# Detecting Textual Out-Of-Distribution Examples with Pretrained Transformers

**Julien Mereau***
ENSAE
julien.mereau@ensae.fr

**Agathe Minaro***
ENSAE
agathe.minaro@ensae.fr

## Abstract

Modern deep learning models have the capability to train large datasets effectively for Natural Language Processing (NLP) tasks, resulting in exceptional performance. However, these models are prone to susceptibility when there is a distribution shift between the training and the application data. Hence, it is imperative to develop methods for identifying data that does not conform to the same distribution. This article provides an overview of the out-of-distribution issue and outlines various detection methods, beginning with their mathematical underpinnings. Additionally, the `DistilBERT` model is modified to incorporate diverse aggregations. Finally, the detectors are applied to multiple datasets and aggregations to make a comparative analysis.

## 1 Introduction

### 1.1 State-of-the-art

With the development of Deep Learning, NLP models are more and more present and use more and more important datasets. Nevertheless, these training datasets have their own distribution in the textual space. Applying these models to datasets that do not have the same distribution as the one that made it possible to fit the model is one of the major vulnerabilities of NLP (Picot et al., 2023a,b). This problem is named Out-Of-Distribution (OOD) detection (Darrin et al., 2023a; Gomes et al.; Darrin et al., 2023b).

In this article, we use an already built classifier for textual data named `DistilBERT` (Sanh et al., 2019). The first OOD detectors mainly used the output of the last layer, called logits, which correspond to scores assigned to each class that can then be transformed into a probability assigned to each class. With these logits, first simplistic detectors have been created like the *likelihood* (Gangal et al., 2019; Choi et al., 2018), the *max softmax* (Hendrycks and Gimpel, 2018), the *entropy score* (Kuan and Mueller, 2022) and the *energy score* (Zhou et al., 2021). Some are more complex like the Mahalanobis distance (Fort et al., 2021).

The development of deeper metrics has prompted researchers to create more complex but also more accurate detectors. First, by applying the Mahalanobis distance to deep layer outputs (Serfling and Zuo, 2000), then using metrics specific to the depth of the models such as the *Integrated Rank-Weighted depth* (Ramsay et al., 2019; Staerman et al., 2021).

### 1.2 Our contribution

We have first realized the state of the art of the main OOD detectors by quickly re-explaining their concept and their mathematical basis in order to provide a global understanding of the different solutions that can be used to solve OOD problems. Then, we apply these detectors to different datasets, both to understand how to apply them and also to compare these different detectors and determine the most efficient.

Moreover, we make our code available on github[1] in order to be able to reuse these OOD detectors when using the `DistilBERT` model. In this context, we have made a modification to the initial code of the *transformers* package in order to have direct access to the outputs of the intermediate layers of `DistilBERT` and to be able to concatenate them directly if needed. Particular care has been taken to optimize the computation time and the memory needed to perform these calculations.

[1] https://github.com/Julien2048/NLP_Project_OOD.git

## 2 Problem Framing

**Notation.** We denote by $\mathcal{X}$ the textual input space and by $\mathcal{Y}$ the target space for a multiclass model. We also denote by $\mathbb{X}_{\text{train}}$ the dataset used to fit the model. In the following, $x$ represents a sample and $z$ the classification of $x$ between OOD ($z = 0$) and IN ($z = 1$). In a similar way, we have $\mathbb{X}$ and $\mathbb{Z}$ when we study a dataset.

**Score function.** Our goal is to build a score function $s : \mathcal{X} \to \mathbb{R}$ which accounts for the proximity of samples with the training dataset. Then, to classify samples between OOD and IN, we introduce a threshold $\gamma$ such that the sample $x$ is OOD if $s(x) \leq \gamma$ (i.e. $\hat{z} = 0$) and IN if $s(x) > \gamma$ (i.e. $\hat{z} = 1$).

**Performance evaluation.** The OOD problem is then a classification problem and to measure the performance of our detectors we use the *true detection rate* which is the proportion of samples that are OOD and that are classified as OOD and the *false alarm rate* which is the proportion of samples that are predicted OOD when they are IN.

