AlleNoise – large-scale text classification benchmark dataset with real-world label noise

Alicja Rączkowska* Allegro.com
Aleksandra Osowska-Kurczab* Allegro.com
Jacek Szczerbiński* Allegro.com
Kalina Jasińska-Kobus* Allegro.com
Klaudia Nazarko* Allegro.com

Abstract

Label noise remains a challenge for training robust classification models. Most methods for mitigating label noise have been benchmarked using primarily datasets with synthetic noise. While the need for datasets with realistic noise distribution has partially been addressed by web-scraped benchmarks such as WebVision and Clothing1M, those benchmarks are restricted to the computer vision domain. With the growing importance of Transformer-based models, it is crucial to establish text classification benchmarks for learning with noisy labels. In this paper, we present *AlleNoise*, a new curated text classification benchmark dataset with real-world instance-dependent label noise, containing over 500,000 examples across approximately 5,600 classes, complemented with a meaningful, hierarchical taxonomy of categories. The noise distribution comes from actual users of a major e-commerce marketplace, so it realistically reflects the semantics of human mistakes. In addition to the noisy labels, we provide human-verified clean labels, which help to get a deeper insight into the noise distribution, unlike web-scraped datasets typically used in the field. We demonstrate that a representative selection of established methods for learning with noisy labels is inadequate to handle such real-world noise. In addition, we show evidence that these algorithms do not alleviate excessive memorization. As such, with *AlleNoise*, we set the bar high for the development of label noise methods that can handle real-world label noise in text classification tasks. The code and dataset are available for download at https://github.com/allegro/AlleNoise.

Proceedings of the 28th International Conference on Artificial Intelligence and Statistics (AISTATS) 2025, Mai Khao, Thailand. PMLR: Volume 258.

1 INTRODUCTION

The problem of label noise poses a sizeable challenge for classification models (Frenay et al., 2014; Song, Kim, Park, et al., 2022). With modern deep neural networks, due to their capacity, it is possible to memorize all labels in a given training dataset (Rolnick et al., 2018). This, effectively, leads to overfitting to noise if the training dataset contains noisy labels, which in turn reduces the generalization capability of such models (Arpit et al., 2017; C. Zhang et al., 2017; C. Zhang et al., 2021).

Most previous works on training robust classifiers have focused on analyzing relatively simple cases of synthetic noise (Jindal, Nokleby, et al., 2017; Patrini et al., 2017), either uniform (i.e. symmetric) or class-conditional (i.e. asymmetric). It is a common practice to evaluate these methods using popular datasets synthetically corrupted with label noise, such as MNIST (L. Deng, 2012), ImageNet (J. Deng et al., 2009), CIFAR (Krizhevsky, 2009) or SVHN (Netzer et al., 2011). However, synthetic noise is not indicative of realistic label noise and thus deciding to use noisy label methods based on such benchmarks can lead to unsatisfactory results in real-world machine learning practice. Moreover, it has been shown that these benchmark datasets are already noisy themselves (Northcutt et al., 2021; Bo Liu et al., 2022), so the study of strictly synthetic noise in such a context is intrinsically flawed.

Realistic label noise is instance dependent, i.e. the labeling mistakes are caused not simply by label ambiguity, but by input uncertainty as well (Goldberger et al., 2017). This is an inescapable fact when human annotators are responsible for the labeling process (Krishna et al., 2016). However, many existing approaches for

*Authors contributed equally.

mitigating instance-dependent noise have one drawback in common - they had to, in some capacity, artificially model the noise distribution due to the lack of existing benchmark datasets (Nguyen et al., 2022; Gu et al., 2021; Chen et al., 2020; Xia et al., 2020; Algan et al., 2020; Berthon et al., 2021). In addition, most of the focus in the field has been put on image classification, but with the ever-increasing importance of Transformer-based (Vaswani et al., 2017) architectures, the problem of label noise affecting the fine-tuning of natural language processing models needs to be addressed as well. There are many benchmark datasets for text data classification (Maas et al., 2011; Lin et al., 2019; Wang et al., 2019; Bhatia et al., 2016), but none of them are meant for the study of label noise. In most cases, the actual level of noise in these datasets is unknown, so using them for benchmarking label noise methods is unfeasible.

Moreover, the datasets used in this research area usually contain relatively few labels. The maximum reported number of labels is 1000 (Li et al., 2017). As such, there is a glaring lack of a benchmark dataset for studying label noise that provides realistic real-world noise, a high number of labels and text data at the same time.

We see a need for a textual benchmark dataset that would provide realistic instance-dependent noise distribution with a known level of label noise, as well as a relatively large number of target classes, with both clean and noisy labels. To this end, in this paper we provide the following main contributions:

- We introduce *AlleNoise* - a benchmark dataset for multi-class text classification with real-world label noise. The dataset consists of 502,310 short texts (e-commerce product titles) belonging to 5,692 categories (taken from a real product assortment tree). It includes a noise level of 15%, stemming from mislabeled data points. This amount of noise reflects the actual noise distribution in the data source (Allegro.com e-commerce platform). For each of the mislabeled data instances, the true category label was determined by human domain experts.

- We benchmark a comprehensive selection of well-established methods for classification with label noise against the real-world noise present in *AlleNoise* and compare the results to synthetic label noise generated for the same dataset. Our results reveal that while SOTA methods perform well on synthetic noise, they struggle with real-world label noise, exposing the limitations of synthetic noise distributions as a basis for evaluating model robustness in the field.

2 RELATED WORK

Several classification benchmarks with real-world instance-dependent noise have been reported in the literature. ANIMAL-10N (Song, Kim, and Lee, 2019) is a human-labeled dataset of confusing images of animals, with 10 classes and an 8% noise level. CIFAR-10N and CIFAR-100N (Wei et al., 2022) are noisy versions of the CIFAR dataset, with labels assigned by crowdsourced human annotators. CIFAR-10N is provided in three versions, with noise levels of 9%, 18% and 40%, while CIFAR-100N has a noise level of 40%. Clothing1M (Xiao et al., 2015) is a large-scale dataset of fashion images crawled from several online shops. It contains 14 classes and the estimated noise rate is 38%. Similarly, WebVision (Li et al., 2017) comprises of images crawled from the web, but it is more general - it has 1000 categories of diverse images. The estimated noise level is 20%. DCIC (Schmarje et al., 2022) is a benchmark that consists of 10 real-world image datasets, with several human annotations per image. This allows for testing algorithms that utilize soft labels to mitigate various kinds of annotation errors. The maximum number of classes in the included datasets is 10.

With the focus in the label noise field being primarily on images, the issue of noisy text classification remains relatively unexplored. Previous works have either utilized existing classification datasets with synthetic noise (Jindal, Pressel, et al., 2019; Bo Liu et al., 2022; Nguyen et al., 2022) or introduced new datasets with real-world noise. NoisyNER (Hedderich, Zhu, et al., 2021) contains annotated named entity recognition data in the Estonian language, assigned to 4 categories. The authors do not mention the noise level, only that they provide 7 variants of real-world noise. NoisywikiHow (Wu et al., 2023) is a dataset of articles scraped from the wikiHow website, with accompanying 158 article categories. The data was manually cleaned by human annotators, which eliminated the real-world noise distribution. The authors performed experiments by injecting synthetic noise into their dataset. Thus, NoisywikiHow is not directly comparable to *AlleNoise*. Another two datasets are Hausa and Yorùbá (Hedderich, Adelani, et al., 2020), text classification datasets of low-resource African languages with 5 and 7 categories respectively. They both include real-world noise with the level of 50.37% for the former, and 33.28% for the latter.

While there is a number of text datasets containing e-commerce product data (Lin et al., 2019; Nguyen et al., 2022; Bhatia et al., 2016), none of them have verified clean labels and in most cases the noise level is unknown. Similarly, classification settings with large

(a) **(b)**

Figure 1: Symmetric noise vs. *AlleNoise* in examples. Correct and noisy labels are marked in green and red, respectively. **(a)** Symmetric noise: an electric toothbrush incorrectly labeled as a winter tire is easy to spot, even for an untrained human. **(b)** *AlleNoise*: a ceiling dome is mislabeled as a pendant lamp. This error is semantically challenging and hard to detect. Note: *AlleNoise* dataset does not include images.

Table 1: Comparison of *AlleNoise* to previously published datasets created for studying the problem of learning with noisy labels. All datasets contain real-world noise. *AlleNoise* is the biggest text classification dataset in this field, has a known level of label noise and provides clean labels in addition to the noisy ones.

Dataset	Modality	Total examples	Classes	Noise level	Clean label
ANIMAL10N	Images	55k	10	8%	✓
CIFAR10N	Images	60k	10	9/18/40%	✓
CIFAR100N	Images	60k	10	40%	✓
WebVision	Images	2.4M	1000	∼20%	✗
Clothing1M	Images	1M	14	∼38%	✗
Hausa	Text	2,917	5	50.37%	✓
Yorùbá	Text	1,908	7	33.28%	✓
NoisyNER	Text	217k	4	unspecified	✓
AlleNoise	**Text**	**500k**	**5692**	**15%**	✓

numbers (i.e. more than 1000) of classes were not addressed up to this point in the existing datasets (**Tab. 1**).

3 ALLENOISE DATASET

We introduce *AlleNoise* - a benchmark dataset for large-scale multi-class text classification with real-world label noise. The dataset consists of 502,310 e-commerce product titles listed on Allegro.com in 5,692 assortment categories, collected in January of 2022. 15% of the products were listed in wrong categories, hence for each entry the dataset includes: the product title, the category where the product was originally listed, and the category where it should be listed according to human experts. See Appendix **E** for exploratory data analysis of the dataset.

Additionally, we release the taxonomy of product categories in the form of a mapping (category ID → path in the category tree), which allows for fine-grained exploration of noise semantics.

3.1 Real-world noise

We collected 74,094 mislabeled products from two sources: 1) customer complaints about a product being listed in the wrong category - such requests usually suggest the true category label, 2) assortment clean-up by internal domain experts, employed by Allegro - products listed in the wrong category were manually moved to the correct category.

The resulting distribution of label noise is not uniform over the entire product assortment - most of the noisy instances belong to a small number of categories. Such asymmetric distribution is an inherent feature of real-world label noise. It is frequently modeled with class-conditional synthetic noise in related literature. However, since the mistakes in *AlleNoise*

Table 2: *AlleNoise* dataset contents. **(a)** The primary *AlleNoise* table includes the true and noisy label for each product title. **(b)** The second table maps the labels to category names.

Product title	Category label	True category label
Emporia PURE V25 BLACK	352	170
Metal Hanging Lid Rack Suspended	68710	321104
Miraculum Asta Plankton C Active Serum-Booster	5360	89000

(a) Primary *AlleNoise* data

Category label	Category name
352	Electronics > Phones and Accessories > GSM Accessories > Batteries
170	Electronics > Phones and Accessories > Smartphones and Cell Phones
68710	Home and Garden > Equipment > Kitchen Utensils > Pots and Pans > Lids
321104	Home and Garden > Equipment > Kitchen Utensils > Pots and Pans > Organizers
5360	Allegro > Beauty > Care > Face > Masks
89000	Allegro > Beauty > Care > Face > Serum

(b) Category label mapping data

were based not only on the category name, but also on the product title, our noise distribution is in fact instance-dependent.

3.2 Clean data sampling

The 74,094 mislabeled products were complemented with 428,216 products listed in correct categories. The clean instances were sampled from the most popular items listed in the same categories as the noisy instances, proportionally to the total number of products listed in each category. The high popularity of the sampled products guarantees their correct categorization, because items that generate a lot of traffic are curated by human domain experts. Thus, the sampled distribution was representative for a subset of the whole marketplace: 5,692 categories out of over 23,000, for which label noise is particularly well known and described.

3.3 Post-processing

We automatically translated all 500k product titles from Polish to English. Machine translation is a common part of e-commerce, many platforms incorporate it in multiple aspects of their operation (Tan et al., 2020; B. Zhang et al., 2023). Moreover, it is an established practice to publish machine-translated text in product datasets (Ni et al., 2019). Categories related to sexually explicit content were removed from the dataset altogether. Finally, categories with less than 5 products were removed from the dataset to allow for five-fold cross-validation in our experiments.

