Improved Multi-label Classification under Temporal Concept Drift
Rethinking Group-Robust Algorithms in a Label-Wise Setting
Chalkidis, Ilias; Sogaard, Anders

Published in:
FINDINGS OF THE ASSOCIATION FOR COMPUTATIONAL LINGUISTICS (ACL 2022)

Publication date:
2022

Document version
Peer reviewed version

Document license:
CC BY

Citation for published version (APA):
Improved Multi-label Classification under Temporal Concept Drift: Rethinking Group-Robust Algorithms in a Label-Wise Setting

Ilias Chalkidis and Anders Søgaard
Department of Computer Science, University of Copenhagen, Denmark [ilias.chalkidis,soegaard]@di.ku.dk

Abstract

In document classification for, e.g., legal and biomedical text, we often deal with hundreds of classes, including very infrequent ones, as well as temporal concept drift caused by the influence of real world events, e.g., policy changes, conflicts, or pandemics. Class imbalance and drift can sometimes be mitigated by resampling the training data to simulate (or compensate for) a known target distribution, but what if the target distribution is determined by unknown future events? Instead of simply resampling uniformly to hedge our bets, we focus on the underlying optimization algorithms used to train such document classifiers and evaluate several group-robust optimization algorithms, initially proposed to mitigate group-level disparities. Reframing group-robust algorithms as adaptation algorithms under concept drift, we find that Invariant Risk Minimization and Spectral Decoupling outperform sampling-based approaches to class imbalance and concept drift, and lead to much better performance on minority classes. The effect is more pronounced the larger the label set.

1 Introduction

Large-scale multi-label document classification is the task of assigning a subset of labels from a large predefined set – of, say, hundreds or thousands of labels – to a given document. Common applications include labeling scientific publications with concepts from ontologies (Tsatsaronis et al., 2015), associating medical records with diagnostic and procedure labels (Johnson et al., 2017), pairing legislation with relevant legal concepts (Mencia and Fürnkranz, 2007), or categorizing product descriptions (Lewis et al., 2004). The task in general presents interesting challenges due to the large label space and two-tiered skewed label distributions.

Class Imbalance In multi-label classification, datasets often exhibit class imbalance, i.e., skewed label distributions (Figure 2). Common methods include resampling and reweighting based on heuristic assumptions, but methods are known to suffer from unstable performance, poor applicability, and high computational cost in complex tasks where their assumptions do not hold (Liu et al., 2020). Datasets with long-tail frequency distributions, like the ones considered below – sometimes referred to as power-law datasets (Rubin et al., 2012) – can be particular challenging. Also, the heuristics fix the trade-off between exploiting as much of the training data as possible and balancing the classes, instead of trying to learn the optimal trade-off.

Temporal Concept Drift Moreover, class distributions may change over time. This is one dimension of the temporal generalization problem (Lazaridou et al., 2021). Recently, Søgaard et al. (2021) argued chronological data splits are necessary to estimate real-world performance, contrary to random splits (Gorman and Bedrick, 2019), because random splits artificially removes drift. Temporal concept drift, which we focus on here – in-
Figure 2: Label distributions of all datasets (UK-LEX, EUR-LEX, BIOASQ) and settings (small and medium sized label sets). Labels (bars) are ranked from most (left) to least (right) represented in the training set. Class imbalance across labels in the x-axis and temporal concept drift across subsets depicted with different coloured bars in the y-axis, i.e., a higher misalignment of color bars denotes a higher label distribution swift.

stead of covariate shift (Shimodaira, 2000), for example – is an instance of concept drift (Gama et al., 2014), often discussed in the domain adaptation literature, e.g., Chan and Ng (2006).

2 Related Work

Temporal Drift Temporal drift has been studied in several NLP tasks, including document classification (Huang and Paul, 2018, 2019), sentiment analysis (Lukes and Søgaard, 2018), Named Entity Recognition (NER) (Rijhwani and Preotiuc-Pietro, 2020), Neural Machine Translation (NMT) (Levenberg et al., 2010) and Language Modelling (Lazariadou et al., 2021). None of these papers focus on class imbalance and temporal concept drift. These papers have mainly been diagnostic, not providing technical solutions that are applicable in our case.

Multi-label Class Imbalance Class imbalance in (large-scale) multi-label classification has so far been studied through the lens of network architectures, searching for the best neural architecture for handling few- and zero-shot labels in the multi-label setting. To improve the performance for underrepresented (few-shot) classes, Snell et al. (2017) introduced Prototypical Networks that average all instances in each class to form prototype label vectors (encodings), a form of inductive bias, which improved few-shot learning. In a similar direction, Mullenbach et al. (2018) developed the Label-Wise Attention Network (LWAN) architecture, in which label-wise document representations are learned by attending to the most informative words for each label, using trainable label encodings (representations). Rios and Kavuluru (2018) extended LWAN and the idea of prototype label encodings. They combined label descriptors with information from a graph convolutional network (Kipf and Welling, 2017) that considered the relations of the label hierarchy to improve the results in few-shot and zero-shot settings. Alternatives to LWAN were considered by Chalkidis et al. (2020a), presenting minor improvements in the few-shot setting, but harming the overall performance.