We use two metrics to compare the different OOD detectors, the first one is the *Area Under the Receiver Operating Characteristic curve* (`AUROC`) which is the area under the ROC curve computed as follows: $\gamma \to (P(\hat{\mathbb{Z}} = 0|\mathbb{Z} = 0), P(\hat{\mathbb{Z}} = 1|\mathbb{Z} = 1))$. Intuitively, this metric represents the probability that the score of an IN sample is higher than the score of an OOD sample.

The second metric is the *Area Under the Precision-Recall curve* (`AUPR`) which represents the area under the precision-recall curve : $\gamma \to (P(\hat{\mathbb{Z}} = 1|\mathbb{Z} = 1), P(\mathbb{Z} = 1)|\hat{\mathbb{Z}} = 1))$.

## 3 OOD Detectors

### 3.1 From logits

Logits are the last element of the NLP model and allow us to assimilate a score to each class of prediction, which we denote by $F_L(x) \in \mathbb{R}^{card(\mathcal{Y})}$ the logits of the sample $x$. To get a probability for each class we then use a `softmax` function.

**Max Softmax.** This method is applied to many datasets in the article of (Hendrycks and Gimpel, 2018). Intuitively, if the sample is in the distribution of the dataset train, then the model will classify it in a class with a high probability. Conversely, if the sample is OOD, then it will be difficult to classify it and no class will really stand out. We then use the score function $s_{MSM}(x) = \max \texttt{softmax}(F_L(x))$. If this score is too low, then there is no class that stands out from the others and the sample is OOD.

**KL Divergence.** Similar to the above function, to best study the separations between classes, we use the Kullback-Leibler (`KL`) divergence. The score function becomes $s_{KL}(x) = \texttt{KL}(\texttt{softmax}(F_L(x)))$. A large divergence between the probabilities of the classes indicates that one class stands out from the others and therefore the sample is IN, conversely a divergence that is too small means that the probabilities are more or less at the same level and therefore the sample is classified as OOD.

As we will see later, these detectors allow a good classification of the samples between OOD and IN. Nevertheless, they only use the last layer of the NLP models. It would be interesting to introduce detectors that use more intermediate layers to determine OODs. This is what we will see in the following.

### 3.2 From prelogits or other hidden states

Overall, we will use the outputs of some intermediate layers of the model. More precisely, we will use the prelogits (output before the last layer - for the Mahalanobis distance) and a special aggregation of hidden layers (explained after - for the IRW). In what follows, we note $F(x)$ a vector of size $d$, the output of one of these layers for the sample $x$. In the case of the `DistilBERT` model $F(x)$ is a vector of size 768.

**Mahalanobis Distance.** The use of the Mahalanobis distance in OOD problems is largely detailed in this article (Fort et al., 2021). For any output $F$ of our model, we can use the Mahalanobis distance as a score function with :

$$s_{MD}(x) = (F(x) - \mu)^T \Sigma^{-1} (F(x) - \mu),$$

where $\mu = \mathbb{E}(F(\mathbb{X}_{\text{train}}))$ and $\Sigma = \text{cov}(F(\mathbb{X}_{\text{train}}))$. Intuitively, this distance represents the deviation from the mean of the training dataset with respect to the output of an intermediate layer.

**Integrated Rank-Weighted (IRW).** Introduced in (Ramsay et al., 2019) and developed in (Staerman et al., 2021), the IRW depth corresponds

to a need to have metrics specific to data depth. To have a good approximation of the IRW depth (Colombo et al., 2022a) we first need to introduce $\mathbb{S}^{d-1}$ the unit sphere, $\forall k \in [1, n_{\text{proj}}], u_k \in \mathbb{S}^{d-1}$ with $n_{\text{proj}}$ the number of direction sampled in the sphere (high number in practice) and $(x_i)_{1,...,n}$ the training dataset. We also defined $h_k^+(x) = \frac{1}{n}\sum_{i=1}^{n} \mathbb{I}(\langle u_k, F(x_i) - F(x)\rangle > 0)$ and $h_k^-(x) = \frac{1}{n}\sum_{i=1}^{n} \mathbb{I}(\langle u_k, F(x_i) - F(x)\rangle \leq 0)$. A good approximation of the IRW depth which we use as a score function is then defined as :

$$s_{IRW}(x) = \frac{1}{n_{proj}} \sum_{k=1}^{n_{proj}} min\big(h_k^+(x), h_k^-(x)\big)$$

This computation is really time-consuming in practice because its complexity is in $\mathcal{O}(n_{proj} n d)$.