4 METHODS

4.1 Problem statement

Let $\mathcal{X}$ denote the input feature space, and $\mathcal{Y}$ be a set of class labels. In a typical supervised setting, each instance x_i has a true class label y_i. However, in learning with noisy labels, $\tilde{y}_i$ is observed instead, which is with an unknown probability p (noise level) changed from the true y_i.

In this setting, we train a classifier $f : \mathcal{X} \to \mathcal{Y}$ that generalizes knowledge learnt from a dataset $\mathcal{D}$, consisting of training examples $(x_i, \tilde{y}_i)$. Because $\tilde{y}_i$ can be affected by label noise, the model's predictions $\hat{y}_i = f(x_i)$ might be corrupted by the distribution of noisy labels as well. Maximizing the robustness of such a classifier implies reducing the impact of noisy training samples on the generalization performance. In the *AlleNoise* dataset, x_i corresponds to the product title, $\tilde{y}_i$ is the original product category, and y_i is the correct category.

4.2 Synthetic noise generation

In order to compare the real-world noise directly with synthetic noise, we applied different kinds of synthetic noise to the clean version of *AlleNoise*: the synthetic noise was applied to each instance's true label y_i, yielding a new synthetic noisy label $\tilde{y}_i$. Overall, the labels were flipped for a controlled fraction $p = 15\%$ of all instances. We examined the following types of synthetic noise:

- Symmetric noise: each instance is given a noisy label different from the original label, with uniform

probability p.

- Class-conditional pair-flip noise: each instance in class j is given a noisy label $j+1$ with probability p.

- Class-conditional nested-flip noise: we only flip categories that are close to each other in the hierarchical taxonomy of categories. For example, for the parent category *Car Tires* we perform a cyclic flip between its children categories: *Summer* $\rightarrow$ *Winter* $\rightarrow$ *All-Season* $\rightarrow$ *Summer* with probability p. Thus, the noise transition matrix is a block matrix with a small number of off-diagonal elements equal to p.

- Class-conditional matrix-flip noise: the transition matrix between classes is approximated with the baseline classifier's confusion matrix. The confusion matrix is evaluated against the clean labels on 8% of the dataset (validation split) (Patrini et al., 2017). The resulting noise distribution is particularly tricky: we flip the labels between the classes that the model is most likely to confuse.

4.3 Model training

Next, we evaluated several algorithms for training classifiers under label noise. For a fair comparison, all experiments utilized the same classifier architecture as well as training and evaluation loops. We followed a fine-tuning routine that is typical for text classification tasks. In particular, we vectorized text inputs with XLMRoberta (Conneau et al., 2019), a multilingual text encoder based on the Transformer architecture (Vaswani et al., 2017). To provide the final class predictions, we used a single fully connected layer with a softmax activation and the number of neurons equal to the number of classes. During training, all weights in the model, including those of the representation layers, were unfrozen and subject to adaptation. The baseline model used cross-entropy (CE) as a loss function. To isolate the effects of label noise, we employed a canonical architecture without additional regularization or modifications.

Models were trained with the AdamW optimiser and linear LambdaLR scheduling (warmup steps = 100). We have not used any additional regularization, i.e. weight decay or dropout. Key training parameters, such as batch size (bs = 256) and learning rate (lr = 10^{-4}) were tuned to maximize the validation accuracy on the clean dataset. All models have been trained for 10 epochs. Training of the baseline model, accelerated with a single NVIDIA A100 40GB GPU, lasted for about 1 hour.

We used five-fold stratified cross-validation to comprehensively evaluate the results of the models trained with label noise. For each fold, the full dataset was divided into three splits: $\mathcal{D}_{train}$, $\mathcal{D}_{val}$, $\mathcal{D}_{test}$, in proportion 72% : 8% : 20%. Following the literature on learning with noisy labels (Song, Kim, Park, et al., 2022), both $\mathcal{D}_{train}$ and $\mathcal{D}_{val}$ were corrupted with label noise, while $\mathcal{D}_{test}$ remained clean.

All of the results presented in this study correspond to the last checkpoint of the model. We use the following format for presenting the experimental results: [m]$\pm$[s], where m is an average over the five cross-validation folds, while s is the standard deviation. Experiments used a seeded random number generator to ensure the reproducibility of the results.

4.4 Evaluation metrics

Accuracy on the clean test set is the key metric in our study. We expect that methods that are robust to the label noise observed in the training phase, should be able to improve the test accuracy when compared to the baseline model.

Additionally, to better understand the difference between synthetic and real-world noise, we collected detailed validation metrics. The validation dataset $\mathcal{D}_{val}$ contained both instances for which the observed label $\tilde{y}_i$ was incorrect ($\mathcal{D}_{val}^{\texttt{noisy}}$) and correct ($\mathcal{D}_{val}^{\texttt{clean}}$). Noisy observations from $\mathcal{D}_{val}^{\texttt{noisy}}$ were used to measure the memorization metric $\texttt{memorized}_{val}$, defined as a ratio of predictions $\hat{y}_i$ that match the noisy label $\tilde{y}_i$. Notice that our memorization metric is computed on the validation set, contrary to the training set typically used in the literature (S. Liu et al., 2020). Our metric increases when the model not only memorizes incorrect classes from the training observations, but also repeats these errors on unseen observations. Furthermore, we compute accuracy on $\mathcal{D}_{val}^{\texttt{noisy}}$ denoted as $\texttt{correct}_{val}^{\texttt{noisy}}$ and its counterpart on the clean fraction, $\texttt{correct}_{val}^{\texttt{clean}}$.

4.5 Benchmarked methods

We evaluated the following methods for learning with noisy labels: Self-Paced Learning (SPL) (Kumar et al., 2010), Provably Robust Learning (PRL) (Boyang Liu et al., 2021), Early Learning Regularization (ELR) (S. Liu et al., 2020), Generalized Jensen-Shannon Divergence (GJSD) (Englesson et al., 2021), Co-teaching (CT) (Han et al., 2018), Co-teaching+ (CT+) (Yu et al., 2019), Mixup (MU) (H. Zhang et al., 2018) and Pseudo-Label Selection (PLS) (Albert et al., 2022). Additionally, we implemented Clipped Cross-Entropy as a simple baseline (see Appendix **A**). These approaches represent a comprehensive selection of different method families: novel loss functions (GJSD), noise filtration

Table 3: Accuracy of the evaluated methods on the clean dataset compared to various noisy datasets with 15% noise level. The noisy datasets include *AlleNoise*, symmetric synthetic noise, and asymmetric synthetic noises: pair-flip, nested-flip, and matrix-flip. * marks cases equivalent to the baseline CE. ↓ marks results significantly worse (>1 p.p.) than the baseline CE. Best results for each noise type are bolded.

	Clean set	Symmetric	Pair-flip	Nested-flip	Matrix-flip	AlleNoise
CE	**72.00 ± 0.10**	69.56 ± 0.10	69.38 ± 0.11	69.16 ± 0.09	68.23 ± 0.07	61.00 ± 0.17
ELR	71.97 ± 0.22	69.73 ± 0.16	**70.56 ± 0.09**	**70.53 ± 0.15**	**69.71 ± 0.16**	**61.13 ± 0.15**
MU	71.84 ± 0.16	69.39 ± 0.12	69.42 ± 0.14	69.22 ± 0.18	68.25 ± 0.11	60.81 ± 0.26
CCE	71.88 ± 0.12	70.34 ± 0.12	69.32 ± 0.14	69.10 ± 0.10	68.17 ± 0.07	60.95 ± 0.11
CT	*72.00 ± 0.10	70.08 ± 0.09	69.54 ± 0.14	69.15 ± 0.22	68.29 ± 0.12	60.69 ± 0.10
CT+	*72.00 ± 0.10	↓66.89 ± 0.61	↓67.38 ± 0.35	↓67.42 ± 0.31	↓66.30 ± 0.61	↓58.76 ± 0.18
PRL	*72.00 ± 0.10	69.51 ± 0.18	69.61 ± 0.15	69.46 ± 0.11	68.72 ± 0.11	↓59.67 ± 0.16
SPL	*72.00 ± 0.10	70.02 ± 0.09	↓66.18 ± 0.16	↓65.42 ± 0.29	↓63.65 ± 0.09	↓57.09 ± 0.11
GJSD	71.79 ± 0.14	**70.55 ± 0.12**	69.17 ± 0.19	68.82 ± 0.15	68.09 ± 0.07	60.97 ± 0.05
PLS	71.08 ± 0.14	69.14 ± 0.20	↓67.38 ± 0.24	↓67.34 ± 0.11	↓67.07 ± 0.21	60.18 ± 0.23

(SPL, PRL, CCE, CT, CT+), robust regularization (ELR, PLS), data augmentation (MU) and training loop modifications (CT, CT+, PLS).

These methods are implemented with a range of technologies and software libraries. As such, in order to have a reliable and unbiased framework for comparing them, it is necessary to standardize the software implementation. To this end, we re-implemented these methods using PyTorch (version 1.13.1) and PyTorch Lightning (version 1.5.0) software libraries. We publish our re-implementations and the accompanying evaluation code on GitHub at this URL: https://github.com/allegro/AlleNoise.

To select the best hyperparameters (see Appendix **A**) for each of the benchmarked algorithms, we performed a tuning process on the *AlleNoise* dataset. We focused on maximizing the fraction of correct clean examples $\texttt{correct}_{val}^{\texttt{clean}}$ within the validation set for two noise types: 15% real-world noise and 15% symmetric noise. The tuning was performed on a single fold selected out of five cross-validation folds, yielding optimal hyperparameter values (**Tab. S1**). We then used these tuned values in all further experiments.

5 RESULTS

The selected methods for learning with noisy labels were found to perform differently on AlleNoise than on several types of synthetic noise. Below we highlight those differences in performance and relate them to the dissimilarities between real-world and synthetic noise.

5.1 Synthetic noise vs *AlleNoise*

The selected methods were compared on the clean dataset, the four types of synthetic noise and on the real-world noise in *AlleNoise* (**Tab. 3**). The accuracy score on the clean dataset did not degrade for any of the evaluated algorithms when compared to the baseline CE. When it comes to the performance on the datasets with symmetric noise, the best method was GJSD, with CCE not too far behind. GJSD increased the accuracy by 1 percentage point (p.p.) over the baseline. For asymmetric noise types, the best method was consistently ELR. It significantly improved the test accuracy in comparison to CE, by 1.34 p.p. on average. Interestingly, some methods deteriorated the test accuracy. CT+ was worse than the baseline for all synthetic noise types (by 2.67 p.p., 2.00 p.p., 1.74 p.p., 1.93 p.p. for symmetric, pair-flip, nested-flip and matrix-flip noises, respectively), while SPL decreased the results for all types of asymmetric noise (by 3.20 p.p., 3.74 p.p., 4.58 p.p. for pair-flip, nested-flip and matrix-flip noises, respectively). PLS also underperformed for asymmetric noise types. CT+ seems to perform better for noise levels higher than 15% (see Appendix **B**). On *AlleNoise*, we observed nearly no improvement in accuracy for any of the evaluated algorithms, and CT+, PRL and SPL all significantly (>1 p.p.) deteriorated the metric (by 2.24 p.p., 1.33 p.p. and 3.91 p.p., respectively).

5.2 Noise type impacts memorization

To better understand the difference between synthetic noise types and *AlleNoise*, we analyze how the $\texttt{memorized}_{val}^{\texttt{noisy}}$, $\texttt{correct}_{val}^{\texttt{noisy}}$ and $\texttt{correct}_{val}^{\texttt{clean}}$ metrics (see 4.4) evolve over time. Memorization and correctness should be interpreted jointly with test accuracy (**Tab. 3**). Additional plots of training, validation and test accuracy enriching this analysis are included in Appendix **C**.