Robustness The literature on inducing robust models from skewed data is rapidly growing. See Koh et al. (2021) for a recent survey. The group-robust learning algorithms we adapt and evaluate, e.g., Group Distributionally Robust Optimization (Sagawa et al., 2020), are discussed in detail in Section 4. Recent studies targeting fairness show that class imbalance has connections to bias (Blakeney et al., 2021; Subramanian et al., 2021), i.e., mitigating class-wise disparities has a chain effect on lowering group-wise disparities.

Main Contributions We focus on (large-scale) multi-label document classification and study a fundamental component of the learning process leading to performance disparities across labels, i.e., the underlying optimization algorithm used for training. We consider group-robust optimization algorithms initially proposed to mitigate group disparities given specific attributes (e.g., gender, race),
Table 1: Main characteristics of the examined datasets. We report the application domain, the number of documents, the available settings and the corresponding number of labels (used / total), and the label distribution swift between random and chronological splits using the Wasserstein Distance (WS) between train-test label probability distributions, i.e., \( WS_{\text{chronological}} = N \times WS_{\text{random}} \).

but re-frame these algorithms to optimize performance across labels rather than across groups.

3 Datasets

We experiment with three datasets (Table 1) from two domains (legal and biomedical), which support two different classification settings (label granularities), i.e., label sets including more abstract or more specialized concepts (labels).\(^1\)

**UK-LEX** United Kingdom (UK) legislation is publicly available as part of the United Kingdom’s National Archives.\(^2\) Most of the laws have been categorized in thematic categories (e.g., health-care, finance, education, transportation, planning) that are presented in the document preamble and are used for archival indexing purposes.

We release a new dataset, which comprises 36.5k UK laws (documents).\(^3\) The dataset is chronologically split in training (20k, 1975–2002), development (8.5k, 2002–2008), test (8.5k, 2008–2018) subsets. We manually extract and cluster the topics to support two different label granularities, comprising 18, and 69 topics (labels), respectively.

**EUR-LEX** European Union (EU) legislation is published in EUR-Lex.\(^4\) All EU laws are annotated by EU’s Publications Office with multiple concepts from EuroVoc, a thesaurus maintained by the Publications Office.\(^5\) EuroVoc has been used to index documents in systems of EU institutions, e.g., in web legislative databases, such as EUR-Lex and CELLAR, the EU Publications Office’s common repository of metadata and content.

\(^{1}\)We originally also considered the MIMIC-III dataset of Johnson et al. (2017) including discharge summaries from US hospitals annotated with ICD-9 medical codes, but the publication date of the documents has been “counterfeited” as part of the anonymization process. Experimental results with random splits are presented in Appendix B.1.

**BIOASQ** The BIOASQ (Task A: Large-Scale Online Biomedical Semantic Indexing) dataset (Tsatsaronis et al., 2015; Nentidis et al., 2021) comprises biomedical articles from PubMed,\(^7\) annotated with concepts from the Medical Subject Headings (MeSH).\(^8\) MeSH is a controlled and hierarchically-organized vocabulary produced by the National Library of Medicine. The current version of MeSH contains more than 29k concepts referring to various aspects of the biomedical research (e.g., Diseases, Chemicals and Drugs). It is used for indexing, cataloging, and searching of biomedical and health-related information, e.g., in MEDLINE/PubMed, and the NLM databases.

We use a subset of 100k documents derived from the latest version (v.2021) of the dataset.\(^9\) We subsample documents in the period 2000-2021, and we consider chronologically split training (80k, 1964–2015), development (10k, 2015–2018), test (10k, 2018–2020) subsets. We use the 1st and 2nd levels of MeSH, including 16 and 116 categories.

4 Fine-tuning Algorithms

In our experiments, we rely on pre-trained English language models (Devlin et al., 2019) and fine-tune these using different learning objectives. Our main goal during fine-tuning is to find a hypothesis \( (h) \) for which the risk \( R(h) \) is minimal:

\[^6\]The EUR-LEX dataset is available at https://hf.co/datasets/multi_eurlex.

\[^7\]https://www.nlm.nih.gov/mesh/


\[^9\]The BIOASQ dataset is available upon request at http://participants-area.bioasq.org/datasets.
\[ h^* = \arg \min_{h \in \mathcal{H}} R(h) \]  
(1)

\[ R(h) = \mathbb{E}[\mathcal{L}(h(x), y)] \]  
(2)

where \( y \) are the targets (ground truth) and \( h(x) = \hat{y} \) is the system hypothesis (model’s predictions).

