appliance power measurements from five UK households. We evaluate on five appliance categories: kettle, fridge, dishwasher, microwave, and washing machine. Training data is drawn from Houses 1, 3, and 5, with house availability per appliance determined by metering coverage (e.g. fridge and dishwasher are available in Houses 1 and 5 only). House 2 is used only for testing; the validation data of each backbone are described in Appendix A.1.

REFIT. The REFIT Electrical Load Measurements dataset (Murray et al., 2017) provides two-year longitudinal measurements from 20 UK households. We evaluate on four appliance categories: kettle, dishwasher, microwave, and washing machine. Fridge is excluded in accordance with Murray et al. (2017). Training houses vary by appliance based on metering availability (9–12 houses per appliance). House 6 is used for validation and Houses 2 and 3 for testing, consistently across all appliances.

3.2 Training Configurations

To study the impact of the proposed conditioning mechanism, we evaluate four experimental configurations across all models and datasets.

- **OFON-Individual:** The standard backbone without appliance embedding conditioning, trained independently per appliance. This is the one-for-one baseline.
- **OFAN-Individual:** A separate model trained per appliance, but each model retains the full OFAN conditioning architecture with a single-row embedding table. This isolates the effect of the embedding mechanism from joint multi-appliance training.
- **OFAN-Combined:** A single model trained jointly across all appliances simultaneously. Training batches are randomly drawn from all appliance channels, and the embedding table contains one row per appliance. This is the primary deployment target of OFAN, reducing model storage from $\mathcal{O}(|\mathbb{A}| \times |f_{\theta_a}|)$ independent networks to a single model with an embedding table of size $|\mathbb{A}| \times d$. Joint training across appliances with very different power profiles (a kettle at $\sim$2000 W, a fridge at $\sim$50 W) introduces competing gradient signals, and some accuracy tradeoff relative to individual training is expected.
- **OFAN-Act-Combined:** A single model trained jointly across all appliances, using an activation-waveform embedding in place of a learned index vector. A representative power activation for each appliance is extracted from training data as described in §2.2. This waveform is encoded at training time by a learned five-layer dilated CNN (the *appliance encoder*), producing $\boldsymbol{e}_a$ jointly trained with the backbone. Unlike the index embedding, the conditioning vector is computed from the appliance's recorded power signature, so a new waveform can be supplied at test time without adding parameters.

We detail the full preprocessing setup, Hyperparameters and Training Protocol and Evaluation Metrics in the appendix.

4 Results

Table 1 reports test MAE for every configuration; per-appliance standard deviations, F1 and MRE are in Tables 3–6. We call a difference an improvement or a degradation only when the 95% confidence interval of the paired difference over the five seeds excludes zero. Dataset-level statements

Table 1: Test MAE (W) per appliance, mean over five seeds. Bold: lowest MAE among the four configurations of a backbone. QueryNILM (Yang et al., 2025) is included as an external baseline rather than a configuration of our framework, so its columns are not bolded; it uses its own architecture and learning rate with the BERT4NILM data and training protocol. †: the 95% confidence interval of the paired difference from OFON-Indiv excludes zero (in either direction); for QueryNILM the difference is taken from the BERT4NILM OFON-Indiv models, which share its protocol. Pooled Δ: relative MAE change from OFON-Indiv averaged over appliances, with the interval taken over the five per-seed averages. Standard deviations, F1 and MRE are in Tables 3–6 and Table 10.

	BERT4NILM				NILMformer				QueryNILM	
	OFON Indiv	OFAN Indiv	OFAN Comb	OFAN Act-Comb	OFON Indiv	OFAN Indiv	OFAN Comb	OFAN Act-Comb	One-hot query	Act. query
UK-DALE										
Kettle	9.05	**8.75**	18.14†	19.54†	14.12	15.54	**10.66**†	23.74†	18.11†	17.92†
Fridge	19.39	19.11	**18.81**	21.54	**19.64**	19.74	28.42†	32.26†	19.57	22.28†
Dishwasher	31.70	**29.11**	34.72	35.09	**16.51**	20.12	19.00	19.38	32.69	33.31
Microwave	**6.51**	6.64	6.62	7.71†	7.60	**7.38**	8.52†	11.94†	10.04†	10.65†
Wash. Machine	4.93	**4.80**	5.26	7.56†	**12.36**	13.07	13.11	14.41	4.01†	5.30
Pooled Δ (%)	–	-1.9	$+24.6$†	$+42.3$†	–	$+9.8$	$+12.8$†	$+48.6$†	$+29.1$†	$+39.1$†
REFIT										
Kettle	9.63	**9.62**	10.60†	10.43†	**13.58**	13.96	17.68	15.11	9.83	25.07†
Dishwasher	30.62	34.50	**27.16**	29.43	55.97	54.45	57.54	**43.49**†	25.54†	31.18
Microwave	6.15	6.01	**4.93**†	5.22†	3.85	**3.64**	8.85	8.98†	4.74†	19.99
Wash. Machine	16.48	16.56	**16.19**	17.63	25.97	26.99	31.30	**24.18**	13.44†	22.99
Pooled Δ (%)	–	$+2.8$	-5.6†	-0.8	–	0.0	$+43.5$	$+31.3$†	-13.9†	$+107.9$

average the relative MAE difference over appliances within each seed and test over the five seeds (Appendix A.2).

Conditioning token in single-appliance models. OFAN-Indiv differs from OFON-Indiv only by the appended token, and none of the 18 per-appliance MAE differences between them is significant. The pooled changes across seeds lie between -1.9% and $+9.8\%$, all with intervals that contain zero. This is expected: with a single-row embedding table the token is identical for every input, so it cannot carry appliance identity. It acts as one extra token, which BERT4NILM trains while NILMformer keeps close to its initialization (Section 5). Its significant effects appear only for BERT4NILM, and mostly in detection. On UK-DALE, pooled F1 rises by 0.034 (95% CI $[0.012, 0.056]$); the washing machine is the only appliance whose F1 gain is significant on its own ($0.71 \rightarrow 0.81$), while its On-MAE rises ($220 \rightarrow 266$ W). On REFIT the pooled F1 change is not significant, but the dishwasher F1 falls significantly ($0.54 \rightarrow 0.48$).

Cost of joint training. The cost of replacing the per-appliance models with one OFAN-Comb model depends on the backbone and the dataset. For BERT4NILM on UK-DALE, pooled MAE rises by 24.6%, mostly because of the kettle: its MAE doubles ($9.05 \rightarrow 18.14$ W) and its F1 falls from 0.91 to 0.63, while the other four appliances show no significant MAE change. On REFIT, joint training lowers pooled MAE by 5.6% (CI $[-10.8, -0.4]\%$); the microwave improves (-19.9%) and the kettle again degrades ($+10.1\%$). The microwave also gains on UK-DALE, where its F1 rises from 0.17 to 0.41 (CI of the difference $[0.11, 0.39]$). On both datasets, then, the kettle loses accuracy under joint BERT4NILM training and the microwave gains. The two appliances have similar short, high-power activations, and their learned embeddings form the most similar pair (Section 5), which suggests that the shared model partly confuses them.