Synthetic noise types are memorized to a smaller extent than the real-world *AlleNoise* (**Fig. 2a**). For the two

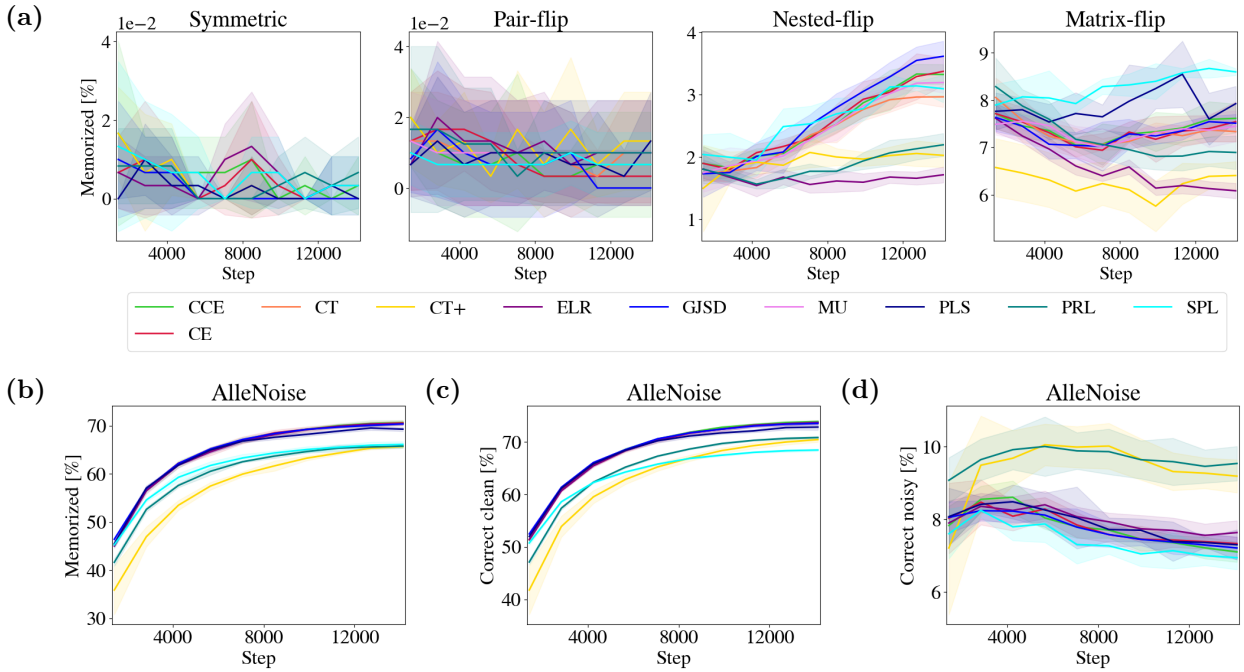

Figure 2: Memorization and correctness metrics as a function of the training step. **(a)** The value of $\texttt{memorized}_{val}$ for synthetic noise types. **(b)** The value of $\texttt{memorized}_{val}$ for *AlleNoise*. **(c)** The value of $\texttt{correct}^{\texttt{clean}}_{val}$ for *AlleNoise*. **(d)** The value of $\texttt{correct}^{\texttt{noisy}}_{val}$ for *AlleNoise*.

simplest synthetic noise types, symmetric and pair-flip, the value of $\texttt{memorized}_{val}$ is negligible (very close to zero). For the other two synthetic noise types, nested-flip and matrix-flip, memorization is still low (2-8%), but there are clearly visible differences between the benchmarked methods. While ELR, CT+ and PRL all keep the value of $\texttt{memorized}^{\texttt{noisy}}_{val}$ low for both nested-flip and matrix-flip noise types, it is only ELR that achieves test accuracy higher than the baseline.

However, for *AlleNoise*, the situation is completely different. All the training methods display increasing $\texttt{memorized}_{val}$ values throughout the training, up to 70% (**Fig. 2b**). PRL, SPL and CT+ give lower memorization than the other methods, but this is not reflected in higher accuracy. While these methods correct some of the errors on noisy examples, as measured by $\texttt{correct}^{\texttt{noisy}}_{val}$ (**Fig. 2d**), they display $\texttt{correct}^{\texttt{clean}}_{val}$ lower than other tested approaches (**Fig. 2c**), and thus overall they achieve low accuracy.

These results show that reducing memorization is necessary to create noise-robust classifiers. In this context, it is clear that *AlleNoise*, with its real-world instance-dependent noise distribution, is a challenge for the existing methods.

5.3 Noise distribution

To get even more insight into why the real-world noise in *AlleNoise* is more challenging than synthetic noise types, we analyzed the class distribution within our dataset. For synthetic noise types, there are very few highly-corrupted categories (**Fig. 3a**, **Fig. 3b**, **Fig. S4**). On the other hand, for *AlleNoise*, there is a significant number of such categories (**Fig. 3a**, **Fig. 3c**). The baseline model test accuracy is much lower for these classes than for other, less corrupted, categories. The set of those highly-corrupted classes is heavily populated by the following:

- *Specialized categories* that can be easily mistaken for a more generic category. For example, items belonging to the class *safety shoes* are frequently listed in categories *derby shoes*, *ankle boots* or *other*. In these cases, the model encounters a high true noise level, as it sees a large number of mislabeled instances with very few correctly labeled ones, which hinders its ability to learn accurate class associations (**Fig. 3c**).
- *Archetypal categories* that are considered the most representative examples of a broader parent category. For instance, car tires are most frequently listed in *Summer tires* even when they actually should belong to *All-season tires* or other spe-

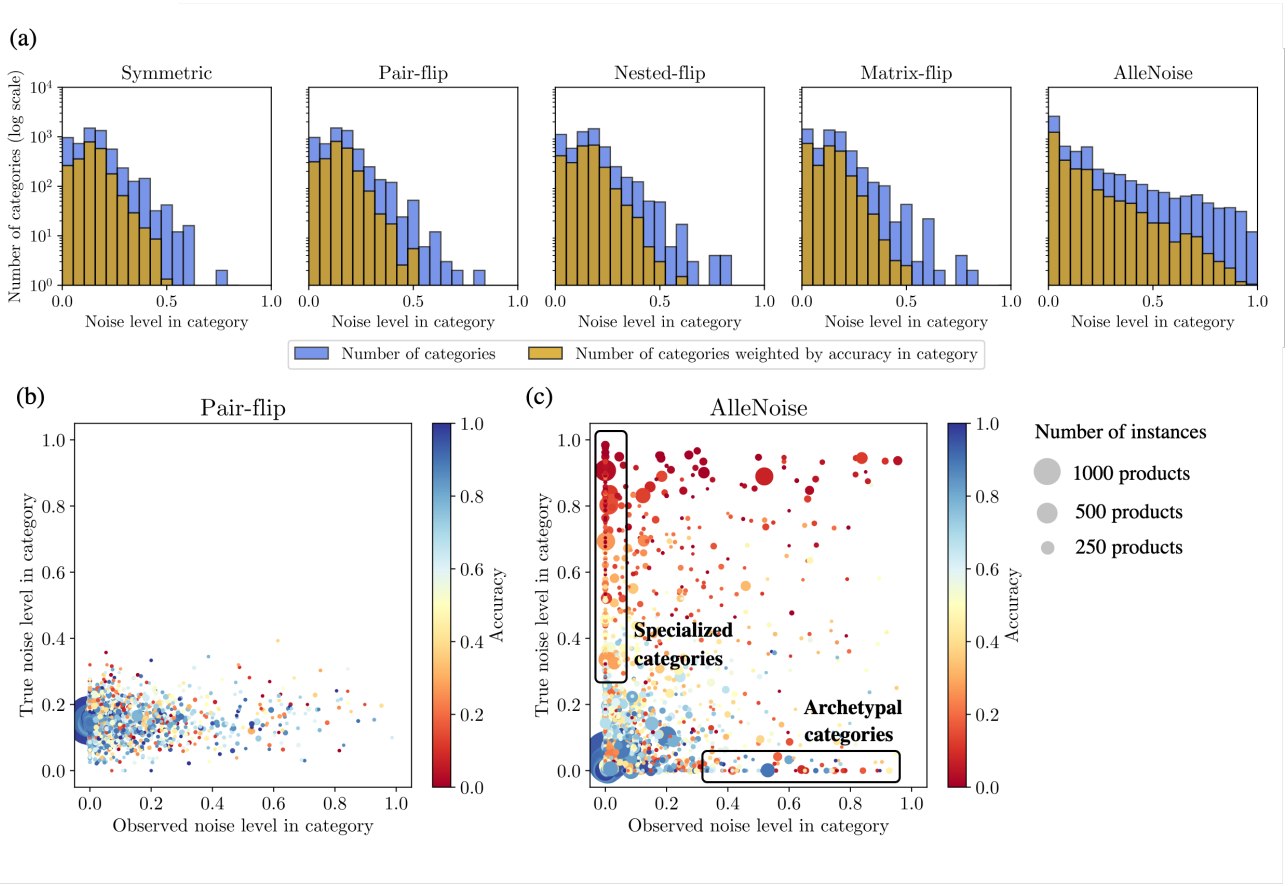

Figure 3: Noise distribution and patterns of wrong predictions across different noise types. **(a)** Noise level distribution over target categories (blue bars) shows that *AlleNoise* has a substantial fraction of classes with noise level over 0.5, contrary to synthetic noise. The same distribution multiplied by per-bin macro accuracy (yellow bars) shows that those specialized categories are particularly difficult to predict correctly. **(b)** Scatter plot of true noise level versus observed noise level in each category for synthetic pair-flip noise. Marker color represents accuracy, and marker size reflects category size. True noise levels are concentrated around 15%. Pair-flip noise does not model distinct specialized or archetypal categories. The plot includes only categories with at least 25 products. **(c)** Scatter plot for real-world *AlleNoise*, highlighting the presence of many specialized and archetypal categories. Accuracy in specialized categories is negatively correlated with the true noise level. A significant number of categories exhibit both high true noise and high observed noise levels.

cialized categories. In this scenario, the model encounters a high observed noise level, as it sees a large number of specialized items mislabeled as the archetypal class, distorting its learned representation of that category

We hypothesize that these categories, with their respective high true and observed noise levels, are the primary contributors to the models' poor performance on *AlleNoise* - an issue not present in synthetic noise types, which fail to model the complexity of specialized and archetypal categories (**Fig. 3b**, **Fig. S4**). This challenge is further compounded by the inability of the benchmarked methods to consistently improve accuracy across categories with varying levels of both synthetic and real-world noise. See Appendix **D** for further discussion.

6 DISCUSSION

Our experiments show that the real-world noise present in *AlleNoise* is a challenging task for existing methods for learning with noisy labels. We hypothesize that the main challenges for these methods stem from two major features of *AlleNoise*: 1) real-world, instance dependent noise distribution, 2) relatively large number of categories with class imbalance and long tail. While previous works have investigated challenges 1) (Wei et al., 2022) and 2) (Wu et al., 2023), this paper combines both in a single dataset and evaluation study, while

also applying them to text data. We hope that making *AlleNoise* available publicly will spark new method development, especially in directions that would address the features of our dataset.

Based on our experiments, we make several interesting observations. The methods that rely on removing examples from within a batch perform noticeably worse than other approaches. We hypothesize that this is due to the large number of classes and the unbalanced distribution of their sizes (especially the long tail of underrepresented categories) in *AlleNoise* - by removing samples, we lose important information that is not recoverable. This is supported by the fact that such noise filtration methods excel on simple benchmarks like CIFAR-10, which all have a completely different class distribution. The methods that originally relied on image augmentations (i.e. PLS) also do not exhibit good performance in our text-oriented dataset, even when applied to synthetic noise types. We could not replicate the image augmentations one-to-one, so we augmented our data by introducing random word swaps, word crops and word splits (see Appendix **A**). It is possible that the change in modality and the required adjustment of augmentation strategies affected the effectiveness of the method.

In order to mitigate the noise in *AlleNoise*, a more sophisticated and tailor-made approach is necessary. A promising direction seems to be the one presented by ELR. While for the real-world noise it did not increase the results above the baseline CE, it was the best algorithm for class-dependent noise types. The outstanding performance of ELR might be attributed to its target smoothing approach. The use of such soft labels may be particularly adequate to extreme classification scenarios where some of the classes are semantically close. Extending this idea to include an instance-dependent component may lead to an algorithm robust to the real-world noise in *AlleNoise*. Furthermore, based on the results of the memorization metric, it is evident that this realistic noise pattern needs to be tackled in a different way than synthetic noise. With the clean labels published as a part of *AlleNoise*, we enable researchers to further explore the issue of memorization in the presence of real-world instance-dependent noise.

