Improving Neural Political Statement Classification with Class Hierarchical Information

Erenay Dayanik[1], **André Blessing**[1], **Nico Blokker**[2], **Sebastian Haunss**[2], **Jonas Kuhn**[1], **Gabriella Lapesa**[1], and **Sebastian Padó**[1]

[1]IMS, University of Stuttgart, Germany
[2]SOCIUM, University of Bremen, Germany

Abstract

Many tasks in text-based computational social science (CSS) involve the classification of political statements into categories based on a domain-specific codebook. In order to be useful for CSS analysis, these categories must be fine-grained. The typically skewed distribution of fine-grained categories, however, results in a challenging classification problem on the NLP side. This paper proposes to make use of the *hierarchical relations among categories* typically present in such codebooks: e.g., *markets* and *taxation* are both subcategories of *economy*, while *borders* is a subcategory of *security*. We use these ontological relations as prior knowledge to establish additional constraints on the learned model, thus improving performance overall and in particular for infrequent categories. We evaluate several lightweight variants of this intuition by extending state-of-the-art transformer-based text classifiers on two datasets and multiple languages. We find the most consistent improvement for an approach based on regularization.

1 Introduction

The argumentative or discursive turn in policy analysis and political science more generally has long established the value of textual sources for the analysis of politics and policies (Fischer and Forester, 1993). Traditionally, data sources such as interviews or newspaper reports were annotated using various methods of qualitative text analysis (Wagenaar, 2011; Mayring, 2019). At the heart of this analysis is always a *codebook*, i.e., guidelines that map actual statements or textual passages to the abstract concepts relevant for the respective research.

Categories in codebooks are almost always arranged *hierarchically*, with *fine-grained* categories being grouped together into supercategories that are often, but not always, more abstract. Fine-grained categories are generally generated inductively from the analyzed texts in an iterative process of summarizing and abstracting from the original text, while the supercategories are deductively generated from existing knowledge of the relevant policy field and from theoretical and conceptual findings of prior research. For example, the codebook of the long-running Comparative Manifesto Project (CMP), which analyzes party manifestos across several countries, includes 7 supercategories (such as *external relations* or *economy*) with 56 subcategories: for *economy*, among others, *free market*, *market regulation*, *economic goals*, etc. (Merz et al., 2016; Werner et al., 2011). Here, supercategories represent the separation of policy fields that is reflected in political institutions, e.g., ministries. Fine-grained, hierarchical schemes help researchers both with data annotation and with analysis. Annotation is often easier when the annotation decision is (implicitly) first based on a supercategory and then on fine-grained subcategories. For analysis, supercategories structure the annotated material according to different levels of abstraction, thereby supporting interpretation and modeling.

While such a hierarchical process a natural choice in manual annotation, the situation is different when we move to (semi)-automatic analysis in NLP: due to the large number of fine-grained subcategories, the available data is distributed among many categories. In addition, most categories are infrequently attested, since categories typically show a skewed distribution. This makes for a difficult classification problem, and existing prediction studies have often only addressed the more coarse-grained supercategory level (Glavaš et al., 2017a; Subramanian et al., 2018; Padó et al., 2019).

In this study, we ask whether we can use the hierarchical structure of political science codebooks to our advantage: knowing that two subcategories (as *free market* and *market regulation*) belong to the same supercategory (*economy*) could lead us to expect that the representations learned for these categories should be more similar to one another

Findings of the Association for Computational Linguistics: ACL 2022, pages 2367 - 2382
May 22-27, 2022

than to categories that belong to other supercategories. In this manner, the representations learned for smaller categories can be biased in the right direction by their larger neighbor categories. This paper makes the following contributions:

- In Section 3, we define an ontology of lightweight methods implementing this intuition on top of a state-of-the-art transformer-based text classifier. Crucially, these methods introduce almost no additional parameters, thereby addressing the issues related to the limited amounts of annotated data typically available in CSS studies.
- We evaluate the resulting models on two datasets and five different languages, covering single label (Experiment 1) as well as multi label classification (Experiment 2). We establish that regularized methods yield consistent improvements and establish a new state of the art for political statement classification. In particular, these methods improve predictions on low-frequency categories, improving model fairness (Dayanik and Padó, 2020).

This paper builds on an earlier study of ours (Dayanik et al., 2021), whose scope is extended in multiple dimensions. At the phenomenon level, we broaden the focus from (forward-looking) political claims to (general) political statements. At the methodological level, we propose an ontology of methods for encoding hierarchical information. At the experimental level, we now take into consideration two text types involving five different languages. The code, models and dataset splits used in this study are available at `https://www.ims.uni-stuttgart.de/data/inpsc`.

2 Background and Related Work

Codebooks for Political Statement Categorization Codebooks used in large-scale annotation projects cover a broad variety of research interests and text types. Yet, regardless of whether they have been created to analyze political party manifestos (Volkens et al., 2020), political statements in the European public sphere (Koopmans, 2002), legitimation discourses about political and economic regimes (Nullmeier et al., 2015), or the migration debate in Germany (Blessing et al., 2019), they all group their categories of interest into a limited number of supercategories which reflect the existing research in the respective field.