NILMformer behaves differently. On UK-DALE joint training improves the kettle (-24.5%) but degrades the fridge ($+44.7\%$) and the microwave ($+12.1\%$), for a pooled change of $+12.8\%$. On REFIT the pooled change of $+43.5\%$ is not significant, and neither is any single appliance, because the seed variance of the joint model is large (standard deviations up to 13.7 W, Table 6). The two backbones follow the training protocols of their original papers (Table 2), so differences between them reflect architecture and training protocol together.

Index versus activation embedding. Conditioning on an encoded activation waveform (OFAN-Act-Comb) rather than a learned index vector (OFAN-Comb) raises pooled MAE on UK-DALE for both backbones, by 18.1% for BERT4NILM and 40.2% for NILMformer; on REFIT the differences ($+5.8\%$ and $+1.0\%$) are not significant. Against the per-appliance baseline, OFAN-Act-Comb is significantly worse in three of the four backbone–dataset pairs ($+42.3\%$ and $+48.6\%$ on UK-DALE, $+31.3\%$ for NILMformer on REFIT).

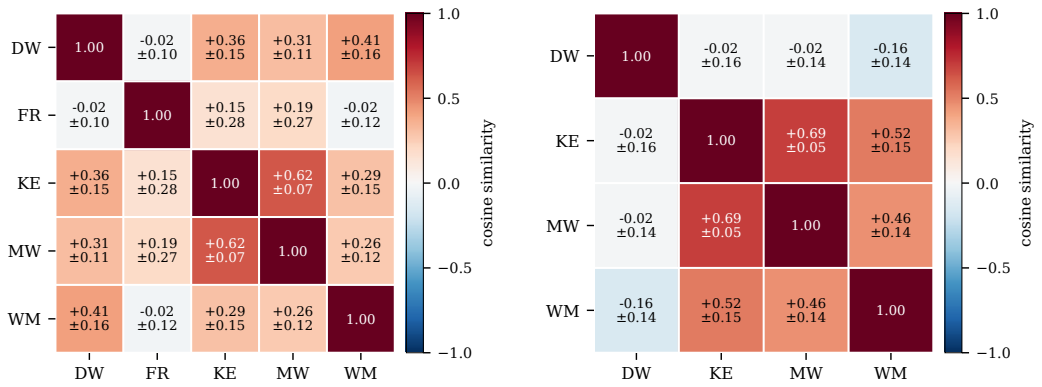
Comparison with QueryNILM. QueryNILM (Yang et al., 2025) conditions a dilated convolutional stack on a query injected into every residual block, rather than a transformer on an appended token. We train it with its own architecture and learning rate under our data and training protocol, on five seeds (Appendix D), and compare each query type against the OFAN configuration that uses the same signaling scheme. With index queries the two are indistinguishable on UK-DALE (pooled MAE $+6.6\%$, $[-4.9, +18.1]$), while on REFIT QueryNILM is more accurate than OFAN-Comb by 8.2% $[-10.0, -6.4]$. With activation queries they are again indistinguishable on UK-DALE ($+1.5\%$, $[-13.0, +16.0]$), but on REFIT QueryNILM's seed variance is extreme (microwave 19.99 ± 12.98 W against 5.22 ± 0.59 W for OFAN-Act-Comb), so its pooled degradation of $+121.7\%$ spans zero. A global receptive field is therefore not required at this scale, but the encoder that turns a waveform into a query matters: ours is stable across seeds where QueryNILM's is not. Where QueryNILM loses accuracy it loses it on the same appliances we do, reaching the same kettle MAE our shared model degrades to (18.11 against 18.14 W), which suggests the cost of sharing follows from the appliances rather than from the conditioning mechanism.

Transfer to a new dataset without retraining. We apply the BERT4NILM OFAN-Act-Comb models to the dataset they were not trained on, conditioned either on the appliance signature learned in training (zero-shot inference), which needs no data from the target home, or on a query built from the earliest 10% of the target channel's own record (Appendix C). Over 101 sub-metered channels the training signature gives a mean F1 of 0.43, above a random detector with the model's on-rate on every channel and above an always-on detector on 88 of them; the REFIT-trained model reaches 0.76 on the UK-DALE kettle and 0.63 on its dishwasher and washing machine. The target home's own query is worse by 0.199 F1 pooled over seeds (95% CI $[0.172, 0.227]$) and wins on only 13 of the 101 channels, so a single 10% segment describes an appliance less well than the signature the model was trained with. Transfer is weakest where the appliance is close to always on: on the REFIT fridge the model reaches 0.49 against an always-on baseline of 0.48, and only 21 of those 34 channels beat it. Appliance types absent from training are not detected: median F1 is 0.02–0.05, and only 10 of 58 such UK-DALE channels for the UK-DALE-trained model and 5 of 61 for the REFIT-trained model beat both references, none above 0.14.

5 Learned Appliance Embeddings

We examine the index-embedding tables of the OFAN-Comb models, whose structure can only come from the disaggregation objective. Hart (1992) described appliances as clusters in a measured signature space of real and reactive power changes; we use the embedding table as a learned proxy for this space. For each seed we compute the cosine similarity between every pair of appliance embeddings and report the mean and standard deviation over the five seeds (Figure 2; NILMformer in Figure 6). We compare these values with two references. First, the cosine between two independent random vectors in $d = 256$ dimensions has a standard deviation of $1/\sqrt{d} = 0.0625$, so a similarity beyond ± 0.12 is unlikely to arise by chance. Second, both backbones initialize the table from a truncated normal distribution with a standard deviation of 0.02 (norm ≈ 0.28), drawn separately for each seed, so a similarity pattern that repeats across seeds arises from training rather than from the initialization.

BERT4NILM. The embeddings grow to norms of 0.73–1.17. The kettle and microwave form the most similar pair on both datasets (0.62 ± 0.07 on UK-DALE, 0.69 ± 0.05 on REFIT). The two loads differ electrically, a resistive element against a magnetron with its high-voltage supply, but both draw 1–3 kW in short, near-rectangular activations, which suggests that the embedding reflects the shape of the power trace more than the load type. Other pairs are less stable: the dishwasher–washing machine similarity is 0.41 ± 0.16 on UK-DALE but -0.16 ± 0.14 on REFIT, and the fridge's similarities to the kettle and microwave vary by ± 0.28 across seeds. The correlation between the



(a) UK-DALE (b) REFIT

Figure 2: Pairwise cosine similarity between the BERT4NILM OFAN-Comb appliance embeddings, mean ± std over five seeds. PCA projections are in Figure 5 (appendix).

off-diagonal cosines of two seeds averages 0.94 on REFIT and 0.54 on UK-DALE. The kettle and microwave are also the pair whose accuracy trades off under joint training (Section 4).