7 LIMITATIONS

Our dataset presents several notable characteristics and limitations. It includes ∼500,000 product titles from over 5,000 categories, sampled to reflect the broader product catalog while controlling for the tangible 15% noise level. However, the dataset's focus on a Polish marketplace might limit its diversity and applicability to other regions, particularly outside the EU. The specialized nature of e-commerce text might not be completely transferable to other NLP domains. Moreover, the translation accuracy of our in-house scale neural machine translation system remains imperfect, which has an impact on classification accuracy (see Appendix **G**). Despite these challenges, the *AlleNoise* dataset is a useful resource for benchmarking text classification models, especially with its known noise level, distinguishing it from other e-commerce datasets (Hou et al., 2024; Lin et al., 2019; Akritidis, Fevgas, Bozanis, and Makris, 2020; Akritidis, Fevgas, and Bozanis, 2018). For an extended discussion on limitations of AlleNoise please refer to Appendix **F**.

8 CONCLUSIONS AND FUTURE WORK

In this paper, we presented a new dataset for the evaluation of methods for learning with noisy labels. Our dataset, *AlleNoise*, contains a real-world instance-dependent noise distribution, with both clean and noisy labels. It provides a large-scale classification problem, and unlike most previously available datasets in the field of learning from noisy labels, features textual data. We performed an evaluation of established noise-mitigation methods, which showed quantitatively that these approaches are not enough to alleviate the noise in our dataset. With *AlleNoise*, we hope to jump-start the development of new robust classifiers that would be able to handle demanding, real-world instance-dependent noise, reducing errors in practical applications of text classifiers.

The scope of this paper is limited to BERT-based classifiers. While Large Language Models (LLMs) exhibit potential for denoising in zero-shot or in-context classification, applying LLMs to tasks with thousands of labels is non-trivial due to challenges like handling long-tail label distributions and prompt constraints (Jung et al., 2023; Zhou et al., 2024). As *AlleNoise* includes clean label names in addition to noisy labels, it could be used to benchmark Large Language Models in few-shot or in-context learning scenarios. We leave this as a future research direction.

Acknowledgements

We thank Mikołaj Koszowski for his help with translating the product titles. We also thank Karol Jędrzejewski for his help in pin-pointing the location of appropriate data records in Allegro data warehouse.

References

Akritidis, Leonidas, Athanasios Fevgas, and Panayiotis Bozanis (Nov. 2018). "Effective Products Categorization with Importance Scores and Morphological Analysis of the Titles". In: *2018 IEEE 30th International Conference on Tools with Artificial Intelligence (ICTAI)*. ISSN: 2375-0197, pp. 213–220. DOI: 10.1109/ICTAI.2018.00041. URL: https://ieeexplore.ieee.org/document/8576039.

Akritidis, Leonidas, Athanasios Fevgas, Panayiotis Bozanis, and Christos Makris (Oct. 2020). "A self-verifying clustering approach to unsupervised matching of product titles". en. In: *Artificial Intelligence Review* 53.7, pp. 4777–4820. ISSN: 1573-7462. DOI: 10.1007/s10462-020-09807-8. URL: https://doi.org/10.1007/s10462-020-09807-8.

Albert, Paul et al. (2022). *Is your noise correction noisy? PLS: Robustness to label noise with two stage detection*. arXiv: 2210.04578 [cs.CV]. URL: https://arxiv.org/abs/2210.04578.

Algan, Görkem and İlkay Ulusoy (Mar. 2020). "Label Noise Types and Their Effects on Deep Learning". In: *arXiv:2003.10471 [cs]*. arXiv: 2003.10471. URL: http://arxiv.org/abs/2003.10471.

Arpit, Devansh et al. (July 2017). *A Closer Look at Memorization in Deep Networks*. arXiv:1706.05394 [cs, stat]. DOI: 10.48550/arXiv.1706.05394. URL: http://arxiv.org/abs/1706.05394.

Berthon, Antonin et al. (Feb. 2021). "Confidence Scores Make Instance-dependent Label-noise Learning Possible". In: *arXiv:2001.03772 [cs, stat]*. arXiv: 2001.03772. URL: http://arxiv.org/abs/2001.03772.

Bhatia, K. et al. (2016). *The extreme classification repository: Multi-label datasets and code*. URL: http://manikvarma.org/downloads/XC/XMLRepository.html.

Chen, Pengfei et al. (Dec. 2020). *Beyond Class-Conditional Assumption: A Primary Attempt to Combat Instance-Dependent Label Noise*. en. Number: arXiv:2012.05458 arXiv:2012.05458 [cs]. URL: http://arxiv.org/abs/2012.05458.

Conneau, Alexis et al. (2019). "Unsupervised Cross-lingual Representation Learning at Scale". In: *CoRR* abs/1911.02116. arXiv: 1911.02116. URL: http://arxiv.org/abs/1911.02116.

Deng, Jia et al. (2009). "Imagenet: A large-scale hierarchical image database". In: *2009 IEEE conference on computer vision and pattern recognition*. Ieee, pp. 248–255.

Deng, Li (2012). "The mnist database of handwritten digit images for machine learning research". In: *IEEE Signal Processing Magazine* 29.6, pp. 141–142.

Englesson, Erik and Hossein Azizpour (Oct. 2021). "Generalized Jensen-Shannon Divergence Loss for Learning with Noisy Labels". In: *arXiv:2105.04522 [cs, stat]*. arXiv: 2105.04522. URL: http://arxiv.org/abs/2105.04522.

Frenay, Benoit and Michel Verleysen (May 2014). "Classification in the Presence of Label Noise: A Survey". en. In: *IEEE Transactions on Neural Networks and Learning Systems* 25.5, pp. 845–869. ISSN: 2162-237X, 2162-2388. DOI: 10.1109/TNNLS.2013.2292894. URL: http://ieeexplore.ieee.org/document/6685834/.

Goldberger, Jacob and Ehud Ben-Reuven (2017). "Training deep neural-networks using a noise adaptation layer". In: *International Conference on Learning Representations*. URL: https://openreview.net/forum?id=H12GRgcxg.

Gu, Keren et al. (Oct. 2021). "An Instance-Dependent Simulation Framework for Learning with Label Noise". In: *arXiv:2107.11413 [cs]*. arXiv: 2107.11413. URL: http://arxiv.org/abs/2107.11413.

Han, Bo et al. (Oct. 2018). "Co-teaching: Robust Training of Deep Neural Networks with Extremely Noisy Labels". In: *arXiv:1804.06872 [cs, stat]*. arXiv: 1804.06872. URL: http://arxiv.org/abs/1804.06872.

Hedderich, Michael A., David Adelani, et al. (Nov. 2020). "Transfer Learning and Distant Supervision for Multilingual Transformer Models: A Study on African Languages". In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Ed. by Bonnie Webber et al. Online: Association for Computational Linguistics, pp. 2580–2591. DOI: 10.18653/v1/2020.emnlp-main.204. URL: https://aclanthology.org/2020.emnlp-main.204.

Hedderich, Michael A., Dawei Zhu, and Dietrich Klakow (2021). *Analysing the Noise Model Error for Realistic Noisy Label Data*. arXiv: 2101.09763 [cs.LG].

Hou, Yupeng et al. (Mar. 2024). *Bridging Language and Items for Retrieval and Recommendation*. arXiv:2403.03952 [cs]. DOI: 10.48550/arXiv.2403.03952. URL: http://arxiv.org/abs/2403.03952.

Jindal, Ishan, Matthew Nokleby, and Xuewen Chen (2017). *Learning Deep Networks from Noisy Labels with Dropout Regularization*. arXiv: 1705.03419 [cs.CV].

Jindal, Ishan, Daniel Pressel, et al. (2019). *An Effective Label Noise Model for DNN Text Classification*. arXiv: 1903.07507 [cs.LG].

Jung, Taehee et al. (2023). "Cluster-guided label generation in extreme multi-label classification". In: *EACL 2023*. URL: https://www.amazon.science/publications/cluster-guided-label-generation-in-extreme-multi-label-classification.

Krishna, Ranjay A. et al. (May 2016). "Embracing Error to Enable Rapid Crowdsourcing". In: *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. CHI'16. ACM. DOI: 10.1145/2858036.2858115. URL: http://dx.doi.org/10.1145/2858036.2858115.

Krizhevsky, Alex (2009). "Learning Multiple Layers of Features from Tiny Images". In: URL: https://api.semanticscholar.org/CorpusID:18268744.

Kumar, M., Benjamin Packer, and Daphne Koller (2010). "Self-Paced Learning for Latent Variable Models". In: *Advances in Neural Information Processing Systems*. Vol. 23. Curran Associates, Inc. URL: https://papers.nips.cc/paper/2010/hash/e57c6b956a6521b28495f2886ca0977a-Abstract.html.

Li, Wen et al. (Aug. 2017). "WebVision Database: Visual Learning and Understanding from Web Data". In: *arXiv:1708.02862 [cs]*. arXiv: 1708.02862. URL: http://arxiv.org/abs/1708.02862.

Lin, Yiu-Chang et al. (Sept. 2019). "A Dataset and Baselines for e-Commerce Product Categorization". In: *Proceedings of the 2019 ACM SIGIR International Conference on Theory of Information Retrieval*. ICTIR '19. New York, NY, USA: Association for Computing Machinery, pp. 213–216. ISBN: 9781450368810. DOI: 10.1145/3341981.3344237. URL: https://doi.org/10.1145/3341981.3344237.

Liu, Bo et al. (Oct. 2022). "Noise Learning for Text Classification: A Benchmark". In: *Proceedings of the 29th International Conference on Computational Linguistics*. Ed. by Nicoletta Calzolari et al. Gyeongju, Republic of Korea: International Committee on Computational Linguistics, pp. 4557–4567. URL: https://aclanthology.org/2022.coling-1.402.

Liu, Boyang et al. (Feb. 2021). "Learning Deep Neural Networks under Agnostic Corrupted Supervision". In: *arXiv:2102.06735 [cs, stat]*. arXiv: 2102.06735. URL: http://arxiv.org/abs/2102.06735.

Liu, Sheng et al. (Oct. 2020). "Early-Learning Regularization Prevents Memorization of Noisy Labels". In: *arXiv:2007.00151 [cs, stat]*. arXiv: 2007.00151. URL: http://arxiv.org/abs/2007.00151.

Ma, Edward (2019). *NLP Augmentation*. https://github.com/makcedward/nlpaug.

Maas, Andrew L. et al. (June 2011). "Learning Word Vectors for Sentiment Analysis". In: *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. Portland, Oregon, USA: Association for Computational Linguistics, pp. 142–150. URL: http://www.aclweb.org/anthology/P11-1015.

Netzer, Yuval et al. (2011). "Reading Digits in Natural Images with Unsupervised Feature Learning". In: *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*. URL: http://ufldl.stanford.edu/housenumbers/nips2011_housenumbers.pdf.

Nguyen, Huy and Devashish Khatwani (May 2022). "Robust Product Classification with Instance-Dependent Noise". In: *Proceedings of The Fifth Workshop on e-Commerce and NLP (ECNLP 5)*. Dublin, Ireland: Association for Computational Linguistics, pp. 171–180. URL: https://aclanthology.org/2022.ecnlp-1.20.

Ni, Jianmo, Jiacheng Li, and Julian McAuley (Nov. 2019). "Justifying Recommendations using Distantly-Labeled Reviews and Fine-Grained Aspects". In: *Proceedings of the 2019 EMNLP and the 9th IJCNLP*. Ed. by Kentaro Inui et al. Hong Kong, China: Association for Computational Linguistics, pp. 188–197. DOI: 10.18653/v1/D19-1018. URL: https://aclanthology.org/D19-1018.

Northcutt, Curtis G., Anish Athalye, and Jonas Mueller (Nov. 2021). "Pervasive Label Errors in Test Sets Destabilize Machine Learning Benchmarks". In: *arXiv:2103.14749 [cs, stat]*. arXiv: 2103.14749. URL: http://arxiv.org/abs/2103.14749.

Patrini, Giorgio et al. (Mar. 2017). *Making Deep Neural Networks Robust to Label Noise: a Loss Correction Approach*. arXiv:1609.03683 [cs, stat]. DOI: 10.48550/arXiv.1609.03683. URL: http://arxiv.org/abs/1609.03683.

