Unsupervised Layer-Wise Score Aggregation for Textual OOD Detection

Maxime Darrin[1,2,3,4], **Guillaume Staerman**[4,6,9], **Eduardo Dadalto Camara Gomes**[4,5,6,7], **Jackie CK Cheung**[2,3,8], **Pablo Piantanida**[1,2,4,6], **Pierre Colombo**[4,7,10,11]

[1]International Laboratory on Learning Systems,
[2]MILA - Quebec AI Institute,
[3]McGill University,
[4]Université Paris-Saclay,
[5] Laboratoire signaux et systèmes,
[6]CNRS,
[7]CentraleSupélec,
[8] Canada CIFAR AI Chair, Mila,
[9] INRIA, CEA, Paris
[10] Equal, Paris,
[11] MICS

Abstract

Out-of-distribution (OOD) detection is a rapidly growing field due to new robustness and security requirements driven by an increased number of AI-based systems. Existing OOD textual detectors often rely on anomaly scores (*e.g.*, Mahalanobis distance) computed on the embedding output of the last layer of the encoder. In this work, we observe that OOD detection performance varies greatly depending on the task and layer output. More importantly, we show that the usual choice (the last layer) is rarely the best one for OOD detection and that far better results can be achieved, provided that an oracle selects the best layer. We propose a data-driven, unsupervised method to leverage this observation to combine layer-wise anomaly scores. In addition, we extend classical textual OOD benchmarks by including classification tasks with a more significant number of classes (up to 150), which reflects more realistic settings. On this augmented benchmark, we show that the proposed post-aggregation methods achieve robust and consistent results comparable to using the best layer according to an oracle while removing manual feature selection altogether.

Introduction

With the increasing deployment of ML tools and systems, the issue of their safety and robustness is becoming more and more critical. Out-of-distribution robustness and detection have emerged as an important research direction. These OOD samples can cause the deployed AI system to fail as neural models rely heavily on previously seen concepts or patterns and tend to struggle with anomalous samples (Berend et al. 2020) or new concepts. These failures affect users' trust or even rule out the adoption of AI in critical applications.

Distinguishing OOD samples (OUT) from in-distribution (IN) samples is a challenge when working on complex data structures (*e.g.*, text or image) due to their high dimensionality. Although OOD detection has attracted much attention in computer vision, few studies focused on textual data. Furthermore, distortion and perturbation methods used in computer vision are not suitable due to the discrete nature of text.

A fruitful line of research (Lee et al. 2018; Liang, Li, and Srikant 2018) focuses on adding simple filtering methods on top of pre-trained models without requiring retraining the model. They include plug-in detectors that rely on softmax-based- or hidden-layer-based- confidence scores (Lee et al. 2018; Liang, Li, and Srikant 2018). Softmax-based detectors (Liu et al. 2020; Pearce, Brintrup, and Zhu 2021; Techapanurak, Suganuma, and Okatani 2019) rely on the predicted probabilities to decide whether a sample is OOD. In contrast, hidden-layer-based scores (*e.g.*, cosine similarity, data-depth (Colombo et al. 2022), or Mahalanobis distance (Lee et al. 2018)) rely on input embedding of the model encoder. In computer vision and more recently in natural language processing, these methods arbitrarily rely on either the embedding generated by the last layer of encoder or on the logits (Wang et al. 2022a; Khalid et al. 2022) to compute anomaly scores. While Softmax-based detectors can be applied in black-box scenarios, where one can only access the model's output, they have a very narrow view of the model's behaviour. In contrast, hidden-layer-based methods enable one to get deeper insights.

We argue that the choice of the penultimate layers (*i.e.*, the last layer, or the logits) ignores the multi-layer nature of the encoder and should be questioned. We give evidence that these representations are (i) not always the best choices (see Fig. 1) and (ii) that leveraging information from all layers can be beneficial. We introduce a data-driven procedure to exploit the information extracted from existing OOD scores across all the different layers of the encoder.

Our contribution can be summarized as follows:

1.**We introduce a new paradigm.** Previous methods rely on a manual selection of the layer to be used, which ignores the information in the other layers of the encoder. We propose an automatic approach to aggregate information from all hidden layers without human (supervised) intervention. *Our method does not require access to OOD samples and harnesses information available in all model layers by leveraging principled anomaly detection tools.*

2.**We conduct extensive experiments on our newly proposed benchmark**: We introduce MILTOOD-C A **M**ultI **L**ingual **T**ext **OOD** detection benchmark for **C**lassification tasks. MILTOOD-C alleviates two main limitations of previous works: (i) contrary to previous work that relies on datasets involving a limited number of classes (up to 5), MILTOOD-C includes datasets with a higher number of classes (up to 150 classes); (ii) MILTOOD-C goes beyond the English-centric setting and includes French, Spanish, and German datasets. Our experiments involve four models and over 186 pairs of IN and OUT datasets, which show that our new aggregation procedures achieve high performance. Previous methods tend to suffer a drop in performance in these more realistic scenarios.

MILTOOD-C: A More Realistic Benchmark

Background and Notations

We adopt a text classification setting and rely on the encoder section* of a model. Let Ω be a vocabulary and Ω^* its Kleene closure[1]. We consider (X, Y) a random variable with values in $\mathcal{X} \times \mathcal{Y}$ such that $\mathcal{X} \subseteq \Omega^*$ is the textual input space, and P_{XY} is its joint probability distribution. The set $\mathcal{Y} = \{1, \dots, C\}$ represents the classes of a classification task and $\mathcal{P}(\mathcal{Y}) = \left\{ \mathbf{p} \in [0,1]^{|\mathcal{Y}|} : \sum_{i=1}^{|\mathcal{Y}|} p_i = 1 \right\}$ the probability simplex over the classes. It is assumed that we have access to a training set $\mathcal{D}_N = \{(\mathbf{x}_i, y_i)\}_{i=1}^N$ composed of independent and identically distributed (i.i.d) realizations of P_{XY}. The Out-Of-Distribution (OOD) detection problem consists of deciding whether a new, previously unseen sample comes (or not) from the IN distribution P_{XY}. The goal is to build a binary function $g : \mathcal{X} \to \{0, 1\}$ based on the thresholding of an anomaly score $s : \mathcal{X} \to \mathbb{R}_+$ that separates IN samples from OOD samples. Namely, for a threshold $\gamma \in \mathbb{R}_+$, we have:

$$g(\mathbf{x}, \gamma) \;=\; \begin{cases} 1 & \text{if } s(\mathbf{x}) > \gamma, \\ 0 & \text{if } s(\mathbf{x}) \leqslant \gamma. \end{cases}$$

Building an OOD Detector

We assume that we have given a classifier $f_\theta : \mathcal{X} \to \mathcal{P}(\mathcal{Y})$:

$$f_\theta = \text{softmax} \circ h \circ \; f_L^\theta \circ f_{L-1}^\theta \circ \cdots \circ f_1^\theta, \qquad (1)$$

with $L > 1$ layers[2], where $f_\ell : \mathbb{R}^{d_{\ell-1}} \to \mathbb{R}^{d_\ell}$ is the ℓ-th layer of the encoder with d_ℓ being the dimension of the latent space after the ℓ-th layer ($d_0 = d$). It is worth noting that in the case of transformers (Vaswani et al. 2017), all latent spaces have the same dimension. Finally, h represents the logit function of the classifier.