Similar to previous studies, \( R(h) \) is an expectation of the selected loss function (\(\mathcal{L}\)). In this work, we study multi-label text classification (Section 3), thus we aim to minimize the binary cross-entropy loss across \( L \) classes:

\[ \mathcal{L}(x) = -y \log \hat{y} - (1 - y) \log(1 - \hat{y}) \]  
(3)

**ERM** (Vapnik, 1992), which stands for Empirical Risk Minimization, is the most standard and widely used optimization technique to train neural methods. The loss is calculated as follows:

\[ L_{\text{ERM}} = \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}(x_i) \]  
(4)

where \( N \) is the number of instances (training examples) in a batch, and \( \mathcal{L} \) is the loss per instance.

Furthermore, we consider a representative selection of group-robust fine-tuning algorithms that try to mitigate performance disparities with respect to a given attribute (\( A \)), e.g., in a standard scenario that could be the gender of a document’s author in sentiment analysis, or the background landscape in image classification. In our case, the attribute of interest is the labeling of the documents. The attribute is split into \( G \) groups, which in our case are the classes (\( G = L \)). All algorithms rely on a balanced group sampler, i.e., an equal number of instances (samples) per group (\( g_i \)) are included at each batch. Most of the algorithms are built upon group-wise losses (\( \mathcal{L}_{g_i} \)), computed as follows:

\[ \mathcal{L}(g_i) = \frac{1}{N_{g_i}} \sum_{j=1}^{N_{g_i}} \mathcal{L}(x_j) \]  
(5)

In our case, contrary to previous applications of group-robust algorithms, the groups (classes) are not mutually exclusive (documents are tagged with multiple labels). Hence, the group sampler can only guarantee that at least \( N \) groups (labels) will be considered at each step, but most probably even more. In this work, we examine the following group-robust algorithms in a label-wise fashion:

**Group Uniform** is the more naive group robust algorithm that uses the average of the group-wise (label-wise) losses - all groups (labels) are considered equally important, instead of the standard sample-wise average, as follows:

\[ L_{GM} = \frac{1}{G} \sum_{i=1}^{G} \mathcal{L}(g_i) \]  
(6)

**Group DRO** (Sagawa et al., 2020), stands for Group Distributionally Robust Optimization (DRO). Group DRO is an extension of the Group Uniform algorithm, where the group-wise (label-wise) losses are weighted inversely proportional to the group (label) performance. The total loss is calculated as follows:

\[ L_{\text{DRO}} = \sum_{i=1}^{G} w_{g_i} \cdot \mathcal{L}(g_i), \text{ where} \]  
(7)

\[ w_{g_i} = \frac{1}{W} (\hat{w}_{g_i} * e^{|\mathcal{L}(g_i)|}) \]  
(8)

where \( G \) is the number of groups (labels), \( \mathcal{L}_g \) are the averaged group-wise (label-wise) losses, \( w_g \) are the group (label) weights, \( \hat{w}_g \) are the group (label) weights as computed in the previous update step.

**V-REx** (Krueger et al., 2020), which stands for Risk Extrapolation, is yet another proposed group-robust optimization algorithm. Krueger et al. (2020) hypothesize that variation across training groups is representative of the variation later encountered at test time, so they also consider the variance across the group-wise (label-wise) losses. In V-REx the total loss is calculated as follows:

\[ L_{\text{VEX}} = L_{\text{ERM}} + \lambda \cdot \text{Var}(\mathcal{L}_{g_1}, \ldots, \mathcal{L}_{g_N}) \]  
(9)

where Var is the variance among the group-wise (label-wise) losses, and \( \lambda \), a weighting hyper-parameter scalar.

**IRM** (Arjovsky et al., 2020), which stands for Invariant Risk Minimization, mainly aims to penalize variance across multiple training dummy estimators across groups, i.e., performance cannot vary in samples that correspond to the same group. The total loss is computed as follows:

\[ L_{\text{IRM}} = \frac{1}{G} \left( \sum_{i=1}^{G} \mathcal{L}(g_i) + \lambda \cdot P(g_i) \right) \]  
(10)

where \( \mathcal{L}_{g_i} \) is the loss of the \( i \)th instance, which is part of the \( g_i \) group (label). Refer to Arjovsky et al. (2020) for a more detailed introduction of the group penalty terms (\( P_g \)).
Deep CORAL (Sun and Saenko, 2016), minimizes the difference in second-order statistics (covariances) between the source and target feature activations. In practice, it introduces group-pair penalties:

$$L_{CORAL} = L_{ERM} + \lambda \cdot \frac{1}{G} \sum_{i=1}^{G} P(g_i, g_{i+1})$$

$$P(g_i, g_{i+1}) = |C_{g_i} - C_{g_{i+1}}|^2 + |X_{g_i} - X_{g_{i+1}}|^2$$

where $C_{g_i}$ are the averaged covariances of the $i$th group and $X_{g_i}$ are the averaged features (document representations) of the $i$th group, respectively. Refer to Sun and Saenko (2016) for a more detailed introduction of the group penalty terms ($P_g$).