Rolnick, David et al. (Feb. 2018). *Deep Learning is Robust to Massive Label Noise*. en. arXiv:1705.10694 [cs]. URL: http://arxiv.org/abs/1705.10694.

Schmarje, Lars et al. (2022). *Is one annotation enough? A data-centric image classification benchmark for noisy and ambiguous label estimation*. arXiv: 2207.06214 [cs.CV].

Song, Hwanjun, Minseok Kim, and Jae-Gil Lee (2019). "SELFIE: Refurbishing Unclean Samples for Robust Deep Learning". In: *ICML*.

Song, Hwanjun, Minseok Kim, Dongmin Park, et al. (Mar. 2022). *Learning from Noisy Labels with Deep Neural Networks: A Survey*. arXiv:2007.08199 [cs, stat]. DOI: 10.48550/arXiv.2007.08199. URL: http://arxiv.org/abs/2007.08199.

Tan, Liling, Maggie Yundi Li, and Stanley Kok (July 2020). "E-Commerce Product Categorization via Machine Translation". In: *ACM Trans. Manage. Inf. Syst.* 11.3. ISSN: 2158-656X. DOI: 10.1145/3382189. URL: https://doi.org/10.1145/3382189.

Vaswani, Ashish et al. (2017). "Attention is All you Need". In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon et al. Vol. 30. Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Wang, Alex et al. (2019). *GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding*. arXiv: 1804.07461 [cs.CL].

Wei, Jiaheng et al. (Mar. 2022). "Learning with Noisy Labels Revisited: A Study Using Real-World Human Annotations". In: *arXiv:2110.12088 [cs, stat]*. arXiv: 2110.12088. URL: http://arxiv.org/abs/2110.12088.

Wu, Tingting et al. (July 2023). "NoisywikiHow: A Benchmark for Learning with Real-world Noisy Labels in Natural Language Processing". In: *Findings of the Association for Computational Linguistics: ACL 2023*. Ed. by Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki. Toronto, Canada: Association for Computational Linguistics, pp. 4856–4873. DOI: 10.18653/v1/2023.findings-acl.299. URL: https://aclanthology.org/2023.findings-acl.299.

Xia, Xiaobo et al. (Dec. 2020). "Part-dependent Label Noise: Towards Instance-dependent Label Noise". In: *arXiv:2006.07836 [cs, stat]*. arXiv: 2006.07836. URL: http://arxiv.org/abs/2006.07836.

Xiao, Tong et al. (2015). "Learning from massive noisy labeled data for image classification". In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2691–2699. DOI: 10.1109/CVPR.2015.7298885.

Yu, Xingrui et al. (May 2019). "How does Disagreement Help Generalization against Label Corruption?" In: *arXiv:1901.04215 [cs, stat]*. arXiv: 1901.04215. URL: http://arxiv.org/abs/1901.04215.

Zhang, Bryan et al. (2023). "Improve machine translation in e-commerce multilingual search with contextual signal from search sessions". In: *SIGIR 2023 Workshop on eCommerce*. URL: https://www.amazon.science/publications/improve-machine-translation-in-e-commerce-multilingual-search-with-contextual-signal-from-search-sessions.

Zhang, Chiyuan et al. (Feb. 2017). *Understanding deep learning requires rethinking generalization*. arXiv:1611.03530 [cs]. DOI: 10.48550/arXiv.1611.03530. URL: http://arxiv.org/abs/1611.03530.

– (Mar. 2021). "Understanding deep learning (still) requires rethinking generalization". en. In: *Communications of the ACM* 64.3, pp. 107–115. ISSN: 0001-0782, 1557-7317. DOI: 10.1145/3446776. URL: https://dl.acm.org/doi/10.1145/3446776.

Zhang, Hongyi et al. (Apr. 2018). "mixup: Beyond Empirical Risk Minimization". In: *arXiv:1710.09412 [cs, stat]*. arXiv: 1710.09412. URL: http://arxiv.org/abs/1710.09412.

Zhou, Chuang et al. (Nov. 2024). "QUEST: Efficient Extreme Multi-Label Text Classification with Large Language Models on Commodity Hardware". In: *Findings of the Association for Computational Linguistics: EMNLP 2024*. Ed. by Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen. Miami, Florida, USA: Association for Computational Linguistics, pp. 3929–3940. DOI: 10.18653/v1/2024.findings-emnlp.226. URL: https://aclanthology.org/2024.findings-emnlp.226.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Not Applicable]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Not Applicable]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes] - We include the source code to replicate the benchmark results in the supplemental material.

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Not Applicable]
   (b) Complete proofs of all theoretical results. [Not Applicable]
   (c) Clear explanations of any assumptions. [Not Applicable]

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes] - We provide the code and data in the supplemental material.
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes] - See Section 4.3 and Section 4.5.
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes] - We have used 5 split cross-validation to estimate the variance in all experiments. See Section 4.3.
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes] - We used Google Cloud Platform virtual machines with NVIDIA A100 GPUs. See Section 4.3.

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes] - The only existing asset used in the study is the XLMRoBERTa backbone, referenced in Section 4.3.
   (b) The license information of the assets, if applicable. [Yes] - We specify the licence of our dataset in the supplementary data sheet.
   (c) New assets either in the supplemental material or as a URL, if applicable. [Yes] - We include the dataset in the supplemental material.
   (d) Information about consent from data providers/curators. [Yes] - We have not collected any data that would require user consent. It was required to get legal approval from Allegro in order to publish our data, which is stated in the supplementary data sheet.
   (e) Discussion of sensitive content if applicable, e.g., personally identifiable information or offensive content. [Yes] - We described data post-processing in Section 3.3, i.e. filtering out potentially offensive product categories.

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable] - The data in our dataset comes from pre-existing internal logs of Allegro.com.
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable] - The data in our dataset comes from pre-existing internal logs of Allegro.com.
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Yes] - While the data in our dataset comes from pre-existing internal logs of Allegro.com, we do state in the supplementary data sheet the guaranteed wage that human domain experts who originally verified the data were compensated with.

AlleNoise – large-scale text classification benchmark dataset with real-world label noise: Supplementary Materials

A Implementation details

Self-Paced Learning

The Self-Paced Learning (SPL) (Kumar et al., 2010) method sets a threshold λ value for the loss and all examples with loss larger than λ are skipped, since they are treated as hard to learn (because they are possibly noisy). After each training epoch, the threshold is increased by some constant multiplier. For simplification, we adjusted SPL in the following manner.

We set a parameter τ_{SPL}, which controls the percentage of samples with the highest loss within a batch that are excluded. The value of τ_{SPL} should be equal to the noise level present in the training dataset. As such, at each step, we exclude a set percentage of potentially noisy examples, thus reducing the impact of label noise on the training process. We keep the value of τ_{SPL} constant throughout the training.

Provably Robust Learning

The Provably Robust Learning (PRL) (Boyang Liu et al., 2021) algorithm works in a similar manner to SPL. We follow the authors by introducing the τ_{PRL} parameter, which controls the percentage of samples excluded from each training batch on the basis of their gradient norm. Specifically, $\tau_{PRL}\%$ of samples with highest gradient norm are omitted, while the rest is used to update model parameters. The value of τ_{PRL} should be equal to the noise level in the training dataset.

Clipped Cross-Entropy

Since our implementation of SPL doesn't have a hard loss threshold, we introduce a simple Clipped Cross-Entropy (CCE) baseline to check the effectiveness of such an approach. The CCE method checks if the loss is greater than some threshold λ_{CCE}. If so, the loss is clipped to that value. Otherwise, it is left unchanged. Thus, we always use all training samples, but the impact of label noise is alleviated by clipping the loss.

Early Learning Regularization

For Early Learning Regularization (ELR) (S. Liu et al., 2020), we followed the implementation published by the authors. We compute the softmax probabilities for each sample in a batch and clamp them, then compute the soft targets via temporal ensembling and use these targets in the loss function calculation. Our implementation includes one step not present in the publication text - softmax probability clamping in range $[\epsilon, 1-\epsilon]$, where ϵ is a clamp margin parameter. Aside from this, we use the β target momentum and λ_{ELR} regularization parameters just as they were presented by the authors.

Generalized Jensen-Shannon Divergence Loss

The Generalized Jensen-Shannon Divergence (GJSD) (Englesson et al., 2021) loss function is a generalization of Cross-Entropy (CE) and Mean Absolute Error (MAE) losses. We follow the implementation provided by the authors, in which we use the M parameter to set the number of averaged distributions and the π parameter to adjust the weight between CE and MAE. While the authors share separate implementations for GJSD with and without consistency regularization, we implement it as a toggle to make the code more uniform. Since consistency regularization requires data augmentation and the GJSD authors described only augmentations for the image domain, we implemented several textual augmentations of our own: random token dropping, consecutive token dropping, random token swapping. However, in our experiments, we have kept consistency regularization turned off due to its detrimental effect on model convergence and test accuracy.

Co-teaching

While the methods described above modified the loss function in various ways, Co-teaching (CT) (Han et al., 2018) works in a different manner. It requires optimizing two sets of model parameters at the same time. As such, following the algorithm described by the authors, we implemented a custom model class, which manages the update of these two sets of weights and the exchange of low-loss examples at each optimization step. We keep the parameters k and τ_{CT}, to control the starting epoch for CT and the noise level (i.e. the percentage of low-loss examples that are exchanged between networks), respectively.

Co-teaching+

For Co-teaching+ (CT+) (Yu et al., 2019), we again adhere to the algorithm described by the authors. We use the same implementation framework as for CT, adjusting only the sample selection mechanism to look within examples for which there is disagreement between the two networks. Following the advice in the publication text, we use the *recommended* update strategy for the fraction of instances to select, which is calculated based on the epoch number, as well as parameters k and τ_{CT+}.

Table S1: Hyperparameter values for all benchmarked methods, selected through a tuning procedure.

Method	Hyperparameters	Selected values
SPL	τ_{SPL}	equal to noise level
PRL	τ_{PRL}	equal to noise level
ELR	$\epsilon, \beta, \lambda_{ELR}$	1e-5, 0.6, 2
CCE	λ_{CCE}	9.5
MU	α, r_{MU}	0.1, 0.1
GJSD	M, π	2, 0.001
CT	k, τ_{CT}	8, equal to noise level
CT+	k, τ_{CT+}	8, equal to noise level
PLS	k_{warm}, h_{proj}	8, 512

Mixup

The Mixup (MU) (H. Zhang et al., 2018) technique keeps the loss function (CE) and the hyperparameters of the baseline model unchanged, only augmenting the training data during the training procedure. We use in-batch augmentation, fixed per-batch mixing magnitude sampled from $Beta(\alpha, \alpha)$ (where α is provided as input), and the mixed pairs are sampled without replacement from that distribution. Since we cannot mix input in the same way as for images, we implemented in-batch augmentation for logits. In addition, we also keep the r_{MU} ratio parameter, to adjust the percentage of the batch size which is taken for augmentation in MU. Note: our hyperparameter tuning procedure resulted in setting both α and r_{MU} to low values (**Tab. S1**), contrary to what is recommended by the authors.

Pseudo-Label Selection

For the Pseudo-Label Selection (Albert et al., 2022) algorithm, we use the code provided by the authors. The method relies on image augmentations during training. To adapt these transformations to text modality, we utilize the *nlpaug* (Ma, 2019) Python package. For weak augmentations, we use random word swaps with probability 0.3. For strong augmentation, we again apply random word swaps, followed by random word crops and random word splits, all with probability 0.3. We tune two hyperparameters: the number of epochs that are trained with standard cross-entropy k_{warm}, and the size of the contrastive projection layer h_{proj}.

B Results of experiments with higher noise level

For completeness, we evaluate the accuracy for all methods on datasets with 40% synthetic noise (**Tab. S3**). The best methods for this noise level are the same as for the case of 15% noise: for symmetric noise, GJSD is the

Table S2: Noise types and datasets used for original evaluation of each benchmarked method.