NILMformer. From the same initial distribution, NILMformer's embeddings move less (norms 0.33–0.53) and form a weaker structure that agrees less across seeds (mean correlation 0.25 on UK-DALE, 0.32 on REFIT). In the individual configuration, where the per-appliance models of a seed all start from the same vector, NILMformer's tokens stay close to it (pairwise cosine 0.84–0.88 after training, against 0.32–0.36 for BERT4NILM). The kettle–microwave pair is still the most similar on both datasets (0.33 ± 0.08 and 0.27 ± 0.13), above the range expected for random vectors. The backbones differ in sampling rate (6–8 s against 60 s, at which a kettle activation spans only a few samples), architecture and training protocol (Table 2), so we cannot definitely attribute the weaker structure to one of these factors.

6 Conclusion

We studied one-for-all NILM, in which a single transformer disaggregates the appliance whose token is appended to its input. A model with a small embedding table replaces one model per appliance, at a fifth of the parameters for five appliances. The token alone does not change the MAE of a single-appliance model. Sharing one model raises pooled MAE by 12.8–24.6% on UK-DALE, with most of the cost on a few appliances, and lowers it by 5.6% for BERT4NILM on REFIT. Used as a proxy for Hart's appliance signature space, the learned embeddings place the kettle and microwave closest together for both backbones and datasets, and in three of the four settings these two appliances trade accuracy under joint training. Activation queries are less accurate than learned embeddings on UK-DALE, but they transfer: conditioned on the signature it learned in training, a model detects the appliance types it knows in the other dataset without retraining, at a mean F1 of 0.43 over 101 channels, which a query built in the target home does not match. Appliance types absent from training are not detected.

The evaluation has limits. It covers two datasets with four or five appliances each. Storage is counted in parameters and we do not measure memory on meter hardware, so whether a conditioned transformer of this size fits a given metering platform is untested. The two backbones keep the training protocols of their original papers, so differences between them mix architecture and protocol. Future work is to train the index embedding on more appliance types and to study different training procedures for OFAN. The activation encoder is also fitted to very little: one fixed waveform per training appliance, with no augmentation, so nothing pushes it to generalize beyond those few points, which is consistent with its failure on appliance types absent from training. Training it on many more appliance types, or with augmented queries is the obvious next step.

AI USE STATEMENT

In this work, we used generative AI tools to analyze our results and help interpret them, including suggesting the statistical testing procedures and the reference detectors used to evaluate the zero/few-shot results, and proposing explanations that we report as untested hypotheses. We have not used generative AI tools for idea formulation, for designing the experiments, or for creating or cleaning datasets, and the remaining disclosure tasks (synthetic data, mathematical claims and proofs, translation, and qualitative analysis) are not applicable to this work. Additionally, we used generative AI tools to explore prior work and check cited works against their sources, and to improve the writing and structure of the paper. We have reviewed all AI-assisted work and all text was revised by the authors. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

All experiments use the public UK-DALE and REFIT datasets. Section 3 and Appendix A give the training, validation and test houses for each appliance; Appendix A and Table 2 give the preprocessing, hyperparameters and training protocol of each backbone; Figures 3 and 4 give the representative-activation extraction and the conditioning mechanism; Appendix A.2 gives the evaluation metrics and statistical tests; Appendix C gives the protocol for transfer without retraining, its two query sources and the detector baselines; and Appendix D gives the architecture, optimizer and protocol of the QueryNILM baseline, which we reimplemented from the authors' public code and trained under our own data pipeline. Every configuration was trained with seeds 0–4. Upon publication, we will release the training and evaluation code, including our QueryNILM implementation and the configuration files for every run, the dataset split definitions, the trained weights for every configuration and seed, the per-seed test results from which every table and confidence interval in the paper is computed.

REFERENCES

Department for Energy Security and Net Zero. Smart Metering roll-out in Great Britain: Q4 2025 Smart Meters Statistics Report. Technical report, HM Government, 2026. URL https://www.gov.uk/government/statistics/smart-meters-in-great-britain-quarterly-update-december-2025.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018. URL http://arxiv.org/abs/1810.04805.

Michele D'Incecco, Stefano Squartini, and Mingjun Zhong. Transfer learning for non-intrusive load monitoring, 2019. URL https://arxiv.org/abs/1902.08835.

Will Gans, Anna Alberini, and Alberto Longo. Smart meter devices and the effect of feedback on residential electricity consumption: Evidence from a natural experiment in Northern Ireland. *Energy Economics*, 36:729–743, 2013. ISSN 0140-9883. doi: https://doi.org/10.1016/j.eneco.2012.11.022. URL https://www.sciencedirect.com/science/article/pii/S0140988312003209.

L. E. Garcia-Marrero, G. Petrone, and E. Monmasson. Lightweight and scalable transfer learning framework for load disaggregation, 2026. URL https://arxiv.org/abs/2603.04998.

Diego García, Daniel Pérez, Panagiotis Papapetrou, Ignacio Diaz, Abel A. Cuadrado, Jose M. Enguita, and Manuel Dominguez. Conditioned fully convolutional denoising autoencoder for multi-target NILM. *Neural Computing and Applications*, 37(17):10491–10505, 2025. doi: 10.1007/s00521-024-10552-0. URL https://doi.org/10.1007/s00521-024-10552-0.

George William Hart. Nonintrusive appliance load monitoring. *Proceedings of the IEEE*, 80(12): 1870–1891, 1992. doi: 10.1109/5.192069.

Jack Kelly and William Knottenbelt. The UK-DALE dataset, domestic appliance-level electricity demand and whole-house demand from five UK homes. *Scientific Data*, 2(1), March 2015a. ISSN 2052-4463. doi: 10.1038/sdata.2015.7. URL http://dx.doi.org/10.1038/sdata.2015.7.

Jack Kelly and William J. Knottenbelt. Neural NILM: deep neural networks applied to energy disaggregation. *CoRR*, abs/1507.06594, 2015b. URL http://arxiv.org/abs/1507.06594.

Hana Kim, Desy Caesary, Jeongwoo Jang, and Daphne Ngar yin Mah. Who engages in electricity conservation and to what effect after real-world, high-resolution feedback? an empirical analysis of Korean households with smart meters. *Heliyon*, 10(17):e36951, 2024. ISSN 2405-8440. doi: https://doi.org/10.1016/j.heliyon.2024.e36951. URL https://www.sciencedirect.com/science/article/pii/S2405844024129829.

J. Zico Kolter and Matthew J. Johnson. REDD: A public data set for energy disaggregation research. In *SustKDD: ACM SIGKDD Workshop on Data Mining Applications in Sustainability*, pp. 59–62, 2011.

Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 3045–3059, 2021.

Dandan Li, Jiangfeng Li, Xin Zeng, Vladimir Stankovic, Lina Stankovic, Changjiang Xiao, and Qingjiang Shi. Transfer learning for multi-objective non-intrusive load monitoring in smart building. *Applied Energy*, 329:120223, 2023. ISSN 0306-2619. doi: https://doi.org/10.1016/j.apenergy.2022.120223. URL https://www.sciencedirect.com/science/article/pii/S0306261922014805.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization, 2019. URL https://arxiv.org/abs/1711.05101.