Spectral Decoupling (Pezeshki et al., 2020) relies on the idea of Gradient Starvation. Pezeshki et al. state that a network could become over-confident in its predictions by capturing only one or a few dominant features. Thus, adding an L2 penalty on the network’s logits ($\hat{y}_i$) provably decouples the fixed points of the dynamics. The total loss is computed as follows:

$$L_{SD} = L_{ERM} + \lambda \cdot \frac{1}{N} \sum_{i=1}^{N} \hat{y}_i^2$$

In our work, we consider the aforementioned algorithms in a label-wise setting, instead of a group-wise setting given a protected attribute. In our case, $G = L$, where $L$ is the number of labels.

### 5 Experimental SetUp

**Baseline Models** For both legal datasets (UK-LEX, EUR-LEX), we use the small LEGAL-BERT model of Chalkidis et al. (2020b), a BERT (Devlin et al., 2019) model pre-trained on English legal corpora. For BIOASQ, we use the small English BERT model of Turc et al. (2019). Following Devlin et al. (2019), we feed each document to the pre-trained model and obtain the top-level representation $h_{[cls]}$ of the special $[cls]$ token as the document representation. The latter goes through a dense layer of $L$ output units, one per label, followed by a sigmoid activation.

We also experiment with the Label-Wise Attention Network (LWAN) relying on a BERT encoder (Chalkidis et al., 2020a), dubbed BERT-LWAN. Chalkidis et al. reported state-of-art results in EUR-LEX and AMAZON-13K using BERT-LWAN compared to several baselines. BERT-LWAN uses one attention head per label to generate $L$ document representations $d_l$:

$$a_l = \frac{\exp(K(h_i)Q_l)}{\sum_r \exp(K(h_i)Q_r)}$$

$$d_l = \frac{1}{T} \sum_{t=1}^{T} a_l V(h_{t})$$

$T$ is the document length in tokens, $h_i$ the context-aware representation of the $t$-th token, $K$, $V$ are linear transformations of $h_i$, and $Q_l$ a trainable vector used to compute the attention scores of the $l$-th attention head; $Q_l$ can also be viewed as a label representation. Intuitively, each head focuses on possibly different tokens of the document to decide if the corresponding label should be assigned. BERT-LWAN employs $L$ linear layers ($\omega_l$) with sigmoid activations, each operating on a different label-wise document representation $d_l$, to produce the probability of the corresponding label $p_l$:

$$p_l = \text{sigmoid}(d_l \cdot \omega_l)$$

Across experiments, we use BERT models following a small configuration (6 transformer blocks, 512 hidden units and 8 attention heads), which allows us to increase the batch size up to 64 and consider samples with multiple labels (groups) in the group robust algorithms. In practice, this enables us to sample at least 4 samples per group (label) for all labels in the small label sets, and at least 1 sample per group (label) for 64 labels in the medium-sized label sets (69-112 labels).

**Training Details** We fine-tune all models using the AdamW (Loshchilov and Hutter, 2019) optimizer with a learning rate of 2e-5. We use a batch size of 64 and train models for up to 20 epochs using early stopping on the development set. We run three repetitions with different random seeds and report the test scores based on the seed with the best scores on development data. We report development scores on Appendix B.2.

**Evaluation Metrics** Given the large number and skewed distribution of labels, retrieval measures have been favored in large-scale multi-label text classification literature (Mullenbach et al., 2018; You et al., 2019; Chalkidis et al., 2020a). Following Chalkidis et al. (2020a), we report *mean R-Precision* (m-RP) (Manning et al., 2009), while...
we also report the standard micro-$F_1$ ($\mu-F_1$) and macro-$F_1$ (m-$F_1$) to better estimate the class-wise performance disparity.

### Data and Code
In our experiments, we extend the WILDS (Koh et al., 2021) library, which provides an experimental framework for experimenting with group-robust algorithms. We effectively rewrote all parts of code to consider label-wise groups and losses, while we also implemented the unsupported methods (Group Uniform, V-REx, and Spectral Decoupling). For reproducibility and further exploration with new group-robust methods, we release our code on Github.11

### 6 Results

#### Main Results
To highlight the temporal concept drift, we initially fine-tune BERT in all datasets with the standard ERM optimization algorithm using both random and chronological splits. Table 3 shows that the real-world performance achieved using the chronological split is severely overestimated using the random split (approx. +10% across evaluation measures) in two out of three datasets. While all datasets have inherently skewed distributions (class imbalance), which is naturally demonstrated by the performance discrepancy between $\mu-F_1$ and m-$F_1$ scores (especially when we consider the larger label sets), the temporal dimension further exacerbate the performance discrepancy as label distributions also vary across subsets (Figure 2). Surprisingly, the performance discrepancy between chronological and random splits is much lower on BIOASQ (approx. 1-2%), which could be explained by the larger volume of training data (Table 1), and the very high representation for most of the labels in general (Figure 2).