## 4 Experiments Protocol

This section presents the settings used in the project, by describing the datasets, the pre-trained models and the methods.

### 4.1 Datasets selection

Datasets are responsible for any result in OOD detection. Thus, the benchmarks must be carefully chosen, because we can not expect good results on any datasets. We rely on the research from (Zhou et al., 2021; Colombo et al., 2022a; Hendrycks and Gimpel, 2018). Two types of in-distribution were chosen: sentiment analysis and topic classification.

For the sentiment analysis, the in-distribution was IMDB (Maas et al., 2011), a dataset providing polarized movie reviews with 10 000 sampled training and 400 sampled test data. For each one of the out-distributions, we kept 400 test occurrences. We chose SST2 (Socher et al., 2013), a corpus based on sentences extracted from movie reviews, Movie Review (Pang et al., 2002), a dataset where reviews came from professional rather than amateur, MNLI (Williams et al., 2017), a collection of sentences with textual entailment annotations, we use both matched and mismatched test sets, RTE (Dagan et al., 2005; Bar-Haim et al., 2006; Giampiccolo et al., 2007; Bentivogli et al., 2009), a series of annual textual entailment challenges. Note that IMDB, Movie Review, and SST2 belong to the same task and then can not be considered as OOD to each other.

For the topic classification, we took 20Newsgroup (Lang, 1995), a dataset for topic classification with 20 classes, as in and out distribution, the first 15 categories for in and the 5 left for out.

### 4.2 Pre-trained model

**Model.** We used the model `DistilBERT` (Sanh et al., 2019), pre-trained first from "*DistilBERT-base-uncased*". Then, we trained it (Colombo et al., 2022a) with batch-size of 32, weight decay set to 0.01, warmup ratio of 0.06, and learning rate of $10^{-5}$, over 4 epochs, over the in-distribution chosen before (one model for both).

**Aggregation procedures.** We change the baseline of the `DistilBERT` model from the `transformers` package in order to get several aggregations.

First, we create a class to return both pre-logits and logits of the model. The pre-logits correspond to the output of the pre-classifier of `DistilBERT`, while the logits are obtained after the pre-logits thanks to a `ReLU`, followed by a dropout and the classifier.

Then, we create another class to return an aggregation of the latent representations of $x$. Thanks to the output's attention of the model, we take the last representation where the attention is different from zero. Then, we took the mean of each layer on it to obtain what we call after $F_{PM}$ (Colombo et al., 2022a). Rigorously, we have: $F_{PM}(x) = \frac{1}{L}\sum_{l=1}^{L} \phi_l(x)$, where $\{\phi_1, ..., \phi_L\}$ are the layers of the model.

**OOD methods.** We use the four detectors explained before. Thanks to the logits, we compute the *Maximum Softmax* and the *Kullback-Leibler Divergence*. With the pre-logits, the *Mahalanobis Distance* is obtained. Finally, we use the *Integrated Rank Weighted Depth* on the aggregation of the hidden states obtained. Due to a lack of power in our systems, we use only 1000 training data, 100 test data, and a number of directions sampled on the sphere equal to 768 (the size of the output).

## 5 Results

In this section, we analyse the results obtained, depending on the in-distribution used, summarised in table 1. As expected, as Movie Review, SST2 and IMDB came from the same task, and can not

be considered as OOD, it is very difficult for the detector to perform good scores. Similarly, for the 20Newsgroup, given that the in and out distribution came from the same dataset, even if it is from different labels, it is quite impossible for `DistilBERT` to differentiate both. However, even if the scores of the softmax and the KL are bad for SST2, meaning that `DistilBERT` succeeds to be enough certain with one or another label for the test, the Mahalanobis score and the IRW seem to get good scores.