To compute the anomaly score s from f_θ, OOD approaches rely on the hidden representations of the (multi-layer) encoder. For $\mathbf{x} \in \mathcal{X}$ an input sequence, we denote $\mathbf{z}_\ell = (f_\ell \circ \cdots \circ f_1)(\mathbf{x})$ its latent representation at layer ℓ. The latent representation obtained after the ℓ-th layer of the training set is denoted as $\mathcal{D}_N^\ell = \{(\mathbf{z}_{\ell,i}, y_i)\}_{i=1}^N$. Furthermore, we denote by $\mathcal{D}_N^{\ell,y}$ the restriction of $\mathcal{D}_N^\ell$ to the samples with label y, i.e., $\mathcal{D}_N^{\ell,y} = \{(\mathbf{z}_{\ell,i}, y_i) \in \mathcal{D}_N^\ell : y_i = y\}$ with $N_y = |\mathcal{D}_N^{\ell,y}|$ indicates the cardinal of this set.

Feature-based OOD detectors usually rely on three key elements:

(i) **Selecting features**: the layer ℓ whose representation is considered to be the input of the anomaly score.

(ii) **A notion of an anomaly (or novelty) score** built on the mapping $\mathcal{D}_N^\ell$ of the training set on the chosen feature space. We can build such a score $s(\cdot, \mathcal{D}_N^\ell)$ defined on $\mathbb{R}^d \times \left(\mathbb{R}^d\right)^N$ for any notion of abnormality.

(iii) **Setting a threshold** to build the final decision function.

Remark 1. ***Choice of the threshold.*** *To select γ, we follow previous work (Colombo et al. 2022; Picot et al. 2023a) by choosing a number of training samples (i.e., "outliers") the detector can wrongfully detect. A classical choice is to set this proportion to 95%.*

Popular Anomaly Scores

In what follows, we present three common anomaly scores for step (ii) of the previously mentioned procedure.

Mahalanobis distance. Authors of Lee et al. (2018) propose to compute the Mahalanobis distance on the abstract representations of each layer and each class. Precisely, this distance is given by:

$$s_M(\mathbf{z}_\ell, \mathcal{D}_N^{\ell,y}) = (\mathbf{z}_\ell - \mu_{\ell,y})^\top \, \Sigma_{\ell,y}^{-1} \, (\mathbf{z}_\ell - \mu_{\ell,y})$$

on each layer ℓ and each class y where $\mu_{\ell,y}$ and $\Sigma_{\ell,y}$ are the estimated class-conditional mean and covariance matrix computed on $\mathcal{D}_N^{\ell,y}$, respectively. The final score from Lee et al. (2018) is obtained by choosing the minimum of these scores over the classes on the penultimate encoder layer.

Integrated Rank-Weighted depth. Colombo et al. (2022) propose to leverage the Integrated Rank-Weighted (IRW) depth (Ramsay, Durocher, and Leblanc 2019). Similar to the Mahalanobis distance, the IRW data depth measures the centrality/distance of a point to a point cloud. For the ℓ-th layer, a Monte-Carlo approximation of the IRW depth can be defined as:

$$s_{\text{IRW}}(\mathbf{z}_\ell, \mathcal{D}_N^{\ell,y}) = \frac{1}{n_{\text{proj}}} \sum_{k=1}^{n_{\text{proj}}} \min\left\{ \frac{1}{n} \sum_{i=1}^{N_y} \mathbb{1}\left\{ g_{k,i}(\mathbf{z}_\ell) \leqslant 0 \right\}, \; \frac{1}{n} \sum_{i=1}^{N_y} \mathbb{1}\left\{ g_{k,i}(\mathbf{z}_\ell) > 0 \right\} \right\},$$

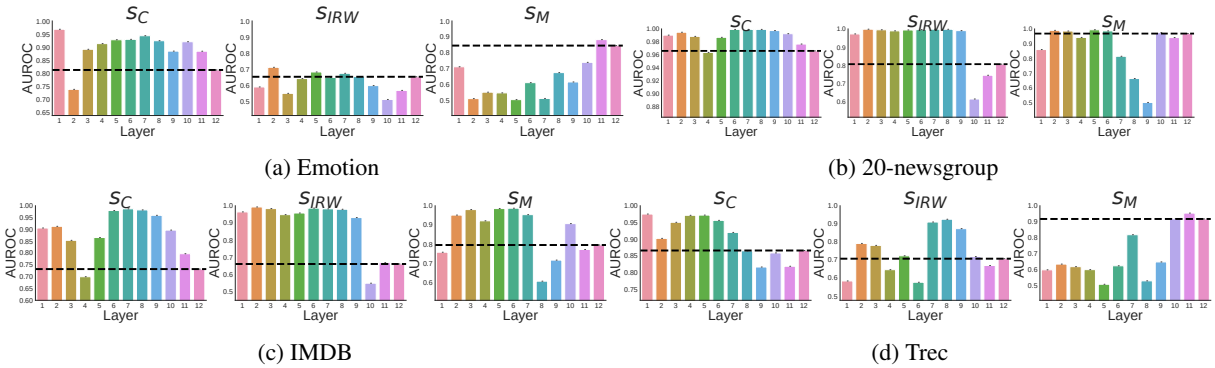
where $g_{k,i}(\mathbf{z}_\ell) = \langle u_k, \mathbf{z}_{\ell,i} - \mathbf{z}_\ell \rangle$, $u_k \in \mathbb{S}^{d-1}, \mathbf{z}_{\ell,i} \in \mathcal{D}_N^{\ell,y}$ where $\mathbb{S}^{d-1} = \{x \in \mathbb{R}^d \; : \; ||x|| = 1\}$ is the unit hypersphere and n_{proj} is the number of directions sampled on the sphere.