Method	Noise types	Datasets
ELR	symmetric, asymmetric pair-flip	CIFAR10, CIFAR100
MU	symmetric	CIFAR10
CT	symmetric, asymmetric pair-flip	MNIST, CIFAR10, CIFAR100
CT+	symmetric, asymmetric pair-flip	MNIST, CIFAR10, CIFAR100, NEWS, T-ImageNet
PRL	symmetric, asymmetric pair-flip	CIFAR10, CIFAR100
SPL	N/A	N/A
GJSD	symmetric[a], asymmetric pair-flip[b], asymmetric nested-flip[c]	CIFAR10[a,b], CIFAR100[a,c]
PLS	symmetric	CIFAR100

best method, while for asymmetric noise types it is ELR. However, it is clear that some methods show more noticeable effect when compared to the baseline for the 40% noise level than for the 15%. While MU and CCU stay close to the baseline results for all noise types and SPL underperforms in all cases, CT consistently gives an improvement over the baseline and CT+ decreases the result for the symmetric noise, but is better than the baseline for asymmetric noise types.

We also plot $\texttt{memorized}_{val}^{\texttt{noisy}}$ for those datasets (**Fig. S1**). For symmetric and pair-flip noise types the memorization for all methods is very low. For nested-flip and matrix-flip it is a bit higher, indicating that these two noise types are more challenging, and thus induce more memorization in the model.

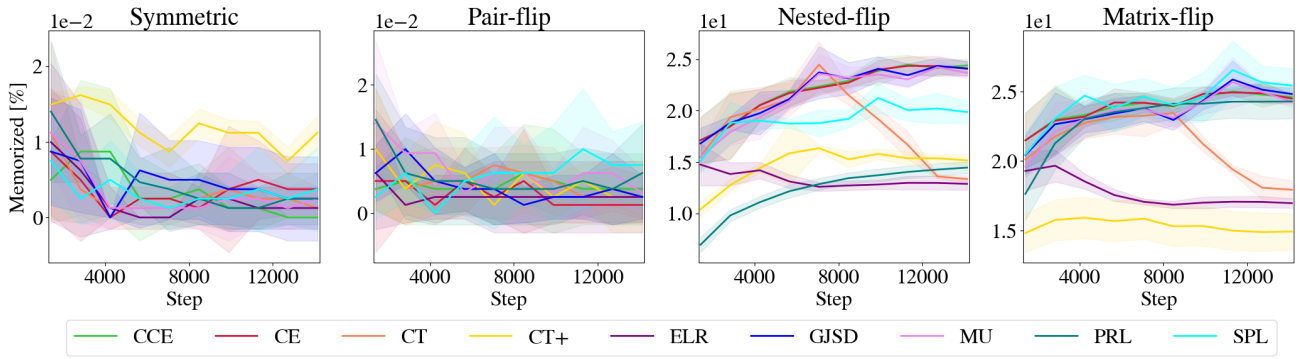

Figure S1: Value of $\texttt{memorized}_{val}$ for different noise types, measured at each training step. In all cases the noise level was set at 40%.

C Training, validation and test accuracy of baseline CE

We measure training, validation and test accuracy of the baseline CE method on clean and noisy datasets (**Fig. S2**). Training accuracy and validation accuracy are measured with observed (potentially noisy) labels, while test accuracy is measured with hidden, clean labels. The real-world noise in *AlleNoise* differs from synthetic noise types. When it comes to training accuracy, the close proximity of *AlleNoise* and clean training curves indicates that the real-world noise distribution is of similar nature to the distribution of clean labels. In other words, the decision boundary for AlleNoise is easier to fit than for synthetic noise types.

For the validation accuracy, we can observe that the memorized knowledge from *AlleNoise* transfers to validation instances with noisy labels, as indicated by the small distance between the validation accuracy curves for *AlleNoise* one for the clean dataset. This is not the case for synthetic noise types. This shows that *AlleNoise* is indeed like

Table S3: Accuracy of the evaluated methods on the clean dataset compared to various noisy datasets with 40% noise level. The noisy datasets include symmetric synthetic noise and asymmetric synthetic noise types: pair-flip, nested-flip, and matrix-flip. * marks cases equivalent to the baseline CE. ↓ marks results significantly worse (>1 p.p.) than the baseline CE. Best results for each noise type are bolded.

	Clean set	Symmetric	Pair-flip	Nested-flip	Matrix-flip
CE	**72.00 ± 0.10**	65.12 ± 0.14	53.09 ± 0.32	50.38 ± 0.33	52.48 ± 0.24
ELR	71.97 ± 0.22	65.03 ± 0.20	**63.41 ± 0.75**	**60.59 ± 0.17**	**60.10 ± 0.26**
MU	71.84 ± 0.16	64.96 ± 0.19	53.52 ± 0.35	50.78 ± 0.25	52.75 ± 0.25
CCE	71.88 ± 0.12	66.41 ± 0.15	53.20 ± 0.32	50.40 ± 0.18	↓50.40 ± 0.18
CT	*72.00 ± 0.10	65.58 ± 0.11	53.28 ± 0.35	56.87 ± 0.29	56.97 ± 0.34
CT+	*72.00 ± 0.10	↓63.20 ± 1.10	57.48 ± 0.24	54.92 ± 0.29	55.59 ± 0.42
PRL	*72.00 ± 0.10	65.31 ± 0.06	57.02 ± 0.19	54.71 ± 0.40	51.51 ± 0.87
SPL	*72.00 ± 0.10	65.49 ± 0.17	↓51.49 ± 0.30	↓41.96 ± 0.25	↓40.90 ± 1.00
GJSD	71.79 ± 0.14	**67.43 ± 0.12**	52.96 ± 0.30	50.39 ± 0.20	52.47 ± 0.16
PLS	71.08 ± 0.14	65.59 ± 0.21	↓48.71 ± 0.66	↓48.26 ± 0.97	↓49.53 ± 1.12

real, clean training data - the model learns from it and generalizes from it just like from clean data. Synthetic noise types do not have this property.

Test accuracy on *AlleNoise* is the lowest of all considered noise types, which shows that the model is misled by the mislabeled training data. Synthetic noisy labels do not mislead the model to such an extent, as even the baseline CE managed to ignore a certain amount of those errors, leading to higher test accuracy than on *AlleNoise*.

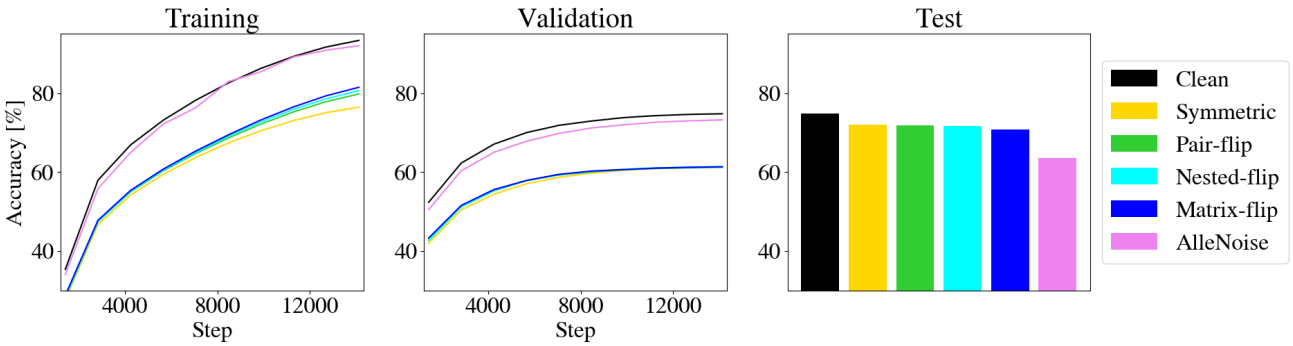

Figure S2: Accuracy of CE for different 15% noise types and the clean dataset. Training and validation accuracy is computed at every validation step during training, test accuracy is computed only for the final model.

D Uneven gains across noise levels and types

Learning from real-world noisy labels in imbalanced datasets like *AlleNoise* is particularly challenging, as it demands consistent accuracy improvements across varying noise levels (**Fig. S4**). To assess the performance of benchmarked methods under diverse noise conditions, we analyzed accuracy gains across different noise levels and types (**Fig. S3**). Our findings reveal the following:

- All methods except CT+ handled symmetric noise effectively up to 15%, with GJSD providing consistent improvements across the entire noise range.
- ELR's poor performance in categories with symmetric noise above 20% neutralized its gains at noise levels below 15%.
- Most methods (ELR, MU, CT, PRL, GJSD) performed well with asymmetric noise between 15% and 25% but struggled with noise below 10%.

- Noise-filtering methods (PRL, SPL, CT, CT+) often undermined the model's performance in categories with asymmetric noise below 15%, likely due to excessive instance removal.

- Gains by MU, CT, CT+, PRL, and GJSD on asymmetric noise between 25% and 40% were balanced out by their poor performance in categories with noise below 10%.

In summary, none of the methods successfully modeled both high and low levels of synthetic noise. This limitation is even more pronounced in the real-world *AlleNoise* dataset, where the majority of instances (56%) are in categories with very low noise levels ($< 5\%$), and there is a long tail (11%) of instances in highly corrupted classes (noise levels $> 40\%$). These characteristics of the real-world noise distribution make the *AlleNoise* dataset a challenge for classification models, setting it apart from both synthetic noise scenarios and other benchmarks in the field of learning with noisy labels.

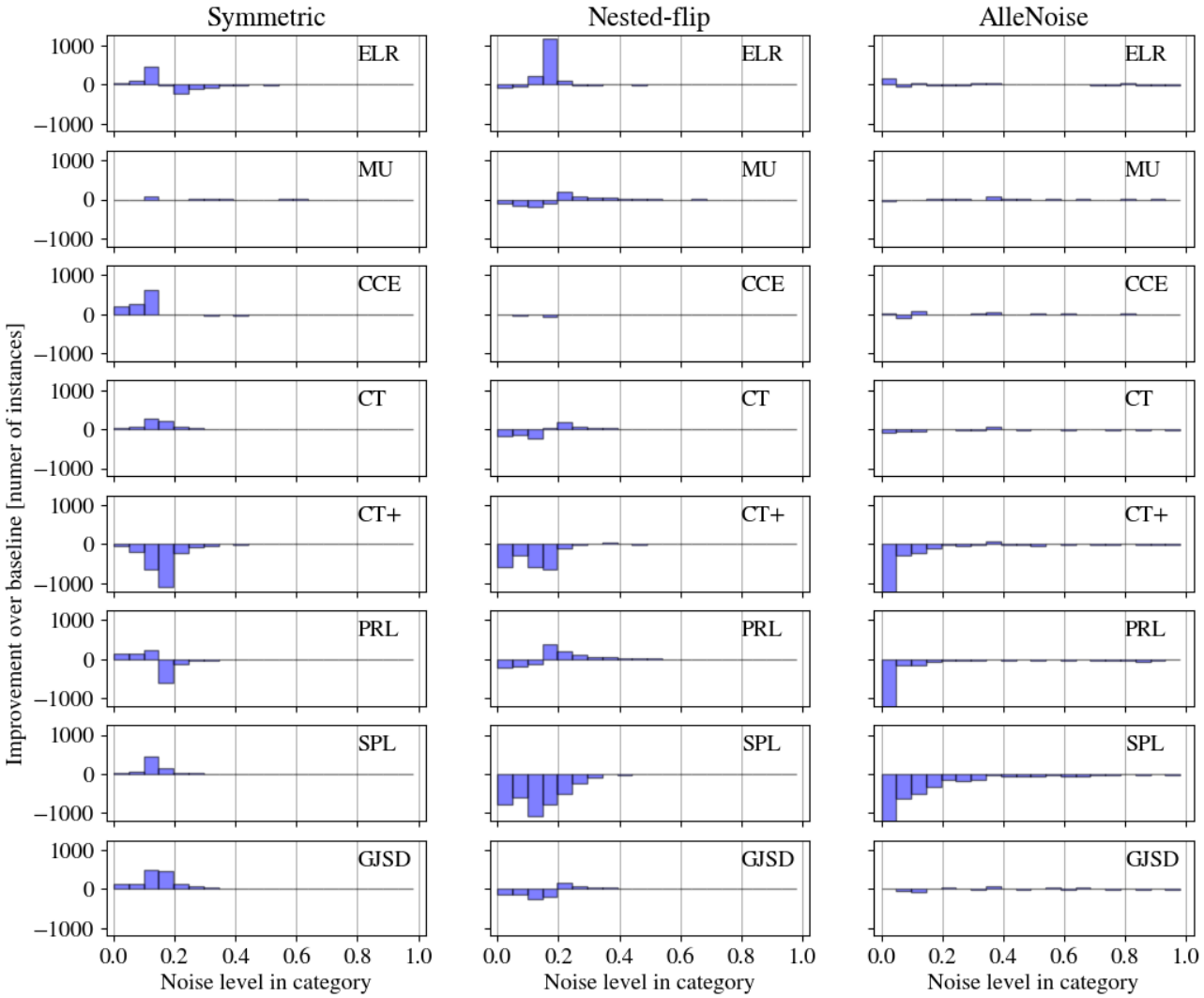

Figure S3: Histograms illustrating the performance of eight learning methods across different noise levels for three noise types (symmetric, nested-flip, and *AlleNoise*). The x-axis represents the noise level within each category, while the y-axis shows the difference in the number of correctly predicted instances between each method and the baseline. Positive bars indicate accuracy gains, while negative bars indicate accuracy losses. The sum of all bars corresponds to the total gain in correctly predicted instances, which determines the final accuracy when normalized by the total number of instances in the dataset.