| **IMDB** | | | | |
|---|---|---|---|---|
| Out | Scores | Aggreg. | AUROC | AUPR |
| Movie Review | MaxSoftmax | Logits | 60.8% | 57.2% |
| | KL | Logits | 60.8% | 57.2% |
| | Mahalanobis | Pre-Logits | **64.4**% | 61.3% |
| | IRW | $F_{PM}$ | 53.8% | **61.5**% |
| MNLI | MaxSoftmax | Logits | 91.8% | 93.3% |
| | KL | Logits | 91.8% | 93.3% |
| | Mahalanobis | Pre-Logits | **97.6**% | **98.6**% |
| | IRW | $F_{PM}$ | 95.0% | 95.6% |
| SST2 | MaxSoftmax | Logits | 61.2% | 54.7% |
| | KL | Logits | 61.2% | 54.7% |
| | Mahalanobis | Pre-Logits | **97.1**% | **98.3**% |
| | IRW | $F_{PM}$ | 92.9% | 93.2% |
| RTE | MaxSoftmax | Logits | 94.9% | 96.2% |
| | KL | Logits | 94.9% | 96.2% |
| | Mahalanobis | Pre-Logits | **97.1**% | **98.3**% |
| | IRW | $F_{PM}$ | 94.2% | 95.5% |
| **20Newsgroup** (15/5) | | | | |
| Out | Scores | Aggreg. | AUROC | AUPR |
| | MaxSoftmax | Logits | 61.1% | 68.4% |
| | KL | Logits | **62.4**% | **68.8**% |
| | Mahalanobis | Pre-Logits | 57.8% | 57.8% |
| | IRW | $F_{PM}$ | 29.9% | 44.2% |

Table 1: OOD detection performance per IN-DS.

However, unlike (Colombo et al., 2022a), our Trusted/IRW score did not have the best results. Due to the lack of power of our systems, we can not run it on 400 samples of the test sets of others, the entire training set, and we could not set the number of directions sampled on the sphere to $768 \times \{\text{number of samples of the test set}\}$.

Figure 1 summarises the results of each OOD detector for the IMDB (IN-DS) and MNLI (OUT-DS) datasets. As we can see, the maximum softmax and the KL divergence gave the same score, however, the KL divergence is a much more visual graph than the softmax. The Mahalanobis score is the best due to the lack of samples for the IRW, which should give the best results according to the literature. The lack of examples can be seen in the non-smoothness of the curve, unlike the other scores.

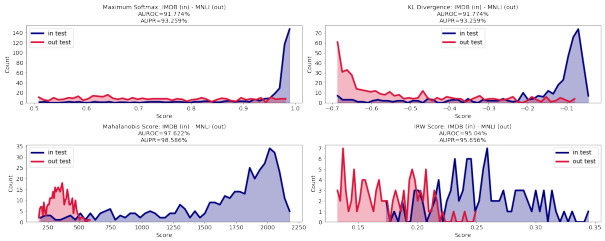

Figure 1: Summary of results: IMDB (IN-DS) - MNLI (OUT-DS)

## 6 Discussion and Future Work

As shown in the results, if the datasets come from the same kind of tasks, it can be very difficult to distinguish the in from the out samples. OOD detection does not seem to be universal, some detectors are more suitable for some tasks than others.

To get better results, the training of the encoder can be done on more epochs. Moreover, the computation of the scores has to be done on more data above all for the IRW one.

As the performance of OOD detectors is highly correlated with the choice of the model, the hyperparameters of the training, the datasets, and the aggregation methods, we would like to deeper the possibilities in future research.

## 7 Conclusion

This work presents several out-of-distribution detectors for pre-trained transformers, precisely using `DistilBERT`. We show the great improvement of using Mahalanobis distance on the pre-logits instead of just the logits. However, currently, combining out-of-distribution detection with fairness (Colombo, 2021; Colombo et al., 2021; Pichler et al., 2022; Colombo et al., 2022b) is an open problem in deep learning. In the future, we aim to address this issue by exploring the potential of using the IRW distance merged with the mean aggregation of the hidden model states, which may require additional computational resources to showcase its superiority. We believe that the combination of out-of-distribution detection and fairness will be crucial for creating trustworthy and unbiased AI models in the future.

Concerning ethical considerations, this work, even if it does not directly impact society, would allow

detecting when an input is different from the training data. This would increase the reliability of models in today's real world. Moreover, all experiments were performed on open datasets.

## References


Eduardo Dadalto Câmara Gomes, Pierre Colombo, Guillaume Staerman, Nathan Noiry, and Pablo Piantanida. A functional perspective on multi-layer out-of-distribution detection.

Ken Lang. 1995. Newsweeder: Learning to filter netnews. pages 331–339.