Cosine similarity. Zhou, Liu, and Chen (2021) propose to compute the maximum cosine similarity between the embedded sample $\mathbf{z}_\ell$ and the training set $\mathcal{D}_N^\ell$ at layer ℓ:

$$s_C(\mathbf{z}_\ell, \mathcal{D}_N^\ell) = - \max_{\mathbf{z}_{\ell,i} \in \mathcal{D}_N^\ell} \; \frac{\langle \mathbf{z}_\ell, \mathbf{z}_{\ell,i} \rangle}{||\mathbf{z}_\ell|| \, ||\mathbf{z}_{\ell,i}||},$$

[1]The Kleene closure corresponds to sequences of arbitrary size written with words in Ω. Formally: $\Omega^* = \bigcup_{i=0}^{\infty} \Omega^i$.

[2]For the sake of brevity, we omit the parameters θ in the following.



(a) Emotion (b) 20-newsgroup

(c) IMDB (d) Trec

Figure 1: OOD detection performance in terms of AUROC ↑ for each features-based OOD score (Mahalanobis distance (s_M), Maximum cosine similarity (s_C) and IRW (s_{IRW})) computed at each layer of the encoder for different OOD datasets for a model fine-tuned on SST2. We observe that the performance of each metric on each layer varies significantly with the OOD task and that OOD detection based on the last layer (dark dotted line) rarely yields the best results.

where $\langle \cdot, \cdot \rangle$ and $||\cdot||$ denote the Euclidean inner product and norm, respectively. They also choose the penultimate layer. It is worth noting they do not rely on a per-class decision.

Related Works and Limitations

We claim that the choice of layer is crucial in textual OOD detection; we report in Fig. 1 the OOD performance of popular detectors described in Sec. , applied at each layer of the encoder. We observe a high variability across different layers. The last layer is rarely the best-performing layer, and there is room for improvement if we could choose the best possible layer or gather useful information from all of them. This observation is consistent with the literature, as neural networks are known to extract different information and construct different abstractions at each layer (Ilin, Watson, and Kozma 2017).

Works relying on a manually selected layer. The choice of layer for step (i) in Sec. is not usually a question. Most work arbitrarily relies on the logits (Liang, Li, and Srikant 2018; Liu et al. 2020) or the last layer of the encoder (Yang et al. 2021; Hendrycks and Gimpel 2016; Wang et al. 2022a). We argue that these choices are unjustified and that previous work gives up on important information in the other layers.

Works that feature layer aggregation. Besides Colombo et al., which proposes to aggregate the features instead of aggregating the OOD scores as we do, we are unaware of any other work suggesting layer aggregation for textual OOD detection. There, however, exist several adjacent works in computer vision. Abdelzad et al. proposes a method which starts by finding the best layer for OOD detection for a validation set and uses that layer afterwards. Still, there is no adaptation for different datasets. Other works such as Lin, Roy, and Li focus on training models with specific architectures (for computer vision) to leverage early-stage exit or different information from different layers to detect OOD samples or improve generalization on OOD samples. In addition, these works rely on image-specific properties that do not translate well in text (lossy compression for example). More closely related work are Wang et al.; Lee et al.. The first proposes to compute OOD scores at each layer (1-SVM), and they select the layer that yields the highest confidence in terms of margin. However, their work features a single OOD score and a single aggregation method in computer vision. In contrast, we propose a more general framework and systematically evaluate different OOD scores and aggregation procedures. The second proposes to learn a linear combination of Mahalanobis scores but does not explore OOD score aggregation or layer selection systematically. In addition, both of these works have been proposed in computer vision and show that aggregation does not lead to significant improvements. This dramatically contrasts with our findings in textual models, consistent with the literature (Picot et al. 2023a; Raghuram et al. 2021).

We propose to compute standard OOD scores on each layer of the encoder (and not only on the logits or the representation generated by the last layer) and to aggregate this score in an unsupervised fashion to select and combine the most relevant following the task at hand.

Leveraging Information from All Layers

In this section*, we describe our aggregation methods that use the information available in the different layers of the encoder.

Problem Statement

For an input $\mathbf{x} \in \mathcal{X}$ and a training dataset $\mathcal{D}_N$, we obtain their set of embedding representation sets: $\{\mathbf{z}_\ell\}_{\ell=1}^L$ and $\{\mathcal{D}_N^\ell\}_{\ell=1}^L$, respectively. Given an anomaly score function $s : \mathbb{R}^d \times \left(\mathbb{R}^d\right)^N \to \mathbb{R}$ (e.g., those described in Sec.), we define the OOD score set of an input $\mathbf{x}$ as $\mathcal{S}_s(\mathbf{x}; \mathcal{D}_N) = \{\{s(\mathbf{z}_\ell; \mathcal{D}_N^{\ell,y})\}_{\ell=1}^L\}_{y=1}^C \in \mathbb{R}^{L \times C}$. Similarly, it is possible to obtain a reference set of $\mathcal{R}(\mathcal{D}_N) = \{\mathcal{S}_s(\mathbf{x}; \mathcal{D}_N),\ \forall\ (\mathbf{x}, y) \in \mathcal{D}_N\}$ from the training data[3]. In what follows, we aim to answer the following question.

Can we leverage all the information available in $\mathcal{S}_s(\mathbf{x}; \mathcal{D}_N)$ and/or $\mathcal{R}(\mathcal{D}_N)$ to build an OOD detector?

[3]When using the cosine similarity, which does not rely on a per-class decision, $\mathcal{S}_s(\mathbf{x}; \mathcal{D}_N)$ is reduced to $\{s(\mathbf{z}_\ell; \mathcal{D}_N^\ell)\}_{\ell=1}^L$.

Proposed Framework

Our framework aims at comparing the set of scores of a sample to the sets of scores of a reference relying on principled anomaly detection algorithms.

The goal of this work is to propose a data-driven aggregation method of OOD scores[4], Agg. Agg is defined as:

$$\begin{aligned}\text{Agg}: \quad & \mathbb{R}^{L\times C} \times \left(\mathbb{R}^{L\times C}\right)^N \to \mathbb{R} \\ & (\mathcal{S}_s(\mathbf{x};\mathcal{D}_N), \mathcal{R}(\mathcal{D}_N)) \to \text{Agg}(\mathcal{S}_s, \mathcal{R}),\end{aligned}$$

where $\mathbf{x}$ denotes the input sample.

Intuition. This framework allows us to consider the whole trace of a sample through the model. This formulation has two main advantages: it avoids manual layer selection and enables us to leverage information from all the encoder layers.

We propose two families of approaches: (i) one solely relies on the score set $\mathcal{S}_s(\mathbf{x};\mathcal{D}_N)$ (corresponding to a no-reference scenario and denoted as $\text{Agg}_\emptyset$) and (ii) the second one (named reference scenario) leverages the reference set $\mathcal{R}(\mathcal{D}_N)$.