Text Classification Automatic political statement classification is fundamentally a text classification task on relatively short texts, with the class inventory given by the codebook. Depending on the properties of the annotation, the task is either single-label or multi-label text classification. In single label text classification, each text is assigned exactly one label, which is used in NLP applications where the labels are mutually exclusive, such as in entailment or stance detection (Kim, 2014; Glavaš and Vulić, 2019; Kennedy et al., 2019; Li and Caragea, 2019). In contrast, multi-label text classification assigns any number of categories to a text, which is better suited for tasks where the categories are overlapping or describe complementary aspects, e.g. topic categorization (Rios and Kavuluru, 2018; Chalkidis et al., 2019; Irsan and Khodra, 2019; Xiao et al., 2019). Currently, transformer-based models (Devlin et al., 2019; Liu et al., 2020) represent the current state of the art for text classification in general (Minaee et al., 2021) and political statement classification in particular (Dayanik et al., 2021). A number of studies have investigated ways to integrate hierarchical information into classification. A first family of approaches develops dedicated architectures such as capsule networks (Aly et al., 2019) or encoders of the hierarchies (Song and Roth, 2014; Zhou et al., 2020). These models are typically trained end-to-end, which requires amounts of data that are rarely available in CSS. We focus on lightweight approaches compatible with fine-tuning, described in Section 3.

Political Statement Classification Political statement classification is a task in political text analysis, other examples of which are political text scaling (Glavaš et al., 2017b), political event detection (Nanni et al., 2017) or detection of frames (Card et al., 2015). Specific studies on political statement classification includes Verberne et al. (2014) who develop models for automatic categorization of political statements in Dutch and Karan et al. (2016) who assign topic labels to political texts in Croatian. A number of studies work with the abovementioned Comparative Manifesto Project dataset (Merz et al., 2016): Zirn et al. (2016) and Glavaš et al. (2017a) address coarse-grained text policy position analysis and Subramanian et al. (2018) introduce multilingual models jointly trained for coarse-grained statement classification and document-level positioning. In our own previous work, we created a corpus of

German newspaper articles on the 2015 refugee crisis, *DebateNet-mig15*, (Lapesa et al., 2020), and carried out coarse-grained classification experiments on the annotated statements regarding the migration policy (Padó et al., 2019).

3 Method

3.1 Base Classifier

In line with previous work in political statement classification, we focus on statement classification and assume that statements have already been detected (Subramanian et al., 2018; Padó et al., 2019). We use a standard pre-trained and fine-tuned BERT (Devlin et al., 2019) transformer as a state of the art base classifier.[1] Pre-trained BERT models are available for many languages and domains, and can be fine-tuned for text classification tasks with a simple fully-connected layer.[2]

Formally, the input consists of a word statement x; we do not consider the statement's context. BERT encodes the input into a representation, $e(x)$, which we obtain from the special token [CLS] prepended to the statement. In the single-label case, the classifier $c(e(x))$ predicts a single label using softmax activation (cf. Section 4). In the multi-label case, it predicts a set of labels using sigmoid activation (cf. Section 5). The objective function $\mathcal{L}_{\text{main}}$ is standard cross entropy loss.

3.2 Introducing Hierarchical Information

As mentioned in Section 2, we focus on lightweight methods that introduce a minimal number of additional parameters and are therefore compatible with fine-tuning as part of the final classification layer of a transformer-based architecture. The suitable methods are summarized in the taxonomy in Figure 1. We distinguish, from top to bottom: (1) Methods that post-process the output of a statement classifier to enforce hard constraints vs. methods that incorporate soft constraints into the end-to-end learning process; (2) among the latter, methods that decompose the parameters for the more specific classes vs. regularization methods; (3) among the regularization methods, we compare those which target the representation of the class vs. of the encoded instance. We now describe the application of these methods and assess their characteristics.

[1] In earlier work (Dayanik et al., 2021), we experimented with other state-of-the-art architectures, including BiLSTMs with and without attention, but obtained worse performance.

[2] The appendix gives details on the BERT models we use.

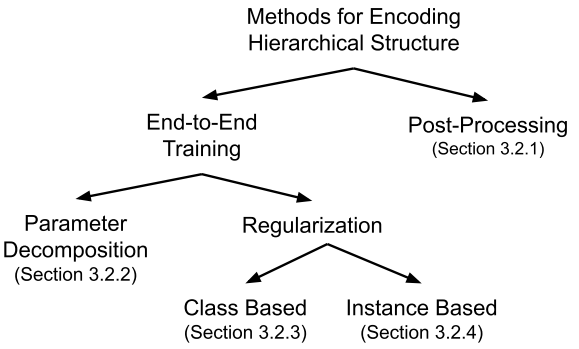

Figure 1: Encoding hierarchical information

3.2.1 Post-processing: ILP

Integer Linear Programming (ILP) is a sub-type of Linear Programming, a family of constrained optimization problems over linear objective functions. ILP introduces the additional constraint that variables can take only integer values. ILP models have been used in NLP tasks such as dependency parsing (Riedel and Clarke, 2006) or semantic role labeling (Punyakanok et al., 2004) to enforce linguistically motivated constraints on predicted structures.