In Table 2, we present the overall results for the different optimization algorithms considering the baseline model, BERT. We observe that using a group sampler (ERM+GS), which equals standard oversampling of minority classes, slightly improve the results in m-$F_1$ (+1-4%) in many cases, while the performance is comparable in $\mu-F_1$ and m-RP. Considering the results of group-robust algorithms, we observe that most of them improve m-$F_1$ across datasets compared to ERM and ERM+GS, +1-4% for small-sized datasets and +5-12% in medium-sized datasets. Again the performance in $\mu-F_1$ and m-RP is mostly comparable or a bit lower, as sample-wise averaged measures are dominated by frequent classes due to class imbalance.

Contrary, Group DRO is consistently outperformed even by the standard ERM. Recall that Group DRO uses a weighted average of the group-wise (label-wise) losses (Equation 7-8), where the group weights rely on the momentum of the group-wise (label-wise) losses (Equation 8). In our case, this regularization acts counter-intuitively, as weights for the infrequent classes, which are rarely present across batches, are not updated (decrease) constantly. This leads to an asymmetry, where some weights are frequently updated, while others not, and in time the latter are almost zeroed-out and not affect the training objective (loss).

The effect of group-robust algorithms in relation to the size of the label set. In Table 2, we can also observe that the performance gains of group-robust algorithms compared to ERM are greater when we use the larger label sets. This is also as the class imbalance and temporal concept drift are more severe when we consider more refined labels, especially considering m-$F_1$.

---

11https://github.com/coastalcph/lw-robust
Table 4: Test results of group-robust algorithms in head and tail classes in the medium-sized datasets. Head are the 50% most represented (frequent) classes in the training set, and tail are the bottom 50%.

![Image](image.png)

Figure 3: Class-wise F1-score results for ERM (blue), IRM (yellow) and Spectral Decoupling (red) on medium-sized EUR-LEX. The classes have been ordered (left-to-right) based on the label distribution in the training subset. Algorithms’ performance on the left part (head classes) is very much aligned, contrary to the right (tail labels).

The effect of group-robust algorithms in relation to class frequency. In Table 4, we present results for the different optimization algorithms considering two groups of classes based on their frequency. Head classes are the 50% most frequent classes in the training set, while tail are the bottom 50%. As expected, the performance in head classes is much better compared to tail ones across datasets (approx. +20-40% in m-F$_1$). We observe that the performance gains of group-robust algorithms compared to ERM are greater in the tail classes (+10-40% in m-F$_1$). This is further highlighted in Figure 3, where we observe that IRM and Spectral Decoupling, the two best performing group-robust algorithms, have larger gains in the right part (tail labels); in fact ERM scores zero in many cases (classes) where the two group-robust algorithms don’t. This is highly expected as the goal of the group-robust algorithms is to minimize the group-wise (in our case, label-wise) disparity. Group DRO is severely out-performed in both head and tail, especially in the tail classes (whose weights have been zeroed-out, as previously noticed).

Why IRM and Spectral Decoupling are a better fit compared to the rest of the algorithms? To answer this question, we need to identify the main differentiation between IRM, Spectral Decoupling and the rest of the methods. Both IRM and Spectral Decoupling follow similar incentives. IRM penalizes variance across losses in the same group (Equation 10), i.e., in our case, the network is penalized if there is a performance disparity between samples labeled with the same classes using as a reference a dummy classifier. Spectral Decoupling penalizes the variance across label predictions (Equation 13), i.e., the network is penalized for being over-confident. The rest of the algorithms mainly rely on an equal consideration of the group-wise (in our case, label-wise) losses (Equation 6), i.e., in our case, all classes are equally important for the training objective.

The latter incentive (averaging across group-wise losses) seems very intuitive, although in practice the groups (labels) co-occur (are not mutually exclusive) in a multi-label setting, thus frequent labels remain “first class citizens” in the optimization process, biasing parameter updates in their favor.

Contrary, both IRM and Spectral Decoupling use a learning component (loss term), which penalizes label degeneration. This is particularly important in multi-label classification, especially when we consider large label sets, as networks tend to over-fit (specialize) in few dominant (frequent) labels that shape the training loss and finally ignore
We hypothesize that IRM and Spectral Decoupling we observe a performance drop (approx. 2%) in with a 10% improvement over the standard BERT. In this part, we compare the ef-
fícit in the medium-sized version of EUR-LEX. Deep CORAL is not applicable (n/a) in LWAN -there is not a universal featurizer.-

Table 5: Test results of group-robust algorithms with different models (BERT, and BERT-LWAN) in the medium-sized version of EUR-LEX. Deep CORAL is not applicable (n/a) in LWAN.