Remark 2. *It is worth noting that our framework through* $\text{Agg}_{\emptyset set}$ *or* Agg *naturally includes previous approaches (Lee et al. 2018; Colombo et al. 2022). For example, the detector of Lee et al. (2018) can be obtained by defining* $\text{Agg}_\emptyset$ *as the minimum of the penultimate line of the matrix* $\mathcal{S}_s(\mathbf{x}, \mathcal{D}_N)$.

Detailed Aggregation Procedures

Intuition. Our framework through Agg and $\text{Agg}_\emptyset$ requires two types of operations to extract a single score from $\mathcal{S}_s(\mathbf{x}, \mathcal{D}_N)$ and $\mathcal{R}_s(\mathcal{D}_N)$: one aggregation operation over the layers and one aggregation operation over the classes, where necessary.

Our framework in a nutshell. We assume we are given an anomaly score, s, that we want to enhance by leveraging all the layers of the encoder. For a given input $\mathbf{x}$, our framework follows 4 steps (see Fig. 2 for a depiction of the procedure):

1. Compute the embeddings $\{\mathbf{z}_l\}_{l=1}^L$ for $\mathbf{x}$ and every element of $\mathcal{D}_N$.
2. Form $\mathcal{S}_s(\mathbf{x};\mathcal{D}_N)$ and $\mathcal{R}(\mathcal{D}_N)$ using the score s.
3. Perform $\text{Agg}_\emptyset$ or Agg:

(a) (*per layer*) Aggregate score information over the layers to obtain a vector composed of C scores.

(b) (*per class*) Take the minimum value of this vector.

4. Apply a threshold γ on that value.

Step (3.b). is inspired by the OOD literature (Lee et al. 2018; Colombo et al. 2022). It relies on the observation that if the input sample is IN-distribution, it is expected to have at least one low score in the class vector, whereas an OOD sample should only have high scores equivalent to a high minimum score.

No-reference scenario ($\text{Agg}_\emptyset$) In the no-reference scenario, we have access to a limited amount of information. We thus propose to rely on simple statistics to aggregate the OOD scores available in $\mathcal{S}_s(\mathbf{x};\mathcal{D}_N)$ to compute step (3). of the proposed procedure. Precisely, we use the *average*, the minimum (*min*), the median (*med*), and *coordinate* (see Remark 2) operators on the column of the matrix $\mathcal{S}_s(\mathbf{x};\mathcal{D}_N)$.

Data-driven scenario (Agg) In the data-driven scenario, Agg also has access to the set of reference OOD scores (*i.e.*, $\mathcal{R}_s(\mathcal{D}_N)$) for the given OOD score s. The goal, then, is to compare the score set $\mathcal{S}_s(\mathbf{x};\mathcal{D}_N)$ of the input with this reference set $\mathcal{R}_s(\mathcal{D}_N)$ to obtain a score vector of size C. *In the following, we propose an original solution for the layer operation.*

For the *per layer* operation we rely on an anomaly detection algorithm for each class $\mathcal{A}_y$ defined as:

$$\begin{aligned}\mathcal{A}_y : \mathbb{R}^L \times (\mathbb{R}^L)^{N_y} &\to \mathbb{R} \\ \mathbf{s}_y \times \mathcal{R}_y &\mapsto \mathcal{A}_y(\mathbf{s}_y, \mathcal{R}_y),\end{aligned} \tag{2}$$

where $\mathbf{s}_y = \{s(\mathbf{z}_\ell; \mathcal{D}_N^{\ell,y})\}_{\ell=1}^L$ and $\mathcal{R}_y = \mathcal{R}(\mathcal{D}_N^y)$.

Remark 3. $\mathcal{A}_y$ *is trained on the reference set* $\mathcal{R}_y$ *for each class and* ***thus does not involve any OOD samples****. The score returned for a vector* $\mathbf{s}_y$ *is the prediction score associated with the trained algorithm.*

Remark 4. *We define a per-class decision for* Agg *since it has been shown to be significantly more effective than global scores (Huang and Li 2021). It is the approach chosen by most state-of-the-art-methods. We have validated this approach by conducting extensive experiments.*

We propose several popular anomaly detection algorithms. First, we offer to reuse common OOD scores (s_M, s_C, s_{IRW}) as aggregation methods: they are now trained on the reference set of sets of OOD scores $\mathcal{R}_s(\mathcal{D}_N)$ and provide a notion of anomaly for the trace of a sample through the model. We also compute the median and average score as natural baselines for the aggregation setting. In addition, we propose more elaborate anomaly detection algorithms such as Isolation Forest (`IF`) (Liu, Ting, and Zhou 2008) and the Local Outlier Factor (`LOF`) (Breunig et al. 2000). Below, we briefly recall the general insights of each of these algorithms. It is important to emphasize that our framework can accommodate any anomaly detection algorithms.

Local Outlier Factor. This method compares a sample's density with its neighbours' density. Any sample with a lower density than its neighbours is regarded as an outlier.

Isolation Forest. This popular algorithm is built on the idea that anomalous instances should be easily distinguished from normal instances. It leads to a score that measures the complexity of separating a sample from others based on the number of necessary decision trees required to isolate a data point. It is computationally efficient, benefits from stable hyper-parameters, and is well suited to the unsupervised setting.

Comparison to Baseline Methods

Current State-of-the-art methods for OOD detection on textual data have been recently provided in Colombo et al. (2022) (`PW`). They aggregate the hidden layers using Power means and then apply an OOD score on this aggregated representation. They achieved previous SOTA performance by coupling

[4]We do not assume that we have access to OOD samples as they are often not available.

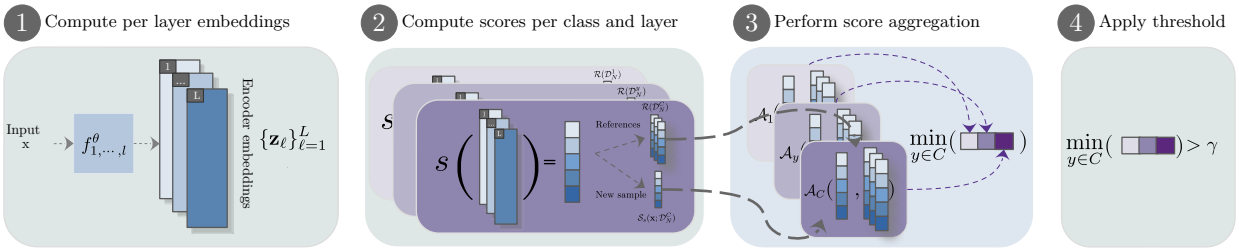


Figure 2: Schema of our aggregation procedure. (1) We extract the embeddings at each layer of the encoder for every sample. (2) We compute the per-class scores for a reference set and the new sample to be evaluated for each layer embedding. (3) We aggregate the scores over every layer to get an aggregated per-class score before taking the min score over the classes. (4) Finally, we apply the threshold on this minimum.

it with the IRW depth and proposed a comparison with Mahalanobis and Cosine versions. We reproduce these results as it is a natural baseline for aggregation algorithms.