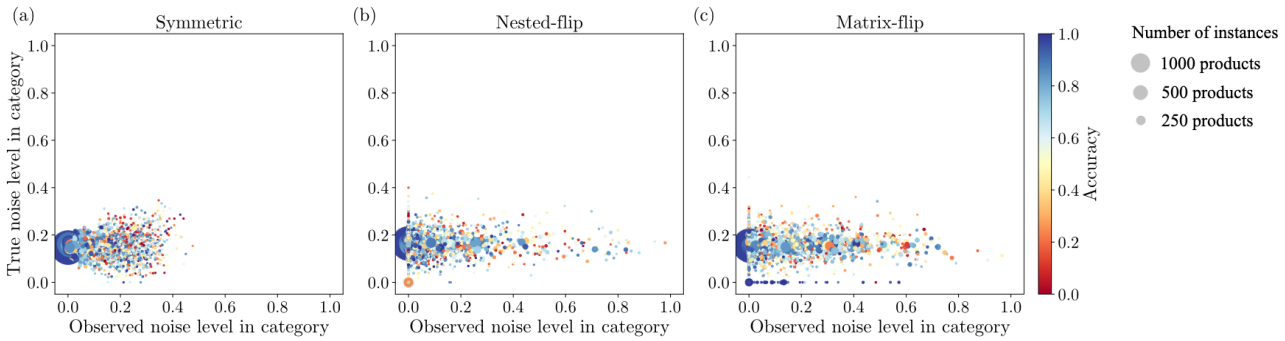

Figure S4: Noise distribution and patterns of wrong predictions across different noise types. Marker color represents accuracy, and marker size reflects category size. True noise levels are concentrated around 15%. **(a)** Symmetric noise fails to capture categories with high observed noise levels. **(b)** Nested-flip noise includes many noise-free categories, similar to real-world noise. **(c)** Matrix-flip noise successfully captures archetypal categories with low true noise and high observed noise. However, all types struggle to model specialized categories. The plot includes only categories with at least 25 products.

E Exploratory Data Analysis

To facilitate the data preprocessing and feature selection stages of future machine learning model development, we provide a detailed analysis of the data in the *AlleNoise* dataset.

E.1 Quality of product titles

During the data sampling stage, we removed instances with titles that hasn't met several quality standards:

- were too short,
- were duplicated,
- were highly repetitive,
- included errors from translation services,
- contained only digits,
- contained derogatory words.

There are no conflicting instances with the same title and different labels, even when lowercase texts are considered. The ratio of special characters (non-alphanumerical characters) is 2%.

The median number of words is 7 (percentiles: $5^{\text{th}} = 4, 95^{\text{th}} = 10$), while the median number of characters is 43 ($5^{\text{th}} = 27, 95^{\text{th}} = 53$) (**Fig. S5**).

E.2 Semantic contents of the dataset

The top-5 popular n-grams in the dataset are: "set", "t shirt", "black", "new" and "white" (**Fig. S6**). Product names are not easily separable in the embedding space, therefore, the classification task should be considered challenging (**Fig. S7**).

E.3 Label distribution

Dataset sampling followed the distribution of products on Allegro, hence the dataset closely mimics the assortment diversity and the proportion of popular to unpopular categories. The "Home and Garden" department has the highest number of categories (994), while "Automotive" has the most instances (106374) (**Tab. S4**). "Home and Garden", "Business and Services" and "Electronics" lead in terms of the amount of noisy labels (**Tab. S5**), when the true underlying category is considered. 17% of mislabeled categories were corrected to a category from a department different than the initial one.

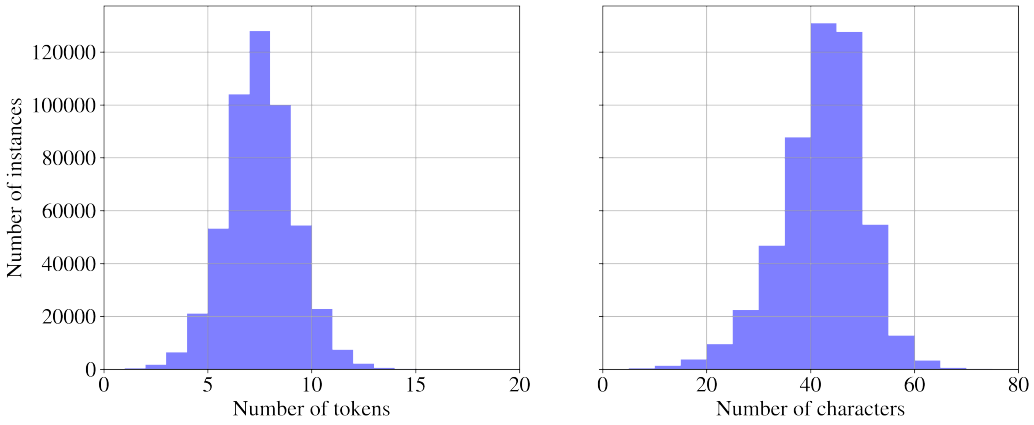

Figure S5: Distribution of product title length in *AlleNoise*. The number of tokens corresponds to the number of words in the product title.

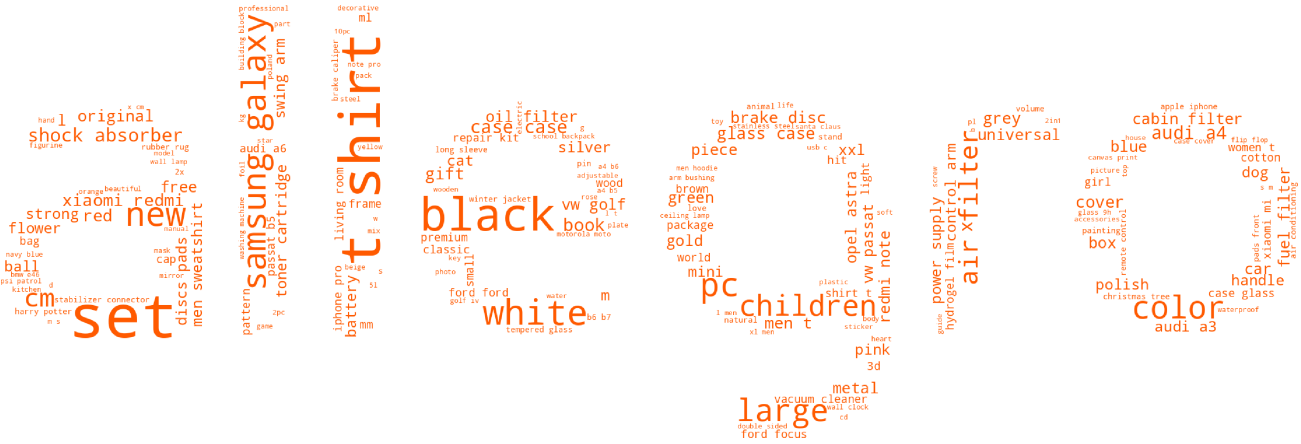

Figure S6: Wordcloud with top-200 n-grams from *AlleNoise* dataset.

While the distribution of categories in the dataset follows the true distribution of products sold in the Allegro marketplace, it poses a challenge due to its highly skewed nature (**Fig. S8**). General, non-specialized, e-commerce platforms typically suffer from a prominent long tail, which presents significant challenges for the automated classification of products into categories. The most populated categories include phone cases, everyday clothing, and home decorations (**Tab. S6**). The least populated categories contain expert tools, smartphone and car models, as well as specific everyday objects (**Tab. S7**).

F Extended discussion of limitations

Selection of Categories Allegro is a general marketplace that represents a wide spectrum of products from various categories and shopping intents. Our dataset comprises over 5,000 categories sampled from nearly 23,000 overall, following the distribution of the Allegro catalog. We undersampled the entire catalog to maintain a manageable dataset size and to control noise levels at around 15%.

Allegro as a Polish Marketplace Since the data originates from a Polish marketplace, the selection of products reflects items typical to this region. The diversity of products catering to minority groups might be limited due to the popularity-based filtering used in the dataset. Additionally, due to EU regulations, the selection of products may not be representative of other marketplaces, such as those originating from the Americas, Asia, or Africa.

E-commerce Domain Our dataset, composed exclusively of e-commerce product names, may not be easily transferable to the broader NLP domain due to its specialized nature. Product titles often include domain-specific jargon, abbreviations, named entities, numbers, codes, and concise text that differs significantly from the more

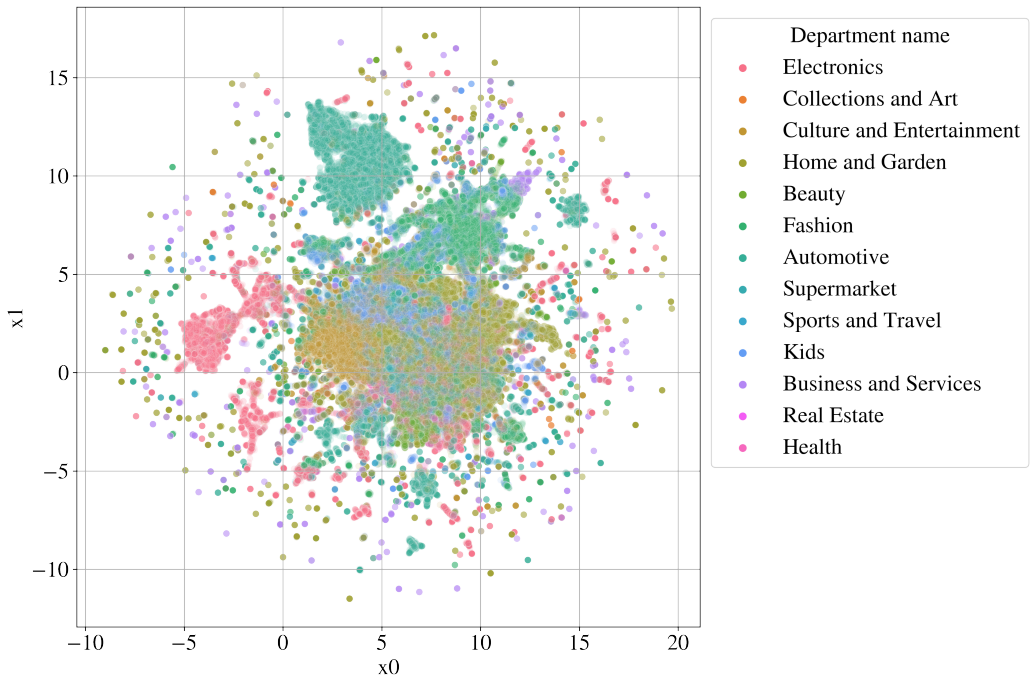

Figure S7: Visualization of the title embedding space. A pretrained USE-v4 model was used as the text embedder. Titles were lowercased before vectorization. A 2D visualization was generated via UMAP dimensionality reduction (number of neighbors: 15, metric: euclidean). Points are colored by department name (meta-category).

diverse and unstructured language found in general NLP tasks, such as web pages, articles, or conversations. This domain-specific focus can limit the generalizability of models trained on this data to other NLP applications.