The effect of group-robust algorithms using BERT-LWAN. In this part, we compare the effect of the group-robust algorithms in between standard BERT and BERT-LWAN on the medium-sized EUR-LEX dataset. In Table 5, we observe that BERT-LWAN closes the gap between EUR and the best-of-group-robust algorithms. The results of EUR when we use BERT-LWAN are improved across measures, especially when we consider m-F1 with a 10% improvement over the standard BERT. Both IRM and Spectral Decoupling seem quite insensitive to the underlying model. Similarly, the results for the rest of the group-robust algorithms are improved. Nonetheless, there are still benefits in m-F1 and less represented (tail) labels in general. Interestingly, Spectral Decoupling improves results in m-F1, with comparable μ-F1 scores. Although, we observe a performance drop (approx. 2%) in m-RP when we consider overall and head classes. We hypothesize that IRM and Spectral Decoupling negatively affect the ability of the BERT-LWAN model to correctly rank labels (Equation 16), as they force the model to consider all labels by not being over-confident (discriminatory) with one way or another, as previously explained.

In Figure 4, we compare the performance of ERM, IRM, and Spectral Decoupling across three EUR-LEX settings, small-sized, medium-sized, and one extra large-sized considering the 3rd level of EuroVoc including 500 concepts (labels). In the small label set, we observe that the use of LWAN-BERT slightly improves the performance when trained with ERM compared to standard BERT (shaded part of the bars). In the medium label set, as already discussed, we observe an approx. 10% improvement with ERM, while in case of the large label set, using LWAN-BERT leads to an approx. 20% improvement with ERM (the performance of BERT is 0%), and 6.5% with Spectral Decoupling, while IRM proves to be remarkably robust across all settings and both neural methods (BERT with or without the LWAN component).

7 Alternative Combined Algorithm

Having a clear understanding of what IRM and Spectral Decoupling offer, it seems that we could combine both to leverage all features: (a) rely on group-wise (label-wise) losses as the main driver of the optimization process (Equation 6); (b) penalize the classifier if there is a performance disparity between samples labeled with the same classes (Equation 10); and (c) penalize the classifier for being over-confident (Equation 13).
We name the new algorithm **Label-Wise Distributional Robust Optimization**, LW-DRO in short, as it mainly aims to mitigate label-wise disparities, and investigate two alternatives (variants):

- In version 1 (v1), we combine the averaged group-wise (label-wise) losses (Equation 6) introduced with Group Uniform, with the Spectral Decoupling penalty (Equation 13). The total loss term ($\mathcal{L}_{\text{LW-DRO}}$), is computed as follows:

$$\frac{1}{G} \sum_{g=1}^{G} L(g_i) + \lambda \ast \frac{1}{N} \sum_{i=1}^{N} \hat{y}_i^2$$ (17)

- In version 2 (v2), we also include the group-wise penalties of IRM (Equation 10). The total loss term ($\mathcal{L}_{\text{LW-DRO}}$), is computed as follows:

$$\frac{1}{G} \left( \sum_{g=1}^{G} L(g_i) + \lambda_1 \ast P(g_i) \right) + \lambda_2 \ast \frac{1}{N} \sum_{i=1}^{N} \hat{y}_i^2$$ (18)

The notation used in Equations 17 and 18 follows the one presented in Section 4.

In Table 5, we observe that the second variant of LW-DRO (v2) has comparable or better performance compared to IRM and Spectral Decoupling, contrary to the first one (v1). LW-DRO (v2) is a straight forward combination of IRM and Spectral Decoupling, while LW-DRO (v1) that relies on a group-averaged loss under-performs, especially considering the m-F$_1$ scores. As previously explained, labels co-occur in a multi-label setting, hence averaging label-wise losses favors frequent classes and in turn limits the possible benefits in under-represented classes (perceived by m-F$_1$).

In Figure 4, we present the results of ERM, and the 3 overall best group-robust algorithms (IRM, Spectral Decoupling, and LW-DRO (v2)) across all EUR-LEX settings. LW-DRO (v2) has comparable performance in the first two settings (small, medium), while being slightly better than IRM in the large-sized setting. While LW-DRO (v2) seems to control the trade-offs between IRM and Spectral Decoupling, we believe that future work should better seek alternative directions with respect to algorithmic advances that possibly mitigate label degeneration and tackles label-wise disparities.