Last Layer. Considering that the model's last layer or logits should output the most abstract representation of an input, it has been the primary focus of attention for OOD detection. It is a natural choice for any architecture or model and therefore removes the hurdle of selecting features for different tasks and architectures. For this heuristic, we obtain OOD scores using the Mahanalobis distance (as in (Lee et al. 2018)), the IRW score (as in (Colombo et al. 2022)), and the cosine similarity.

Additional methods. It is common on OOD detection methods to report the Maximum Softmax Prediction (MSP) (Hendrycks and Gimpel 2016) as well as the Energy Score (E) (Liu et al. 2020).

`MILTOOD-C`: A More Realistic Benchmark

In this section*, we highlight the limitations of existing benchmarks and introduce our own: `MILTOOD-C` A **M**ultI **L**ingual **T**ext **OOD** for classification tasks.

Limitation of Existing Benchmarks

Number of classes. Text classification benchmarks for OOD detection often consist of sentiment analysis tasks involving a small number of classes (Fang and Zhan 2015; Kharde, Sonawane et al. 2016). Those tasks with a larger number of classes have been mostly ignored in previous OOD detection benchmarks (Colombo et al. 2022; Li et al. 2021; Zhou, Liu, and Chen 2021). However, real-world problems do involve vastly multi-class classification tasks (Casanueva et al. 2020). Previous work in computer vision found that these problems require newer and carefully tuned methods to enable OOD detection in this more realistic setting (Deng et al. 2009; Le and Yang 2015).

Monolingual datasets. Most methods have been tested on architectures tailored for the English language (Colombo et al. 2022; Li et al. 2021; Arora, Huang, and He 2021). With inclusivity and diversity in mind (Ruder 2022; van Esch et al. 2022), it is necessary to assess the performance of old and new OOD detection methods on a variety of languages (Srinivasan et al. 2021; de Vries, van Cranenburgh, and Nissim 2020; Baheti et al. 2021; Zhang et al. 2022).

A More Realistic Benchmark

We now present `MILTOOD-C`, which addresses the aforementioned limitations. It consists of more than 25 datasets involving up to 150 classes and 4 languages.

Dataset selection. We gathered a large and diverse benchmark encompassing many shift typologies, tasks, and languages. It covers 27 datasets in 4 different languages (*i.e.*, English, German, Spanish, and French) and classifications tasks involving 2 to 150 classes. Following standard protocol (Hendrycks et al. 2020), we train a classifier for each in-distribution dataset (`IN-DS`) while the OOD dataset (`OUT-DS`) is coming from a different dataset. We provide a comprehensive list of the 180 pairs. It is an order of magnitude larger than recent concurrent work from (Colombo et al. 2022).

English benchmark. We relied on the benchmark proposed by Zhou, Liu, and Chen (2021); Hendrycks et al. (2020). It features three types of `IN-DS`: sentiment analysis (*i.e.*, SST2 (Socher et al. 2013), IMDB (Maas et al. 2011)), topic classification (*i.e.*, 20Newsgroup (Joachims 1996)) and question answering (*i.e.*, TREC-10 and TREC-50 (Li and Roth 2002)). We also included the Massive (FitzGerald et al. 2022) dataset and the Banking (Casanueva et al. 2020) for a larger number of classes and NLI datasets (*i.e.*, RTE (Burger and Ferro 2005; Hickl et al. 2006) and MNLI (Williams, Nangia, and Bowman 2018)) following. To go one step further in terms of number of classes, we considered HINT3(Arora et al. 2020) and clink (Larson et al. 2019). We form IN and OOD pairs between the aforementioned tasks.

Beyond English-centric tasks.[5] For language-specific datasets, we added the same tasks as for English when available and extended it with language-specific datasets such as the PAWS-S datasets (Yang et al. 2019), film reviews in French and Spanish (Blard 2019). For French and German, we also added the Swiss judgments datasets (Niklaus, Stürmer, and Chalkidis 2022). Finally, we added different tweet classification tasks for each language (English, German, Spanish and French) (Zotova et al. 2020; Barbieri, Espinosa Anke, and Camacho-Collados 2022).

[5]We did not work on language changes because they were easily detected with all the methods considered. Instead, we focus on intra-language drifts.

Model selection. To ensure that our results are consistent not only across tasks and shifts, but also across model architectures, we train classifiers based on 6 different Transformer decoders: BERT (Devlin et al. 2018) (base, large and multilingual versions), DISTILBERT (Sanh et al. 2019) and RoBERTa (Liu et al. 2019) (base and large versions) fine-tuned on each task.

Evaluation metrics. The OOD detection problem is a binary classification problem where the positive class is OUT. We follow concurrent work (Colombo et al. 2022; Darrin, Piantanida, and Colombo 2022) and evaluate our detector using threshold-free metrics such as AUROC ↑, AUPR-IN/AUPR-OUT and threshold based metrics such as FPR ↓ at 95% and Err.

Experimental Results

Quantifying Aggregation Gains

Overall results. Data driven aggregation methods (*i.e.*, with reference) consistently outperform any other baselines or tested methods by a significant margin (see Table 1) on our extensive MILTOOD-C benchmark. According to our experiments, the best combination of hidden feature-based OOD score and aggregation function is to use the Maximum cosine similarity as the underlying OOD score and to aggregate these scores using the IRW data depth (s_{IRW}). A first time to get the abnormality of the representations of the input and a second time to assess the abnormality of the set of layer-wise scores through the model. It reaches an average AUROC ↑ of 0.99 and a FPR ↓ of 0.02. It is a gain of more than 6% compared to the previous *state-of-the-art* methods in terms of AUROC ↑ and more than 90% in FPR ↓.

Most versatile aggregation method. While the s_C and s_{IRW} used as aggregation methods achieve excellent performance when paired with s_C as the underlying OOD score, they fail to aggregate as well other underlying scores. Whereas the isolation forest algorithm is a more versatile and consistent data-driven aggregation method: it yields performance gain for every underlying OOD score.