Machine Translated Content Product titles have been translated using an in-house Neural Machine Translation (NMT) service, maintained by a team of over 30 machine learning specialists, software engineers, and language quality experts, following recent advancements in NMT. However, machine translation systems, often trained on general language corpora, may struggle with the domain-specific jargon, abbreviations, and structured product descriptions common in e-commerce, leading to inaccurate or misleading translations. Additionally, brand names, model numbers, and industry-specific terms may lack direct equivalents in other languages, resulting in translation errors that can compromise the clarity and reliability of the content. We mitigate these issues through model fine-tuning on in-house data, the use of translation glossaries, input data exceptions, and no-translate entity detection, but the model's accuracy is not perfect. More information on the quantitative impact of machine translations can be found in Appendix **G**.

Malicious Content The product database is maintained daily by expert category managers to detect any malicious behavior on the platform, such as illegitimate, disrespectful, or offensive products, personally identifiable information, derogatory language, etc. To the best of our knowledge, the dataset should be free from malicious content; however, we did not conduct extensive annotation in this regard.

Intended Use Case The intended use case of the dataset is to develop robust text classifiers for benchmarking algorithms that learn from noisy labels, which was our primary focus during its creation. We discourage any unintended usage of the *AlleNoise* dataset.

Competing Datasets To date, several similar benchmark datasets have been published for e-commerce applications of ML algorithms, such as Amazon (Hou et al., 2024), Rakuten (Lin et al., 2019), Skroutz (Akritidis, Fevgas, Bozanis, and Makris, 2020; Akritidis, Fevgas, and Bozanis, 2018), and Shopmania (Akritidis, Fevgas, Bozanis, and Makris, 2020; Akritidis, Fevgas, and Bozanis, 2018). Our dataset competes in size (500,000 instances) and content (5,000 categories). A key difference is the known noise level available in the *AlleNoise* dataset.

Table S4: Distribution of instances and unique categories across departments in the *AlleNoise* dataset.

Department name	# categories	# instances
Automotive	757	**106374**
Home and Garden	**994**	96781
Fashion	223	66839
Electronics	837	60912
Culture and Entertainment	438	58482
Kids	604	39221
Business and Services	364	23089
Sports and Travel	436	13257
Supermarket	375	12669
Beauty	210	12464
Collections and Art	279	7158
Health	174	5058
Real Estate	1	6

Table S5: Distribution of mislabeled instances across departments in the *AlleNoise* dataset.

Department name	# instances
Home and Garden	18623
Business and Services	11934
Electronics	9236
Culture and Entertainment	9150
Automotive	7326
Kids	6273
Supermarket	2716
Collections and Art	2563
Sports and Travel	1827
Fashion	1734
Beauty	1610
Health	1102

G Impact of title translation on classification performance

The dataset was translated from Polish to English, which could impact text representation and the universality of results. To assess this, we compared classifier performance on the original Polish titles versus the translated English ones (**Tab. S8**). The results show that classification accuracy drops by 3.4 p.p. on the clean dataset and by 3.1 p.p. on the noisy dataset after translation, indicating that the translated titles pose a greater challenge for the classifier. However, the goal of the *AlleNoise* dataset is not to maximize classification accuracy but to minimize the performance gap between its noisy and clean variants. The baseline accuracy gap was preserved in translation, amounting to 11.4 p.p. for the Polish version and 11.1 p.p. for the English version. Additionally, the performance of the top-scoring methods on *AlleNoise* (ELR, CCE) showed no significant deviation from the baseline in either version.

H Relation between classification performance and category size

AlleNoise is an imbalanced dataset, similar to well-established benchmarks such as Clothing-1M and WebVision. Balancing the classes, as done in CIFAR-10N, poses risks because it may distort category-specific noise levels, making them no longer representative of the original data's noise distribution. While class imbalance could potentially impact performance accuracy, in *AlleNoise*, class size and classification accuracy are uncorrelated (Pearson correlation coefficient: 0.134), as illustrated in **Fig. S9**.

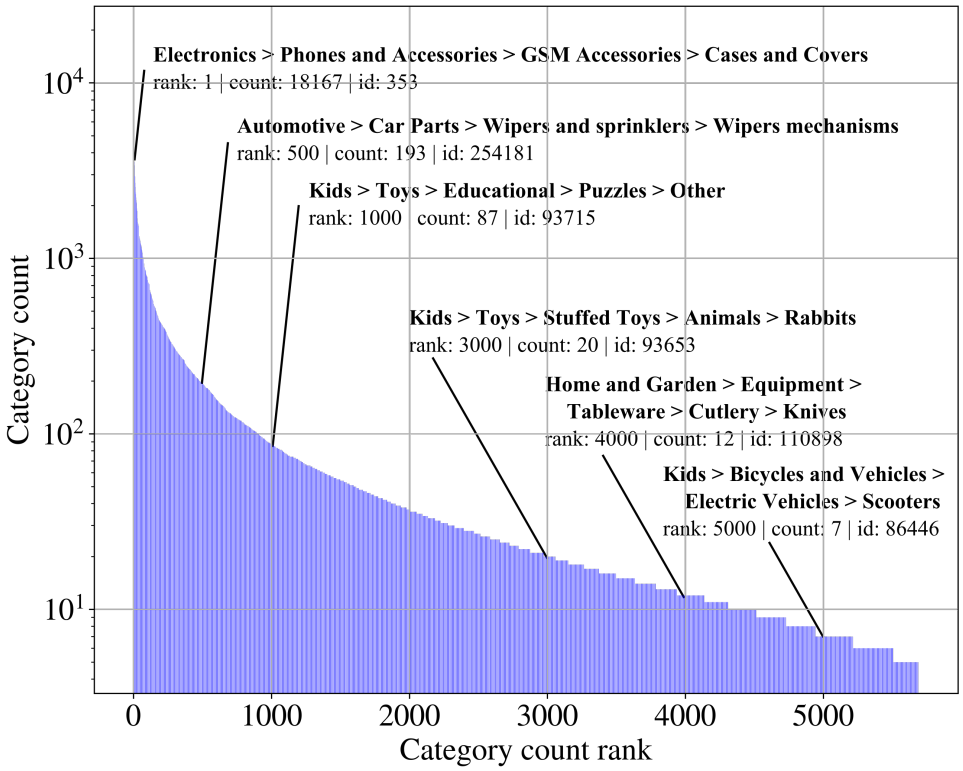

Figure S8: Distribution of category counts in the *AlleNoise* dataset (log y scale). Categories are sorted by their count (x scale).

Table S6: Top-5 categories in terms of the number of instances representing the head of the category tree.

Category name	# instances
Electronics > Phones and Accessories > GSM Accessories > Cases and Covers	18167
Fashion > Clothing, Shoes, Accessories > Men's Clothing > T-shirts	8208
Home and Garden > Equipment > Decorations and Ornaments > Images and Paintings	6143
Fashion > Clothing, Shoes, Accessories > Men's Clothing > Hoodies	5984
Automotive > Car Parts > Brake System > Disc Brakes > Brake Discs	4122

Table S7: Bottom-5 categories in terms of the number of instances representing the long tail of the category tree.

Category name	# instances
Automotive > Cars > Passenger Cars > Tesla > Model S	5
Home and Garden > Tools > Welders > Transformer Welders	5
Electronics > Phones and Accessories > Smartphones and Cell Phones > OnePlus > 8T	5
Home and Garden > Garden > Grilling > Concrete Grills	5
Kids > Toys > Babies > Shushers and Soothers > Other	5

Table S8: Accuracy of the top-scoring methods on the original Polish and translated English versions of *AlleNoise*. Each score represents the average of 5 cross-validation folds.

	Polish		English	
	clean	noisy	clean	noisy
CE	78.23 ± 0.09	66.81 ± 0.13	74.85 ± 0.15	63.71 ± 0.11
ELR	78.18 ± 0.07	66.94 ± 0.14	74.81 ± 0.11	63.72 ± 0.19
CCE	78.17 ± 0.10	66.77 ± 0.16	74.80 ± 0.09	63.73 ± 0.22

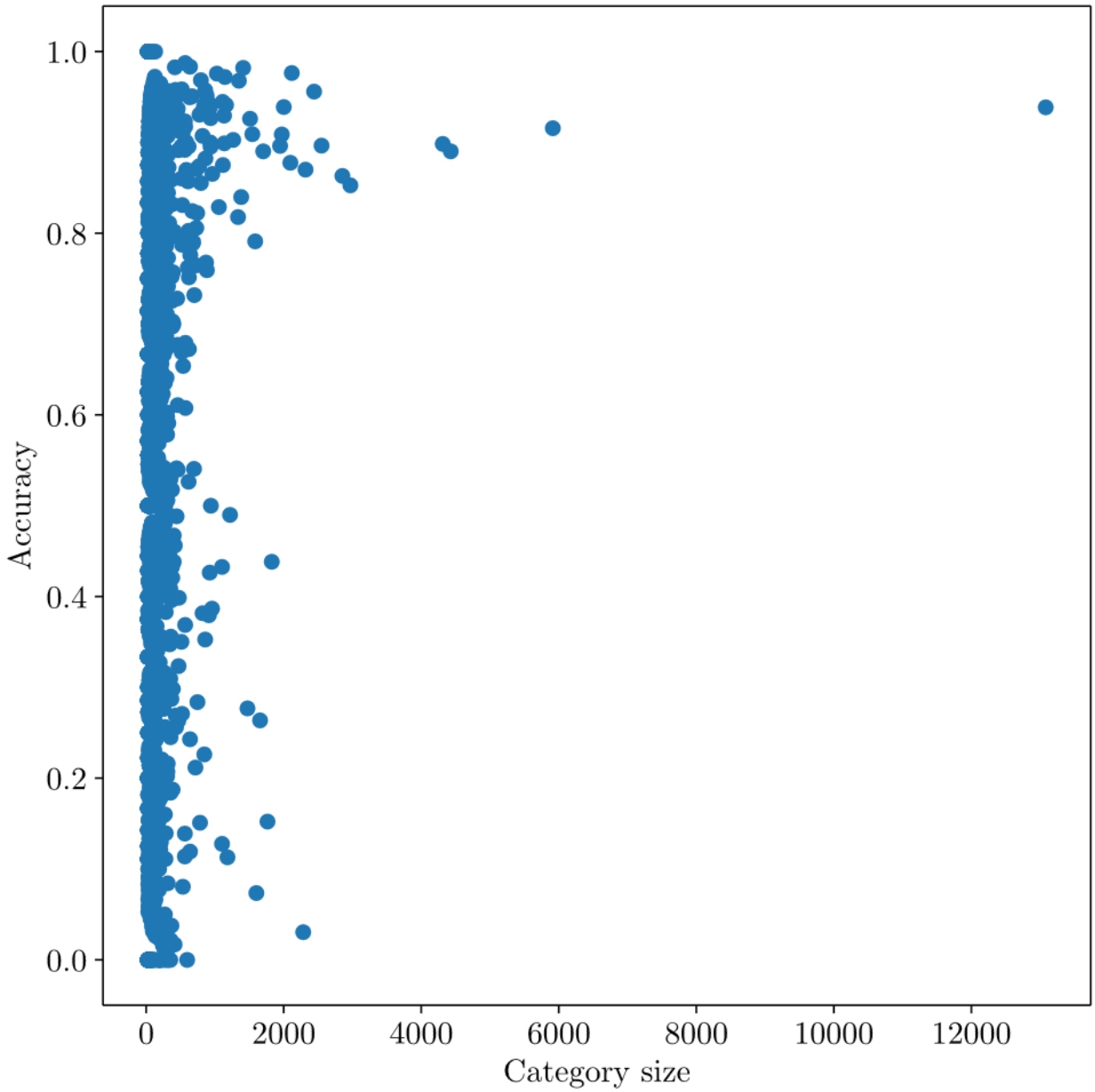

Figure S9: Scatter plot of classification accuracy versus category size for the baseline model using cross-entropy loss. The plot illustrates the lack of correlation between category size and classification accuracy, with a Pearson correlation coefficient of 0.123.