David Murray, Lina Stankovic, and Vladimir Stankovic. An electrical load measurements dataset of United Kingdom households from a two-year longitudinal study. *Scientific Data*, 4(1):160122, 2017. doi: 10.1038/sdata.2016.122. URL https://doi.org/10.1038/sdata.2016.122.

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. FiLM: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

Anthony Petralia, Paul Charpentier, Yassine Kadhi, and Themis Palpanas. NILMFormer: Non-intrusive load monitoring that accounts for non-stationarity. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, volume 2, pp. 4761–4772, 2025. doi: 10.1145/3716013.3725515.

Shijie Yang, Jie Jiang, Yichen Zhang, Qiuqiang Kong, and Qianwen Cao. QueryNILM: Query based non-intrusive load monitoring with appliance embeddings. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, pp. 1–7, 2025. doi: 10.1109/IJCNN64981.2025.11227967.

Peipei You, Sitao Li, Chengren Li, Chao Zhang, Hailang Zhou, Huicai Wang, Huiru Zhao, and Yihang Zhao. Price-based demand response: A three-stage monthly time-of-use tariff optimization model. *Energies*, 16(23), 2023. ISSN 1996-1073. doi: 10.3390/en16237858. URL https://www.mdpi.com/1996-1073/16/23/7858.

Zhenmei Yue, Mingjun Weng, and Yuxuan Meng. BERT4NILM: A bidirectional transformer for non-intrusive load monitoring. In *Proceedings of the 5th International Workshop on Non-Intrusive Load Monitoring*, pp. 1–5, 2020. doi: 10.1145/3427771.3429390.

Ahmed Zoha, Alexander Gluhak, Muhammad Ali Imran, and Sutharshan Rajasegarar. Non-intrusive load monitoring approaches for disaggregated energy sensing: A survey. *Sensors*, 12(12):16838–16866, 2012. ISSN 1424-8220. doi: 10.3390/s121216838. URL https://www.mdpi.com/1424-8220/12/12/16838.

A Experimental Setup

Preprocessing. Raw power signals are resampled to a model-specific target period. BERT4NILM operates at 6 s resolution on UK-DALE and 8 s on REFIT, matching the native dataset sampling rates. NILMformer operates at 60 s on both datasets, consistent with the protocol of the original paper (Petralia et al., 2025). Short measurement gaps of up to 30 samples are forward-filled, treating them as transient sensor dropout. Houses in which a target appliance is not sub-metered are excluded from training entirely; zero-filling absent appliances with zero power was considered but found to degrade MAE substantially (14.3 W vs. 7.8 W for the kettle on UK-DALE) and was not used.

Power signals are normalised using model-specific schemes. BERT4NILM applies Z-score normalisation to the aggregate mains signal and, using per-appliance training-set statistics, normalises the appliance signal to $[0, 1]$; predictions are inverse-transformed to watts before evaluation. NILMformer applies max-scaling: both the mains signal and the appliance target are divided by the maximum mains power observed in the training set, placing all signals in $[0, 1]$ relative to peak consumption. This shared scale ensures the model sees appliance power in physically meaningful proportion to the aggregate, and is consistent with the original NILMformer training procedure (Petralia et al., 2025). Appliance-specific power thresholds are applied after inverse transformation to suppress low-power noise in the predictions.

A.1 Hyperparameters and Training Protocol

All models are trained with the AdamW optimizer (Loshchilov & Hutter, 2019) ($\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$, weight decay $= 0$) using mean squared error on the power signal as the sole training loss. Each configuration is run with five independent random seeds (0–4); all reported metrics are mean $\pm$ standard deviation across seeds. Table 2 summarizes the model-specific hyperparameters.

Table 2: Hyperparameters used for each backbone. Values are taken directly from the respective original implementations.

Hyperparameter	**BERT4NILM**	**NILMformer**
Window size	480	256
Hidden dimension d	256	256
Transformer layers	2	3
Attention heads	2	8
Dropout	0.1	0.2
Batch size	128	64
Learning rate	10^{-4}	10^{-4}
LR schedule	Cosine + warmup	ReduceLROnPlateau
Warmup	10% of steps	—
Plateau patience	—	3 epochs
Training epochs	100 (fixed)	100 (ceiling)
Early stopping	disabled	patience = 5 (val MAE)
Restore best weights	×	✓
Training window stride	240 (50% overlap)	256 (non-overlapping)
Test window stride	480 (non-overlapping)	256 (non-overlapping)
Masked training	25% token masking	—
Dataloader shuffle	✓	×
Validation source	Separate house list	20% of training windows

The two protocols differ in several respects that reflect each model's original design. BERT4NILM uses cosine annealing with a 10% linear warmup, a fixed 100-epoch budget with no early stopping, and BERT-style masked training in which 25% of input positions are selected before the forward pass, of which 80% are set to -1, 10% receive Gaussian noise and 10% are left unchanged. Training windows overlap by 50% (stride 240), and the dataloader is shuffled, consistent with the original BERT4NILM protocol (Yue et al., 2020).

NILMformer uses `ReduceLROnPlateau` (patience 3, factor 0.5), monitoring validation MAE to reduce the learning rate on plateau. Training is stopped early when validation MAE fails to improve for 5 consecutive epochs (patience 5), and the best-validation checkpoint is restored before final evaluation. This mirrors the original NILMformer training procedure (Petralia et al., 2025). Training windows are non-overlapping (stride equal to window size), the dataloader preserves temporal order, and no token masking is applied. Because NILMformer operates as a pure regression model, it reports MAE and MRE only; the classification metrics (F1, precision, recall) reported for BERT4NILM are not applicable to it.

Validation data. BERT4NILM validates on a separate list of houses: House 6 for REFIT and, for UK-DALE, House 1 (House 5 for the washing machine), which are also training houses. Because BERT4NILM trains for a fixed 100 epochs without early stopping and keeps its final weights, the validation data do not select the reported model. NILMformer validates on a random 20% of the training windows, and this split drives both its learning-rate schedule and its early stopping, as in the original NILMformer protocol. Keeping each model's original protocol makes comparisons between configurations of the same backbone controlled, but not comparisons between backbones.

A.2 Evaluation Metrics

All metrics are computed on the test set at native prediction resolution and reported as mean $\pm$ standard deviation across five random seeds. Let y_t and $\hat{y}_t$ denote the ground-truth and predicted appliance power at timestep t, and let $\mathcal{T}$ and $\mathcal{T}^{\text{on}}$ denote the full test set and the subset of timesteps where the true or the predicted state is on, respectively, with on/off states defined below.

MAE. Mean absolute error in watts over all timesteps:

$$\text{MAE} = \frac{1}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} |y_t - \hat{y}_t| \tag{3}$$

On-MAE. MAE restricted to on-state timesteps:

$$\text{On-MAE} = \frac{1}{|\mathcal{T}^{\text{on}}|} \sum_{t \in \mathcal{T}^{\text{on}}} |y_t - \hat{y}_t| \tag{4}$$

This decouples active-cycle accuracy from the trivially easy zero-load regime, which dominates overall MAE for appliances with low duty cycles such as the dishwasher and washing machine.