Performance of common baselines. We show that, on average, using the last layer or the logits as features to perform OOD detection leads to poorer results than almost every other method. **It is interesting to point out that this is not the case in computer vision (Yang et al. 2021; Raghuram et al. 2021; Picot et al. 2023a; Lee et al. 2018). This finding further motivates the development of OOD detection methods tailored for text.**

Impact of data-driven aggregation. In almost all scenarios, aggregating the score using a data-driven anomaly detection method leads to a significant gain in performance compared to baseline methods. This supports our claim that useful information is scattered across the layers currently ignored by most methods. **We show that this information can be retrieved and effectively leveraged to improve OOD detection.**

	Ours	Agg.	AUROC ↑		FPR ↓	
E	Bas.	E	0.83	±0.18	0.39	±0.31
s_M	Agg	s_M	0.90	±0.14	0.27	±0.33
		s_C	0.88	±0.17	0.32	±0.40
		s_{IRW}	0.81	±0.20	0.44	±0.42
		IF	0.94	±0.10	0.19	±0.25
		LOF	0.87	±0.15	0.39	±0.37
	$\text{Agg}_\emptyset$	Mean	0.74	±0.18	0.59	±0.39
		Median	0.75	±0.17	0.61	±0.35
		PW	0.80	±0.17	0.61	±0.38
	Bas.	Last layer	0.92	±0.11	0.25	±0.31
		Logits	0.71	±0.14	0.65	±0.27
s_C	Agg	s_M	0.93	±0.11	0.20	±0.27
		s_C	0.98	±0.10	0.04	±0.19
		s_{IRW}	**0.99**	±0.07	**0.02**	±0.15
		IF	0.94	±0.14	0.12	±0.29
		LOF	0.93	±0.11	0.20	±0.26
	$\text{Agg}_\emptyset$	Mean	0.93	±0.12	0.25	±0.33
		Median	0.92	±0.12	0.27	±0.34
		PW	0.93	±0.11	0.19	±0.27
	Bas.	Last layer	0.92	±0.11	0.22	±0.26
		Logits	0.81	±0.17	0.52	±0.42
s_{IRW}	Agg	s_M	0.81	±0.18	0.50	±0.38
		s_C	0.89	±0.17	0.28	±0.36
		s_{IRW}	0.82	±0.19	0.43	±0.39
		IF	0.89	±0.15	0.34	±0.36
		LOF	0.82	±0.15	0.54	±0.35
	$\text{Agg}_\emptyset$	Mean	0.84	±0.18	0.47	±0.40
		Median	0.82	±0.18	0.50	±0.39
		PW	0.74	±0.17	0.64	±0.34
	Bas.	Last layer	0.66	±0.14	0.79	±0.21
		Logits	0.73	±0.16	0.64	±0.28
MSP	Bas.	MSP	0.83	±0.17	0.39	±0.28

Table 1: Average performance of each considered metric over all the OOD pairs and model architectures in terms of AUROC ↑, Err, and FPR ↓. For each common OOD score, we report the results obtained using every aggregation method or choice of features to consider. The best method overall is highlighted in bold and the best methods per underlying metric and setting are underlined.

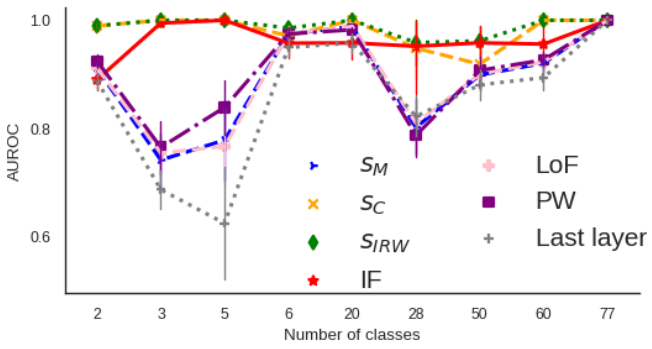


Figure 3: Average performance of OOD detectors in terms of AUROC ↑ for tasks involving different numbers of classes.

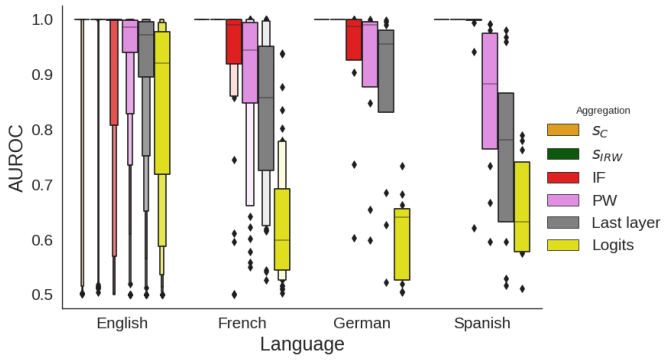


Figure 4: Stability and robustness comparison of the best-performing aggregation methods and underlying OOD scores with S_C as underlying OOD score. Common baselines and SOTA display significant deviations in performance with the different languages, whereas score aggregation methods induce more consistent and better performance.

Post Aggregation is More Stable Across Task, Language, Model Architecture

Most OOD scores have been crafted and finetuned for specific settings. In the case of NLP, they have usually been validated only on datasets involving a small number of classes or on English tasks. In this section*, we study the stability and consistency of the performance of each score and aggregation method in different settings.

Stability of performance across tasks. In Fig. 3, we plot the average AUROC ↑ across our models and datasets per number of classes of the IN dataset. It is, therefore, the number of classes output by the model. Our best post-aggregation methods (*i.e.,* Maximum cosine similarity and Integrated Rank-Weigthed) produced more consistent results across all settings. It can maintain excellent performance for all types of datasets, whereas the performance of baselines and other aggregation methods tends to fluctuate from one setting to another. *More generally, we observe that data-driven aggregation methods tend to perform consistently on all tasks, whereas previous baselines' performance tends to vary.*

Features aggregation vs. OOD scores aggregation. Interestingly, we show that while Power Means pre-aggregation of the features yields better results than single-layer scores, they still follow the same trend, and the gain is more minor and inconsistent.

Stability of results across languages. In Fig. 4 we present the deviations in performance of the different OOD detection methods and show that our methods are significantly more robust across languages and tasks than baselines and previous SOTA.

Comparison with the oracle. As shown in Fig. 1, there often exists a layer that yields very high OOD detection performance. An oracle that knows which layer to consider would perform better than all of the baselines and SOTA, as they know the best layer. In Fig. 5, we show that our aggregation methods can outperform that oracle (Green bar, S_C scores aggregated with S_{IRW}), whereas SOTA and baselines yield significantly worse results.