8 Conclusions & Future Work

We considered one of the main challenges in large-scale multi-label text classification, which comes from the fact that not all labels are well represented in the training set due to the class imbalance and the effect of temporal concept drift. To mitigate label disparities, we considered several group-robust optimization algorithms initially proposed to mitigate group disparities given specific attributes. Experimenting with three datasets in two different settings, we empirically find that group-robust algorithms vastly improve performance considering macro-averaged measures, while two of the group-robust algorithms (Invariant Risk Minimization and Spectral Decoupling) improve performance across all measures. Considering a more well-suited neural method (LWAN-BERT), we observe a vast performance improvement using ERM, leading to comparable overall results (m-F$_1$, m-RP) with the group-robust algorithms; although is still outperformed considering m-F$_1$. Lastly, based on our understanding of what IRM and Spectral Decoupling, the two best group-robust algorithms, offer, we introduced and evaluated a new algorithm, Label-Wise DRO, which combines features from both, and one of its variants has comparable or better performance considering larger label sets.

In the future, we would like to further investigate the two-tier anomaly (class imbalance and temporal concept drift). In this direction, we would like to directly take into consideration the time dimension by utilizing this information in group sampling and algorithms (e.g., groups over period of time). We would also like to consider data augmentation techniques (e.g., paraphrasing via masked-language modeling (Ng et al., 2020), and teacher forcing exploiting unlabeled data (Eisenschlos et al., 2019)) to improve the data (feature) sampling variability, as the group sampler used in group-robust algorithms over-sample minority classes with the same limited instances. Further on, we would like to investigate the use of zero-shot LWAN methods (Rios and Kavuluru, 2018; Chalkidis et al., 2020a), which currently harm averaged performance in favor of improved worst case performance. Label encodings based on contextualized word representations generated by pre-trained language models (Hardalov et al., 2021) may mitigate the effect of using non-contextualized ones (e.g., Word2Vec).

Acknowledgments

This work is funded by the Innovation Fund Denmark (IFD)\textsuperscript{12} under File No. 0175-00011A.

\textsuperscript{12}https://innovationsfonden.dk/en
References


João Gama, Indrundefined Žliobaitundefined, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. 2014. A survey on concept drift adaptation. ACM Comput. Surv., 46(4).


California. Association for Computational Linguistics.


A Measuring class-wise bias

Blakeney et al. (2021) recently introduced two evaluation measures to estimate class-wise bias of two models in comparison to one another in a multi-class setting, and show that these metrics can be also used to measure fairness and bias with respect to protected attributes.

Following Blakeney et al. (2021), in Figure 5 we present the normalized Combined Error Variance (CEV) in-between algorithms. CEV estimates the class-wise bias of a model A relative to another model B has increased of the change between model A and a random predictor. For a detailed analysis of the CEV metric, please refer to Blakeney et al. (2021).

In our case, as different models, we consider BERT trained with a different algorithm. In both UK-LEX and EUR-LEX, swapping Group Uniform, IRM, or Spectral Decoupling with ERM, or Group DRO leads to a higher class-wise bias, which is highly expected given the aforementioned performance analysis, i.e., improved m-F₁ scores.

B Additional Results

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Small m-F₁</th>
<th>μ-F₁</th>
<th>Medium m-F₁</th>
<th>μ-F₁</th>
</tr>
</thead>
<tbody>
<tr>
<td>ERM</td>
<td>71.8</td>
<td>60.2</td>
<td>47.4</td>
<td>10.3</td>
</tr>
<tr>
<td>ERM+GS</td>
<td>71.7</td>
<td>62.4</td>
<td>47.5</td>
<td>12.6</td>
</tr>
<tr>
<td>Group Uniform</td>
<td>71.9</td>
<td>66.1</td>
<td>48.2</td>
<td>13.3</td>
</tr>
<tr>
<td>Group DRO</td>
<td>65.2</td>
<td>47.4</td>
<td>47.5</td>
<td>3.8</td>
</tr>
<tr>
<td>Deep CORAL</td>
<td>72.1</td>
<td>67.1</td>
<td>47.1</td>
<td>12.3</td>
</tr>
<tr>
<td>V-REx</td>
<td>71.9</td>
<td>65.9</td>
<td>47.6</td>
<td>11.3</td>
</tr>
<tr>
<td>IRM</td>
<td>72.0</td>
<td>66.6</td>
<td>53.3</td>
<td>18.3</td>
</tr>
<tr>
<td>Spectral Decoupling</td>
<td>72.3</td>
<td>67.2</td>
<td>53.1</td>
<td>16.1</td>
</tr>
</tbody>
</table>

Table 6: Overall test results of the group-robust algorithms across on MIMIC-III dataset.