MRE / On-MRE. Mean relative error as in BERT4NILM (Yue et al., 2020): the absolute error at each timestep divided by the larger of the true and predicted power, averaged over all timesteps and over on-state timesteps respectively:

$$\text{MRE} = \frac{1}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} \frac{|y_t - \hat{y}_t|}{\max(y_t, \hat{y}_t)} \qquad \text{On-MRE} = \frac{1}{|\mathcal{T}^{\text{on}}|} \sum_{t \in \mathcal{T}^{\text{on}}} \frac{|y_t - \hat{y}_t|}{\max(y_t, \hat{y}_t)} \tag{5}$$

MRE provides a scale-independent view of relative error, facilitating comparison across appliances with very different power magnitudes.

F1, Precision, Recall, Accuracy. Binary on/off state detection metrics. The predicted state is obtained by thresholding the predicted power at an appliance-specific threshold τ_a; the ground-truth state uses the same threshold and additionally discards on and off periods shorter than appliance-specific minimum durations.

NILMformer is a pure regression model with no status output and is evaluated only on MAE and MRE, consistent with the original evaluation protocol (Petralia et al., 2025). F1 is reported for BERT4NILM only; On-MAE, On-MRE, precision, recall and accuracy are computed but not tabulated.

Statistical testing. All configurations are trained with the same five seeds, so seed s of two configurations shares its initialisation and data order. For each appliance we compute the per-seed difference between two configurations and report its mean with a 95% confidence interval from the t distribution with four degrees of freedom. A difference counts as an improvement or a degradation

only if this interval excludes zero. For dataset-level comparisons we first average, within each seed, the relative MAE difference over all appliances of the dataset, and then compute the same interval over the five seed averages. The seed is the unit of replication: in the combined configurations one model per seed produces the results for every appliance, so appliance–seed pairs are not independent. MAE differences are taken relative to the baseline because errors in watts are not comparable across appliances; F1 differences are averaged without normalisation.

```
def extract_representative(signal):
    acts           = extract_activations(signal)    # on-cycles
    fft_norm       = normalise(rfft(acts))          # unit FFT
    labels         = KMeans(n_clusters=5).fit_predict(fft_norm)
    dominant       = argmax(bincount(labels))
    members        = fft_norm[labels == dominant]
    centroid       = members.mean(axis=0)
    best           = argmin(norm(members - centroid, axis=1))
    return acts[where(labels == dominant)[best]]
```

Figure 3: Representative extraction: activations clustered in FFT space via K-means ($k = 5$); the member nearest to the centroid of the largest cluster is returned.

```
def appliance_embedding(x, appliance_input):
    token = appliance_encoder(appliance_input) if
        use_activation_embedding \
            else embedding.get(appliance_input)    # (b, 1, e)
    x     = feature_processor(cat([x, token]))     # (b, s+1, e)
    return x[:, :-1, :]                            # (b, s, e)

def forward(mains_power, appliance_input=None):
    x = embedd(mains_power.unsqueeze(1))           # (b,w) -> (b,w/2,e)
    x = appliance_embedding(x, appliance_input)    # inject query
    x = unpool(x.permute(0,2,1)).permute(0,2,1)    # (b,w/2,e)->(b,w,e)
    return linear_out(x).squeeze(-1)               # (b,w) predictions
```

Figure 4: OFAN forward pass. The appliance token is appended to the embedded power sequence before the transformer (in feature_processor), allowing every self-attention layer to attend jointly over power positions and the appliance conditioning signal. The token is removed after processing so the output length matches the input window.

B Model Results

We present the results as discussed in Section 4 of both models BERT4NILM and NILMformer on both datasets, UK-DALE and REFIT in Tables 3, 4, 5 and 6. Each cell shows mean $\pm$ std across five seeds. Bold denotes the best result per appliance per metric; italics denote the second best.

C Transfer Without Retraining

Two query sources. We evaluate the BERT4NILM OFAN-Act-Comb models in two settings that differ only in where the query comes from. In the *training-signature* setting the model is conditioned on the representative activation it was trained with, so no data from the target home is used and the setting is zero-shot. In the *target-home* setting the query is extracted from the earliest 10% of the target channel's own record, which requires a sub-metered segment of that appliance in that home. Neither setting updates any weight. Table 7 applies both across datasets: each model is evaluated on every sub-metered channel of the dataset it was not trained on whose appliance type it was trained for, with seeds 0–4 and a non-overlapping stride. NILMformer is omitted because it outputs power only. Of 105 such channels, 101 have results for both query sources and are reported here.

Table 3: BERT4NILM results on UK-DALE (MAE in watts; F1 in $[0, 1]$, higher is better for F1, lower for MAE). Best per row in **bold**, second best in *italics*.

	MAE (W) ↓				F1 ↑			
Appliance	OFON-Indiv	OFAN-Indiv	OFAN-Comb	OFAN-Act-Comb	OFON-Indiv	OFAN-Indiv	OFAN-Comb	OFAN-Act-Comb
Kettle	*9.05* ±0.68	**8.75** ±0.72	18.14 ±3.08	19.54 ±1.91	*0.91* ±0.01	**0.92** ±0.02	0.63 ±0.12	0.58 ±0.09
Fridge	19.39 ±1.02	*19.11* ±0.37	**18.81** ±0.26	21.54 ±2.39	*0.84* ±0.01	*0.84* ±0.00	**0.85** ±0.00	0.81 ±0.03
Dishwasher	*31.70* ±5.43	**29.11** ±1.30	34.72 ±2.02	35.09 ±3.16	*0.64* ±0.08	**0.69** ±0.07	*0.64* ±0.09	0.58 ±0.08
Microwave	**6.51** ±0.08	6.64 ±0.31	*6.62* ±0.80	7.71 ±0.52	0.17 ±0.01	0.18 ±0.02	**0.41** ±0.11	*0.22* ±0.08
Wash. Machine	*4.93* ±0.36	**4.80** ±1.08	5.26 ±0.81	7.56 ±2.04	*0.71* ±0.07	**0.81** ±0.05	*0.73* ±0.07	0.61 ±0.08
Average	*14.32*	**13.68**	16.71	18.29	*0.65*	**0.69**	0.65	0.56

Table 4: NILMformer results on UK-DALE (MAE in watts; MRE unitless, lower is better for both). Classification metrics not applicable (regression-only model).