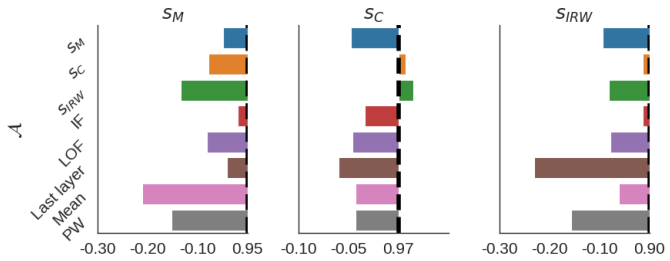


Figure 5: Average performance difference in terms of AUROC between aggregation methods and the oracle (best possible layer).

Results in Computer Vision

The proposed method can be applied to computer vision, however, several works have shown that in that setting the logits are the best layer to perform OOD detection on (Picot et al. 2023b; Lee et al. 2018), alleviating the need for layer selection or aggregation. In Raghuram et al., they fail to demonstrate any gain in OOD detection using score aggregation.

Conclusions

We proposed aggregating OOD scores across all the layers of the encoder of a text classifier instead of relying on scores computed on a single hand-picked layer (logits) to improve OOD detection. We confirmed that all the layers are not equal regarding OOD detection and, more importantly, that the common choices for OOD detection (logits) are often not the best choice. We validated our methods on an extended text OOD classification benchmark MILTOOD-C we introduced. We showed that our aggregation methods are not only able to outperform previous baselines and recent work, but they were also able to outperform an oracle that would be able to choose the best layer to perform OOD detection for a given task. This leads us to conclude that valuable information for OOD detection is scattered across all the encoder layers. While this tool shows promising results, it should not be trusted blindly. It relies on anomaly detection with respect to the training set and thus can remove otherwise well-handled samples. It is also not infallible and can miss OOD samples, which can cause harm to the model's performance.

Acknowledgements

We thank our reviewers for their insights and useful advice during the different reviewing processes. This work was performed using HPC resources from GENCI–IDRIS (Grant 2022-AD011013945). This work was supprted by HPC resources of CINES and GENCI. The authors would like to thank the staff of CINES for technical support in managing the Adastra GPU cluster, in particular; Jean-Christophe Penalva, Johanne Charpentier, Mathieu Cloirec, Jerome Castaings, Gérard Vernou, Bertrand Cirou and José Ricardo Kouakou.

References

Abdelzad, V.; Czarnecki, K.; Salay, R.; Denounden, T.; Vernekar, S.; and Phan, B. 2019. Detecting Out-of-Distribution Inputs in Deep Neural Networks Using an Early-Layer Output. arXiv:1910.10307.

Arora, G.; Jain, C.; Chaturvedi, M.; and Modi, K. 2020. HINT3: Raising the bar for Intent Detection in the Wild. In *Proceedings of the First Workshop on Insights from Negative Results in NLP*, 100–105. Online: Association for Computational Linguistics.

Arora, U.; Huang, W.; and He, H. 2021. Types of Out-of-Distribution Texts and How to Detect Them. *arXiv preprint arXiv:2109.06827*.

Baheti, A.; Sap, M.; Ritter, A.; and Riedl, M. 2021. Just say no: Analyzing the stance of neural dialogue generation in offensive contexts. *arXiv preprint arXiv:2108.11830*.

Barbieri, F.; Espinosa Anke, L.; and Camacho-Collados, J. 2022. XLM-T: Multilingual Language Models in Twitter for Sentiment Analysis and Beyond. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, 258–266. Marseille, France: European Language Resources Association.

Berend, D.; Xie, X.; Ma, L.; Zhou, L.; Liu, Y.; Xu, C.; and Zhao, J. 2020. Cats are not fish: Deep learning testing calls for out-of-distribution awareness. In *Proceedings of the 35th IEEE/ACM International Conference on Automated Software Engineering*, 1041–1052.

Blard, T. 2019. French sentiment analysis with BERT. https://github.com/TheophileBlard/french-sentiment-analysis-with-bert.

Breunig, M. M.; Kriegel, H.-P.; Ng, R. T.; and Sander, J. 2000. LOF: identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, 93–104.

Burger, J.; and Ferro, L. 2005. Generating an entailment corpus from news headlines. In *Proceedings of the ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment*, 49–54. Association for Computational Linguistics.

Casanueva, I.; Temcinas, T.; Gerz, D.; Henderson, M.; and Vulic, I. 2020. Efficient Intent Detection with Dual Sentence Encoders. In *Proceedings of the 2nd Workshop on NLP for ConvAI - ACL 2020*. Data available at https://github.com/PolyAI-LDN/task-specific-datasets.

Colombo, P.; Gomes, E. D. C.; Staerman, G.; Noiry, N.; and Piantanida, P. 2022. Beyond Mahalanobis Distance for Textual OOD Detection. In *Advances in Neural Information Processing Systems*.

Darrin, M.; Piantanida, P.; and Colombo, P. 2022. Rainproof: An Umbrella To Shield Text Generators From Out-Of-Distribution Data. *arXiv preprint arXiv:2212.09171*.

de Vries, W.; van Cranenburgh, A.; and Nissim, M. 2020. What's so special about BERT's layers? A closer look at the NLP pipeline in monolingual and multilingual models. *arXiv preprint arXiv:2004.06499*.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Fang, X.; and Zhan, J. 2015. Sentiment analysis using product review data. *Journal of Big Data*, 2(1): 1–14.

FitzGerald, J.; Hench, C.; Peris, C.; Mackie, S.; Rottmann, K.; Sanchez, A.; Nash, A.; Urbach, L.; Kakarala, V.; Singh, R.; Ranganath, S.; Crist, L.; Britan, M.; Leeuwis, W.; Tur, G.; and Natarajan, P. 2022. MASSIVE: A 1M-Example Multilingual Natural Language Understanding Dataset with 51 Typologically-Diverse Languages. arXiv:2204.08582.

Hendrycks, D.; and Gimpel, K. 2016. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136*.

Hendrycks, D.; Liu, X.; Wallace, E.; Dziedzic, A.; Krishnan, R.; and Song, D. 2020. Pretrained transformers improve out-of-distribution robustness. *arXiv preprint arXiv:2004.06100*.

Hickl, A.; Williams, J.; Bensley, J.; Roberts, K.; Rink, B.; and Shi, Y. 2006. Recognizing textual entailment with LCC's GROUNDHOG system. In *Proceedings of the Second PASCAL Challenges Workshop*.

Huang, R.; and Li, Y. 2021. Mos: Towards scaling out-of-distribution detection for large semantic space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 8710–8719.