B.1 Experiments on MIMIC-III

MIMIC-III dataset (Johnson et al., 2017) contains approx. 50k discharge summaries from US hospitals. Each summary is annotated with one or more codes (labels) from the ICD-9 hierarchy, which has 8 levels. The International Classification of Diseases, Ninth Revision (ICD-9) is the official system of assigning codes to diagnoses and procedures associated with hospital utilization in the United States and is maintained by the World Health Organization (WHO).

MIMIC-III has been anonymized to protect patients privacy, including chronological information (e.g., entry/discharge dates). Hence, it is not possible to split data in chronological splits. We split the dataset randomly in training (30k), development (10k), test (10k) subsets. We use the 1st and 2nd level of ICD-9 including 19 and 184 categories, respectively.

In Table 6, we present the results, which lead to the very same observations discussed for the rest of the datasets.

B.2 Development Results

We run three repetitions with different random seeds and in the main article (Section 6, we report the test scores based on the seed with the best scores on development data. For completeness, in Tables 7, 8, 9, we report the development results of the group-robust (label-robust) algorithms across all datasets (UK-LEX, EUR-LEX, BIOASQ) and settings (small and medium sized label sets) using BERT. We report the mean and standard deviation (±) across all three examined seeds.

13www.who.int/classifications/icd/en/
Figure 5: Class-wise bias in-between algorithms across datasets, measured with the normalized Combined Error Variance (CEV) as defined by Blakeney et al. (2021).

Table 7: Overall $\mu$-F$_1$ development results of the group-robust (label-robust) algorithms across all datasets (UK-LEX, EUR-LEX, BIOASQ) and settings (small and medium sized label sets). We report the mean and standard deviation (±) across three seeds.

Table 8: Overall m-F$_1$ development results of the group-robust (label-robust) algorithms across all datasets (UK-LEX, EUR-LEX, BIOASQ) and settings (small and medium sized label sets). We report the mean and standard deviation (±) across three seeds.
## Table 9: Overall m-RP development results of the group-robust (label-robust) algorithms across all datasets (UK-LEX, EUR-LEX, BIOASQ) and settings (small and medium sized label sets). We report the mean and standard deviation (±) across three seeds.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>UK-LEX Small</th>
<th>UK-LEX Medium</th>
<th>EUR-LEX Small</th>
<th>EUR-LEX Medium</th>
<th>BIO-ASQ Small</th>
<th>BIO-ASQ Medium</th>
</tr>
</thead>
<tbody>
<tr>
<td>ERM</td>
<td>87.5 ± 0.5</td>
<td>77.2 ± 0.6</td>
<td>86.1 ± 0.0</td>
<td>75.5 ± 0.8</td>
<td>88.3 ± 0.0</td>
<td>71.0 ± 0.3</td>
</tr>
<tr>
<td>ERM+GS</td>
<td>88.7 ± 0.3</td>
<td>77.6 ± 0.4</td>
<td>86.5 ± 0.1</td>
<td>75.8 ± 0.6</td>
<td>89.4 ± 0.2</td>
<td>70.5 ± 0.1</td>
</tr>
<tr>
<td>Group Uniform</td>
<td>87.0 ± 0.4</td>
<td>80.1 ± 0.3</td>
<td>85.2 ± 0.6</td>
<td>75.7 ± 0.7</td>
<td>87.4 ± 0.1</td>
<td>70.1 ± 0.4</td>
</tr>
<tr>
<td>Group DRO</td>
<td>86.6 ± 0.3</td>
<td>75.0 ± 0.2</td>
<td>82.6 ± 0.4</td>
<td>69.7 ± 0.2</td>
<td>85.7 ± 0.2</td>
<td>43.1 ± 6.9</td>
</tr>
<tr>
<td>Deep CORAL</td>
<td>87.0 ± 0.3</td>
<td>78.3 ± 0.7</td>
<td>85.7 ± 0.1</td>
<td>75.7 ± 0.2</td>
<td>86.1 ± 0.3</td>
<td>70.9 ± 0.7</td>
</tr>
<tr>
<td>V-REx</td>
<td>87.5 ± 0.2</td>
<td>79.9 ± 0.6</td>
<td>85.5 ± 0.3</td>
<td>75.6 ± 0.0</td>
<td>87.4 ± 0.0</td>
<td>70.0 ± 0.4</td>
</tr>
<tr>
<td>IRM</td>
<td>87.2 ± 0.4</td>
<td>80.7 ± 0.3</td>
<td>85.3 ± 0.4</td>
<td>76.4 ± 0.8</td>
<td>87.4 ± 0.0</td>
<td>70.5 ± 0.4</td>
</tr>
<tr>
<td>SD</td>
<td>87.5 ± 0.1</td>
<td>81.1 ± 0.2</td>
<td>83.9 ± 0.2</td>
<td>76.8 ± 0.1</td>
<td>87.5 ± 0.0</td>
<td>72.6 ± 0.2</td>
</tr>
</tbody>
</table>