	MAE (W) ↓				MRE ↓			
Appliance	OFON-Indiv	OFAN-Indiv	OFAN-Comb	OFAN-Act-Comb	OFON-Indiv	OFAN-Indiv	OFAN-Comb	OFAN-Act-Comb
Kettle	*14.12* ±3.03	15.54 ±2.42	**10.66** ±2.10	23.74 ±1.03	*0.02* ±0.01	0.03 ±0.02	**0.01** ±0.00	0.03 ±0.00
Fridge	**19.64** ±0.65	*19.74* ±0.54	28.42 ±2.04	32.26 ±4.18	**0.62** ±0.01	*0.62* ±0.01	0.66 ±0.04	0.70 ±0.06
Dishwasher	**16.51** ±1.68	20.12 ±3.55	*19.00* ±3.29	19.38 ±4.30	0.07 ±0.04	0.08 ±0.05	**0.03** ±0.00	*0.05* ±0.03
Microwave	*7.60* ±0.40	**7.38** ±0.44	8.52 ±0.19	11.94 ±1.06	0.05 ±0.02	0.04 ±0.01	**0.04** ±0.00	*0.04* ±0.00
Wash. Machine	**12.36** ±3.45	*13.07* ±4.06	13.11 ±1.65	14.41 ±2.70	0.12 ±0.08	0.10 ±0.06	**0.03** ±0.00	*0.08* ±0.04
Average	**14.05**	*15.17*	15.94	20.35	0.17	*0.17*	**0.15**	0.18

The training signature wins on 88 of the 101 channels. Averaging the per-channel difference within each seed and testing over the five seeds gives $+0.199$ F1 (95% CI $[0.172, 0.227]$). The two settings are closest where the appliance has a distinctive high-power cycle that both queries capture (the REFIT kettle, 0.50 against 0.50), and furthest apart where a 10% segment holds few or unrepresentative cycles (the UK-DALE kettle, 0.76 against 0.00). The fridge is the weakest case: it is on 48% of the time in REFIT, so an always-on detector scores 0.48 and only 21 of its 34 channels are beaten.

Appliance types absent from training. The comparison above is restricted to appliance types the models were trained on. The remainder of this appendix asks whether the query alone can extend a model to a type it never saw: the same checkpoints are applied to every sub-metered UK-DALE channel with the query built from that channel's own record, and for 58 of those channels with the UK-DALE-trained model, and 61 with the REFIT-trained model, the appliance type is absent from training.

We test whether OFAN-Act-Comb can disaggregate an appliance from a query built in the target home. We use the BERT4NILM OFAN-Act-Comb checkpoints trained on UK-DALE and on REFIT, with all five seeds, and apply them to every sub-metered UK-DALE channel. NILMformer is omitted because it outputs power only, so the detection baselines below do not apply to it. Each channel is treated on its own, as when an appliance is registered in a single home: the earliest 10% of its record forms the signature segment, from which the representative activation is extracted with the K-means procedure used for the training signatures, and the remaining 90% is disaggregated and scored. Taking the signature from the start of the record keeps the evaluated period out of the query. Each model runs at its training sampling period (6 s or 8 s) with a non-overlapping stride; averaging predictions from strides of 1/2 and 1/4 of the window, or choosing the best stride per seed, changes the median F1 by less than 0.001 and the median MAE by less than 0.5%. Channels with fewer than 100 non-overlapping evaluation windows, or without a complete activation in the signature segment, are excluded, as are the four channels of house 3, which were not evaluated. This leaves 76 of the 103 channels for the UK-DALE-trained model and 75 for the REFIT-trained model; two channels per model combine seen and unseen appliance types and are left out of the tables.

Table 5: BERT4NILM results on REFIT (MAE in watts; F1 in $[0, 1]$). Best per row in **bold**, second best in *italics*.

	MAE (W) ↓				F1 ↑			
Appliance	OFON-Indiv	OFAN-Indiv	OFAN-Comb	OFAN-Act-Comb	OFON-Indiv	OFAN-Indiv	OFAN-Comb	OFAN-Act-Comb
Kettle	*9.63* ±0.13	**9.62** ±0.08	10.60 ±0.24	10.43 ±0.17	**0.66** ±0.01	**0.66** ±0.00	0.63 ±0.01	*0.64* ±0.00
Dishwasher	*30.62* ±1.15	34.50 ±5.22	**27.16** ±1.70	29.43 ±0.63	**0.54** ±0.04	0.48 ±0.01	*0.51* ±0.03	*0.51* ±0.05
Microwave	6.15 ±0.26	6.01 ±0.17	**4.93** ±0.47	*5.22* ±0.52	**0.24** ±0.01	**0.24** ±0.00	**0.24** ±0.01	0.22 ±0.02
Wash. Machine	*16.48* ±0.91	16.56 ±1.51	**16.19** ±0.77	17.63 ±1.19	**0.58** ±0.04	**0.58** ±0.06	*0.56* ±0.03	0.53 ±0.05
Average	15.72	16.67	**14.72**	*15.68*	**0.50**	*0.49*	0.48	0.47

Table 6: NILMformer results on REFIT (MAE in watts; MRE unitless, lower is better for both).

	MAE (W) ↓				MRE ↓			
Appliance	OFON-Indiv	OFAN-Indiv	OFAN-Comb	OFAN-Act-Comb	OFON-Indiv	OFAN-Indiv	OFAN-Comb	OFAN-Act-Comb
Kettle	**13.58** ±1.03	*13.96* ±1.44	17.68 ±6.09	15.11 ±0.31	0.03 ±0.03	*0.03* ±0.02	0.14 ±0.15	**0.02** ±0.00
Dishwasher	55.97 ±4.07	*54.45* ±7.64	57.54 ±13.70	**43.49** ±1.49	0.26 ±0.13	*0.21* ±0.09	0.25 ±0.17	**0.08** ±0.01
Microwave	*3.85* ±0.57	**3.64** ±0.39	8.85 ±7.12	8.98 ±1.34	0.04 ±0.02	*0.03* ±0.00	0.17 ±0.18	**0.02** ±0.00
Wash. Machine	*25.97* ±4.35	26.99 ±3.63	31.30 ±7.91	**24.18** ±1.95	*0.11* ±0.03	0.13 ±0.04	0.25 ±0.17	**0.08** ±0.01
Average	24.84	*24.76*	28.84	**22.94**	0.11	*0.10*	0.20	**0.05**

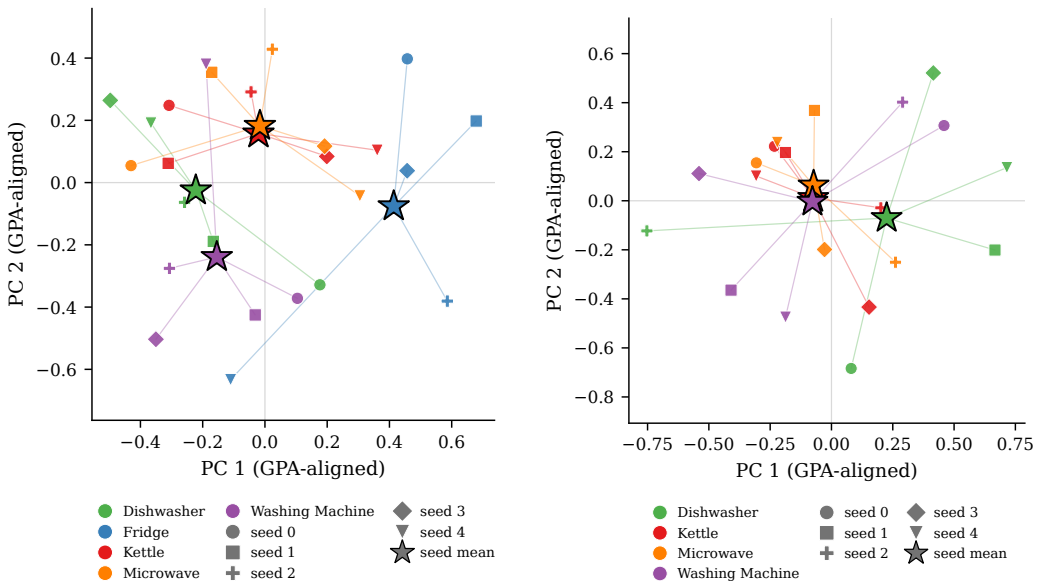
We group channels by whether their appliance type was a training target of the model and whether their house was used in training. F1 is compared with two references computed on the same channel: a detector that always predicts on, and a random detector that predicts on as often as the model does. A channel counts as above chance when the 95% confidence interval over seeds of the difference between the model's F1 and the random detector's excludes zero.