Ilin, R.; Watson, T.; and Kozma, R. 2017. Abstraction hierarchy in deep learning neural networks. In *2017 International Joint Conference on Neural Networks (IJCNN)*, 768–774. IEEE.

Joachims, T. 1996. A Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization. Technical report, Carnegie-mellon univ pittsburgh pa dept of computer science.

Khalid, U.; Esmaeili, A.; Karim, N.; and Rahnavard, N. 2022. Rodd: A self-supervised approach for robust out-of-distribution detection. *arXiv preprint arXiv:2204.02553*.

Kharde, V.; Sonawane, P.; et al. 2016. Sentiment analysis of twitter data: a survey of techniques. *arXiv preprint arXiv:1601.06971*.

Larson, S.; Mahendran, A.; Peper, J. J.; Clarke, C.; Lee, A.; Hill, P.; Kummerfeld, J. K.; Leach, K.; Laurenzano, M. A.; Tang, L.; and Mars, J. 2019. An Evaluation Dataset for Intent Classification and Out-of-Scope Prediction. *CoRR*, abs/1909.02027.

Le, Y.; and Yang, X. 2015. Tiny ImageNet Visual Recognition Challenge.

Lee, K.; Lee, K.; Lee, H.; and Shin, J. 2018. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in neural information processing systems*, 31.

Li, X.; Li, J.; Sun, X.; Fan, C.; Zhang, T.; Wu, F.; Meng, Y.; and Zhang, J. 2021. k Folden: k-Fold Ensemble for Out-Of-Distribution Detection. *arXiv preprint arXiv:2108.12731*.

Li, X.; and Roth, D. 2002. Learning Question Classifiers. In *COLING 2002: The 19th International Conference on Computational Linguistics*.

Liang, S.; Li, Y.; and Srikant, R. 2018. Enhancing The Reliability of Out-of-distribution Image Detection in Neural Networks. In *International Conference on Learning Representations*.

Lin, Z.; Roy, S. D.; and Li, Y. 2021. Mood: Multi-level out-of-distribution detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15313–15323.

Liu, F. T.; Ting, K. M.; and Zhou, Z.-H. 2008. Isolation Forest. In *In Proceedings 8th IEEE International Conference on Data Mining*, 413–422.

Liu, W.; Wang, X.; Owens, J.; and Li, Y. 2020. Energy-based Out-of-distribution Detection. *Advances in Neural Information Processing Systems*.

Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; and Stoyanov, V. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Maas, A. L.; Daly, R. E.; Pham, P. T.; Huang, D.; Ng, A. Y.; and Potts, C. 2011. Learning Word Vectors for Sentiment Analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, 142–150. Portland, Oregon, USA: Association for Computational Linguistics.

Niklaus, J.; Stürmer, M.; and Chalkidis, I. 2022. An Empirical Study on Cross-X Transfer for Legal Judgment Prediction. arXiv:2209.12325.

Pearce, T.; Brintrup, A.; and Zhu, J. 2021. Understanding softmax confidence and uncertainty. *arXiv preprint arXiv:2106.04972*.

Picot, M.; Noiry, N.; Piantanida, P.; and Colombo, P. 2023a. Adversarial Attack Detection Under Realistic Constraints.

Picot, M.; Staerman, G.; Granese, F.; Noiry, N.; Messina, F.; Piantanida, P.; and Colombo, P. 2023b. A Simple Unsupervised Data Depth-based Method to Detect Adversarial Images.

Raghuram, J.; Chandrasekaran, V.; Jha, S.; and Banerjee, S. 2021. A general framework for detecting anomalous inputs to dnn classifiers. In *International Conference on Machine Learning*, 8764–8775. PMLR.

Ramsay, K.; Durocher, S.; and Leblanc, A. 2019. Integrated rank-weighted depth. *Journal of Multivariate Analysis*, 173: 51–69.

Ruder, S. 2022. The State of Multilingual AI. http://ruder.io/state-of-multilingual-ai/.

Sanh, V.; Debut, L.; Chaumond, J.; and Wolf, T. 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Socher, R.; Perelygin, A.; Wu, J.; Chuang, J.; Manning, C. D.; Ng, A.; and Potts, C. 2013. Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, 1631–1642. Seattle, Washington, USA: Association for Computational Linguistics.

Srinivasan, A.; Sitaram, S.; Ganu, T.; Dandapat, S.; Bali, K.; and Choudhury, M. 2021. Predicting the performance of multilingual nlp models. *arXiv preprint arXiv:2110.08875*.

Techapanurak, E.; Suganuma, M.; and Okatani, T. 2019. Hyperparameter-free out-of-distribution detection using softmax of scaled cosine similarity. *arXiv preprint arXiv:1905.10628*.

van Esch, D.; Lucassen, T.; Ruder, S.; Caswell, I.; and Rivera, C. 2022. Writing system and speaker metadata for 2,800+ language varieties. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, 5035–5046.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Wang, H.; Li, Z.; Feng, L.; and Zhang, W. 2022a. ViM: Out-Of-Distribution with Virtual-logit Matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4921–4930.

Wang, H.; Zhao, C.; Zhao, X.; and Chen, F. 2022b. Layer Adaptive Deep Neural Networks for Out-of-Distribution Detection. In *Advances in Knowledge Discovery and Data Mining: 26th Pacific-Asia Conference, PAKDD 2022, Chengdu, China, May 16–19, 2022, Proceedings, Part II*, 526–538. Springer.

Williams, A.; Nangia, N.; and Bowman, S. 2018. A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 1112–1122. New Orleans, Louisiana: Association for Computational Linguistics.

Yang, J.; Zhou, K.; Li, Y.; and Liu, Z. 2021. Generalized Out-of-Distribution Detection: A Survey.

Yang, Y.; Zhang, Y.; Tar, C.; and Baldridge, J. 2019. PAWS-X: A Cross-lingual Adversarial Dataset for Paraphrase Identification. In *Proc. of EMNLP*.

Zhang, Q.; Shen, X.; Chang, E.; Ge, J.; and Chen, P. 2022. MDIA: A Benchmark for Multilingual Dialogue Generation in 46 Languages. *arXiv preprint arXiv:2208.13078*.

Zhou, W.; Liu, F.; and Chen, M. 2021. Contrastive Out-of-Distribution Detection for Pretrained Transformers. *arXiv preprint arXiv:2104.08812*.

Zotova, E.; Agerri, R.; Nuñez, M.; and Rigau, G. 2020. Multilingual Stance Detection in Tweets: The Catalonia Independence Corpus. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, 1368–1375. Marseille, France: European Language Resources Association. ISBN 979-10-95546-34-4.