Robert Serfling and Yijun Zuo. 2000. General notions of statistical depth function. *The Annals of Statistics*, 28(2):461 – 482.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, pages 79–86. Association for Computational Linguistics.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2005. The pascal recognising textual entailment challenge. pages 177–190.

Roy Bar-Haim, Ido Dagan, Bill Dolan, Lisa Ferro, Danilo Giampiccolo, Bernardo Magnini, and Idan Szpektor. 2006. The second pascal recognising textual entailment challenge.

Danilo Giampiccolo, Bernardo Magnini, Ido Dagan, and Bill Dolan. 2007. The third PASCAL recognizing textual entailment challenge. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 1–9, Prague. Association for Computational Linguistics.

Luisa Bentivogli, Bernardo Magnini, Ido Dagan, Hoa Trang Dang, and Danilo Giampiccolo. 2009. The fifth PASCAL recognizing textual entailment challenge. In *Proceedings of the Second Text Analysis Conference, TAC 2009, Gaithersburg, Maryland, USA, November 16-17, 2009*. NIST.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.

Adina Williams, Nikita Nangia, and Samuel R. Bowman. 2017. A broad-coverage challenge corpus for sentence understanding through inference.

Dan Hendrycks and Kevin Gimpel. 2018. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *ArXiv*, abs/1610.02136.

Hyunsun Choi, Eric Jang, and Alexander A. Alemi. 2018. Waic, but why? generative ensembles for robust anomaly detection.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter.

Kelly Ramsay, Stéphane Durocher, and Alexandre Leblanc. 2019. Integrated rank-weighted depth. *Journal of Multivariate Analysis*, 173:51–69.

Varun Gangal, Abhinav Arora, Arash Einolghozati, and S. Gupta. 2019. Likelihood ratios and generative classifiers for unsupervised out-of-domain detection in task oriented dialog. In *AAAI Conference on Artificial Intelligence*.

Guillaume Staerman, Pavlo Mozharovskyi, and Stéphan Clémençon. 2021. Affine-invariant integrated rank-weighted depth: Definition, properties and finite sample analysis.

Wenxuan Zhou, Fangyu Liu, and Muhao Chen. 2021. Contrastive out-of-distribution detection for pretrained transformers. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Stanislav Fort, Jie Ren, and Balaji Lakshminarayanan. 2021. Exploring the limits of out-of-distribution detection.

Pierre Colombo. 2021. *Learning to represent and generate text using information measures*. Ph.D. thesis, (PhD thesis) Institut polytechnique de Paris.

Pierre Colombo, Chloe Clavel, and Pablo Piantanida. 2021. A novel estimator of mutual information for learning to disentangle textual representations. *() ACL 2021*.

Johnson Kuan and Jonas Mueller. 2022. Back to the basics: Revisiting out-of-distribution detection baselines.

Pierre Colombo, Eduardo D. C. Gomes, Guillaume Staerman, Nathan Noiry, and Pablo Piantanida. 2022a. Beyond mahalanobis-based scores for textual ood detection.

Georg Pichler, Pierre Jean A Colombo, Malik Boudiaf, Günther Koliander, and Pablo Piantanida. 2022. A differential entropy estimator for training neural networks. In *() ICML 2022*.

Pierre Colombo, Guillaume Staerman, Nathan Noiry, and Pablo Piantanida. 2022b. Learning disentangled textual representations via statistical measures of similarity. *() ACL 2022*.

Marine Picot, Nathan Noiry, Pablo Piantanida, and Pierre Colombo. 2023a. Adversarial attack detection under realistic constraints.

Maxime Darrin, Pablo Piantanida, and Pierre Colombo. 2023a. Rainproof: An umbrella to shield text generators from out-of-distribution data. *arXiv preprint arXiv:2212.09171*.

Marine Picot, Guillaume Staerman, Federica Granese, Nathan Noiry, Francisco Messina, Pablo Piantanida, and Pierre Colombo. 2023b. A simple unsupervised data depth-based method to detect adversarial images.

Maxime Darrin, Guillaume Staerman, Eduardo Dadalto Câmara Gomes, Jackie CK Cheung, Pablo Piantanida, and Pierre Colombo. 2023b. Unsupervised layer-wise score aggregation for textual ood detection. *arXiv preprint arXiv:2302.09852*.