Table 8 separates the two cases. For appliance types absent from training, median F1 is 0.03–0.05 for the UK-DALE-trained model and 0.02 for the REFIT-trained model, and the median margin over the random detector is at most 0.02. Only 10 of 58 and 5 of 61 of these channels are above chance and beat the always-on detector, and none of them exceeds an F1 of 0.14. The unseen types that pass both references include heating appliances (electric oven, electric stove, toaster), whose power draw resembles that of the kettle and microwave; whether the model uses their queries or responds to kettle-like events is not tested here. For some appliance types seen in training, the query carries over to new houses and to a new dataset. The REFIT-trained model detects the UK-DALE dishwasher in all three houses (F1 0.53, 0.74 and 0.49, against 0.05–0.06 for always-on), and the UK-DALE-trained model detects the fridge in house 4, which it never saw in training (F1 0.74, against 0.55).

Table 9 compares the two query sources on house 2. With its own query, the UK-DALE-trained model matches the supervised dishwasher result (0.59 against 0.58) but loses most of its kettle (0.25 against 0.58) and washing-machine (0.05 against 0.61) detection. On the dishwasher, the F1 of the REFIT-trained model is higher than that of both supervised UK-DALE models. For appliance types seen in training, the source of the query can therefore change F1 by as much as 0.56.

D QUERYNILM BASELINE

Protocol. We use the architecture of the original paper: nine dilated layers with 32 residual, dilation and skip channels, which reproduces its reported parameter counts of 151,297 (one-hot query) and 358,957 (activation query), against 1.94M for one conditioned BERT4NILM and 9.69M for five per-appliance models. We keep the authors' Adam optimizer at $\mathrm{lr} = 10^{-3}$ and their sequence length of 1023 samples, and take everything else from the BERT4NILM protocol of Table 2: the same houses, splits, resampling, normalization and representative activations, 25% masked training, cosine schedule with warmup, batch size 128 and 100 epochs, with training windows overlapping by 50%. Each configuration is run with seeds 0–4, so the paired intervals of §A.2 apply. Two de-



(a) UK-DALE (b) REFIT

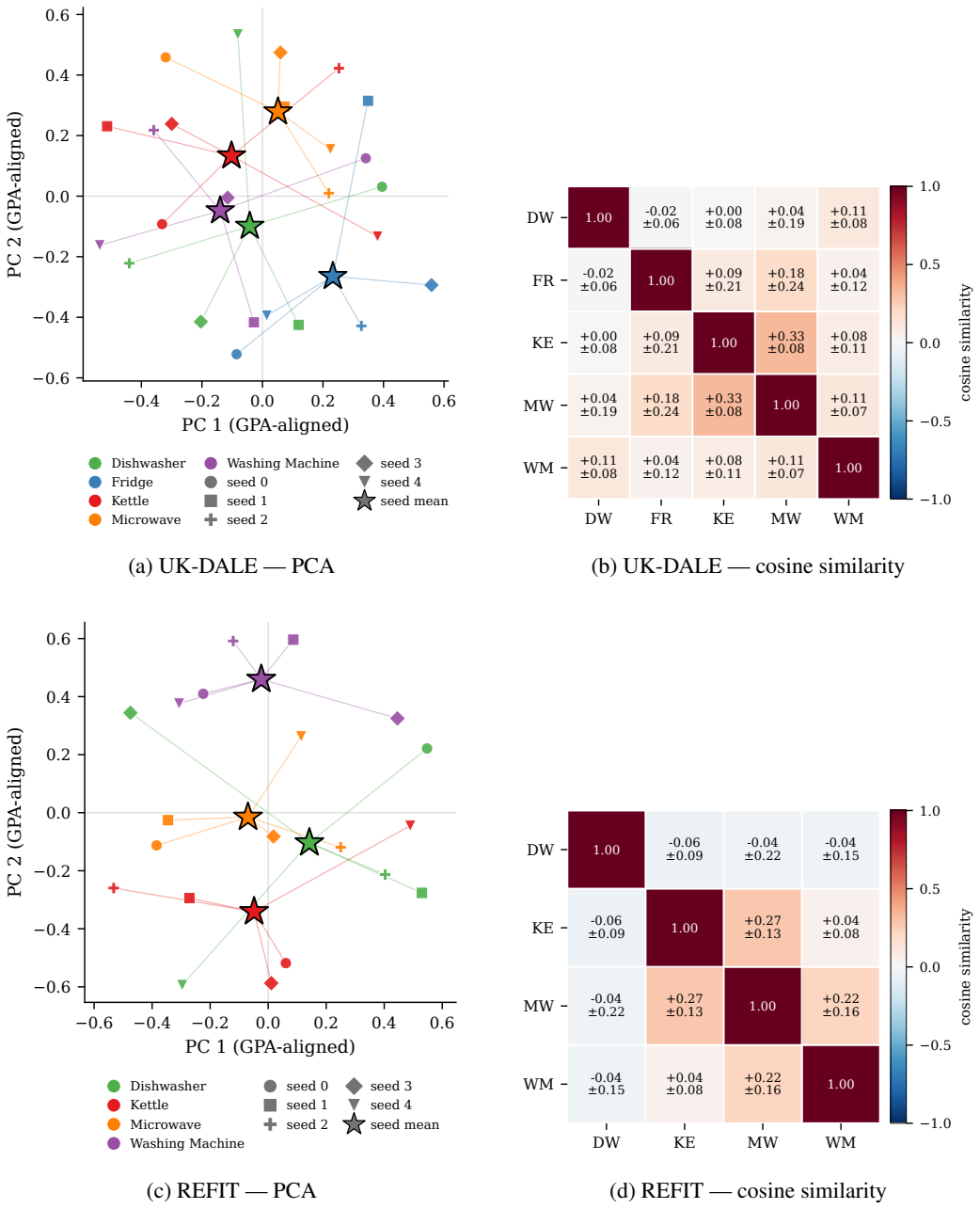
Figure 5: PCA projections of the BERT4NILM OFAN-Comb appliance embeddings. Seeds are aligned by generalised Procrustes analysis; lines join each appliance's per-seed points to its seed mean.

Table 7: Transfer to the other dataset, BERT4NILM OFAN-Act-Comb. F1 is the per-channel mean over five seeds, then averaged over channels. Training signature: no data from the target home. Target home: query from the earliest 10% of the channel. Always-on: a detector that always predicts on. Beats always-on counts channels where the training-signature F1 exceeds it; every one of the 101 channels is above a random detector with the model's on-rate.

Direction	Appliance	Channels	Training signature	Target home	Always-on	Beats always-on
REFIT → UK-DALE	Dishwasher	3	0.63	0.59	0.05	3/3
REFIT → UK-DALE	Kettle	4	0.76	0.00	0.01	4/4
REFIT → UK-DALE	Microwave	3	0.45	0.26	0.01	3/3
REFIT → UK-DALE	Wash. machine	3	0.63	0.17	0.07	3/3
UK-DALE → REFIT	Dishwasher	13	0.32	0.27	0.08	13/13
UK-DALE → REFIT	Fridge	34	0.49	0.23	0.48	21/34
UK-DALE → REFIT	Kettle	12	0.50	0.50	0.01	12/12
UK-DALE → REFIT	Microwave	14	0.17	0.10	0.01	14/14
UK-DALE → REFIT	Wash. machine	15	0.44	0.13	0.07	15/15
All		101	0.43	0.24	0.19	88/101

viations from the original are worth stating: its step-decay learning rate is replaced by the cosine schedule of our protocol, and we do not use its synthetic query augmentation, which our activation-embedding models also do not use. The conditioning is otherwise unchanged: a 128-dimensional query, either a one-hot vector or the output of the paper's convolutional–recurrent encoder applied to a representative activation, is projected by two linear layers inside every dilated block.

Activation queries are unstable. Conditioning QueryNILM on an encoded activation rather than a one-hot vector raises its own pooled MAE by 11.2% $[-3.3, +25.7]$ on UK-DALE and by 141.7% $[-0.2, +283.5]$ on REFIT, neither interval excluding zero. Its pooled F1 falls by 0.022 $[-0.079, +0.036]$ on UK-DALE and, significantly, by 0.212 $[-0.373, -0.052]$ on REFIT. The REFIT MAE interval is wide because the seed variance is extreme: the microwave averages $19.99\,\mathrm{W}$



(a) UK-DALE — PCA

(b) UK-DALE — cosine similarity

(c) REFIT — PCA

(d) REFIT — cosine similarity

Figure 6: NILMformer — OFAN-Comb appliance embeddings (mean ± std over 5 seeds).

with a standard deviation of 12.98 W and the kettle 25.07 ± 10.47 W, against 4.74 ± 0.26 and 9.83 ± 0.08 for the same model with one-hot queries (Table 10). Measured instead against the index-query OFAN-Comb, the activation-query QueryNILM is 15.8% $[+10.9, +20.7]$ worse on UK-DALE and 126.2% $[-14.2, +266.6]$ worse on REFIT; §4 compares it with OFAN-Act-Comb, which uses the same query type. The direction matches what we observe for our own backbones, where the activation query costs 18.1% (BERT4NILM) and 40.2% (NILMformer) on UK-DALE, but both the magnitude and the seed-to-seed spread are larger for QueryNILM's recurrent encoder than for our convolutional one. Because the original work pairs its activation query with a synthetic-waveform augmentation that we do not reproduce, we report this as the behavior of the query type under our protocol rather than as a reproduction of that result.

Table 8: Zero-shot detection with BERT4NILM OFAN-Act-Comb on UK-DALE channels, with each query built from the earliest 10% of the channel. F1 is the per-channel mean over five seeds, summarised by the median and interquartile range. Δ: F1 minus that of a random detector with the model's on-rate. Above chance: channels whose 95% confidence interval of Δ over seeds excludes zero. Beats always-on: channels whose F1 exceeds that of a detector that always predicts on. *Appliance types the model was trained on, in its training houses; these are not zero-shot and are shown for reference.

Trained on	Appliance type	Houses	Channels	F1 median [IQR]	Δ median	Above chance	Beats always-on
UK-DALE	seen	2, 4	6	0.42 [0.17, 0.61]	+0.27	4	5
UK-DALE	seen*	1, 5	10	0.49 [0.20, 0.64]	+0.31	7	9
UK-DALE	unseen	2, 4	10	0.05 [0.02, 0.11]	+0.02	3	1
UK-DALE	unseen	1, 5	48	0.03 [0.01, 0.09]	+0.00	17	14
REFIT	seen	1, 2, 5	12	0.21 [0.02, 0.46]	+0.17	8	9
REFIT	unseen	1, 2, 4, 5	61	0.02 [0.00, 0.03]	+0.00	27	9

Table 9: House 2 of UK-DALE, appliance types seen in training: F1 of the supervised UK-DALE models, which use the training signature, and of zero-shot BERT4NILM OFAN-Act-Comb with the query built from house 2's own record. Mean over five seeds. Zero-shot scores exclude the earliest 10% of the record, which forms the query. †: above chance, as in Table 8. Parentheses: the fridge is not a REFIT training appliance.

		Supervised (UK-DALE)		Zero-shot, trained on	
Appliance	Always-on	OFON-Indiv	OFAN-Act-Comb	UK-DALE	REFIT
Kettle	0.02	0.91	0.58	0.25	0.01
Fridge	0.62	0.84	0.81	0.62†	(0.03)†
Dishwasher	0.07	0.64	0.58	0.59†	0.74†
Microwave	0.01	0.17	0.22	0.15	0.45
Wash. machine	0.03	0.71	0.61	0.05†	0.08†

Table 10: QueryNILM results (MAE in watts; F1 in $[0, 1]$, higher is better for F1, lower for MAE). Each cell is the mean $\pm$ standard deviation over seeds 0–4.

	One-hot query		Activation query	
Appliance	MAE (W) $\downarrow$	F1 $\uparrow$	MAE (W) $\downarrow$	F1 $\uparrow$
UK-DALE				
Kettle	18.11 ± 1.51	0.63 ± 0.06	17.92 ± 3.25	0.63 ± 0.13
Fridge	19.57 ± 0.99	0.85 ± 0.01	22.28 ± 0.72	0.80 ± 0.01
Dishwasher	32.69 ± 2.14	0.61 ± 0.06	33.31 ± 2.22	0.58 ± 0.05
Microwave	10.04 ± 1.06	0.19 ± 0.03	10.65 ± 1.40	0.23 ± 0.05
Wash. Machine	4.01 ± 0.33	0.82 ± 0.01	5.30 ± 0.94	0.76 ± 0.07
Average	16.89	0.62	17.89	0.60
REFIT				
Kettle	9.83 ± 0.08	0.67 ± 0.01	25.07 ± 10.47	0.20 ± 0.26
Dishwasher	25.54 ± 1.76	0.59 ± 0.05	31.18 ± 9.69	0.54 ± 0.06
Microwave	4.74 ± 0.26	0.24 ± 0.01	19.99 ± 12.98	0.09 ± 0.08
Wash. Machine	13.44 ± 0.31	0.68 ± 0.01	22.99 ± 7.75	0.50 ± 0.10
Average	13.39	0.54	24.81	0.33