In our application, where a classifier might predict a subcategory with a mismatching supercategory, ILP can select the most likely legal output from the classifier probabilities so that (1) for each predicted subcategory, the matching supercategory is predicted, and (2) for each predicted supercategory, at least one matching subcategory is predicted. For each category we introduce a binary variable v_i indicating if the category is predicted. The objective function is the log likelihood of the model output (including predicted and non-predicted classes), using the estimates of the neural classifiers P_{NC}:

$$\phi_i = P_{\text{NC}}(v_i = 1) \quad (1)$$

$$\mathcal{L} = \sum_i \log \phi_i v_i + \log[1 - \phi_i](1 - v_i) \quad (2)$$

Let sup(i) denote the supercategory for the subcategory i. Then we formalize constraint (1) as:

$$\text{for each subcat. } v_i : v_i - v_{sup(i)} \leq 0 \quad (3)$$

Correspondingly, let subs(i) denote the set of subcategories for supercategory i. Then the second constraint from above is formalized as:

$$\text{for each supercat. } v_i : v_i - \sum_{j \in subs(i)} v_j \leq 0 \quad (4)$$

Assessment: In contrast to the other methods introduced in this Section, ILP imposes hard constraints

on the output. It does not introduce additional parameters. It is only applicable to multi-label classification. As a post processing step, it does not propagate the errors back into the representations.

3.2.2 Parameter Decomposition: HLE

Hierarchical Label Encoding (HLE), introduced by Shimaoka et al. (2017) for fine-grained named entity recognition, decomposes the representation of each subcategory into a sum of vectors, one for the subcategory itself and one for each of its supercategories. Formally, it creates a binary square matrix, $B \in \{0,1\}^{l \times l}$, where l is the total number of sub- and supercategories. Each cell in the matrix is filled with 1 either if the column class is a subclass of or the same as the row class, and filled with 0 otherwise. The matrix B is not updated during training and integrated into models by multiplying it by the weight matrix W_c of the classifier:

$$W_c^{'} = (W_c^{\top} B) \tag{5}$$

where $W_c \in \mathbb{R}^{l \times hs}$, hs is the size of the hidden state of the encoder and W_c' is the modified parameters of the classifier.

Assessment: HLE imposes soft constraints and does not introduce any parameters. Similar to ILP, HLE can only be used in multi-label classification.

3.2.3 Class Representation Regularization

Class representation regularization (CRR) falls under the umbrella of regularization methods which have been used to encode prior knowledge for different NLP tasks (Eisenstein et al., 2011; Sattigeri and J. Thiagarajan, 2016) and has been shown to improve classification performance on a diverse set of hierarchical datasets under both supervised (Naik and Rangwala, 2015) and semi-supervised learning scenarios (Bui et al., 2018; Stretcu et al., 2019). In our case, the goal is to increase the similarity between the weight vectors of the subcategories belonging to the *same* supercategory while keeping the weight vectors of subcategories *across* supercategories dissimilar.

Formally, the classification layer (cf. Section 3.1) is a weight matrix $W_c \in \mathbb{R}^{l \times hs}$, where l is the number of classes and hs is the output size of the encoder. We use S for the set of supercategories and S_i to denote the i-th supercategory, the set of its subcategories, and their weight vectors, depending on context. Then we define the centroid $\mu(S_i)$ of a supercategory, the average distance between two supercategories, d_{avg}, and the global intra- and inter-supercategory distances $d^{\text{inter}}/d^{\text{intra}}$ as:

$$\mu(S_i) = \frac{1}{|S_i|} \sum_{w \in S_i} w \tag{6}$$

$$d_{\text{avg}}(S_i, S_j) = \frac{1}{|S_i||S_j|} \sum_{\substack{w \in S_i, \\ w' \in S_j}} \text{dist}(w, w') \tag{7}$$

$$d^{\text{inter}} = \sum_{0 \le i < j \le |S|} d_{\text{avg}}(S_i, S_j) \tag{8}$$

$$d^{\text{intra}} = \sum_{i=1}^{|S|} \frac{1}{|S_i|} \sum_{w \in S_i} dist(\mu(S_i), w) \tag{9}$$

Finally, we regularize the learning objective ($\mathcal{L}_{\text{main}}$, cf. Section 3.1) as follows:

$$\mathcal{L} = \mathcal{L}_{\text{main}} + \alpha d^{\text{intra}} - \beta d^{\text{inter}} \tag{10}$$

where the hyperparameters $\alpha, \beta \ge 0$ control regularization strength.

Assessment: CRR imposes soft constraints, adds two hyper parameters, and is applicable to both single and multi label classification.

3.2.4 Instance Representation Regularization

Instance representation regularization (IRR) applies the same intuition as above, but at the level of the instance representations produced $e(x)$ by the encoder. The model is penalized whenever the encoder generates more similar representations for input pairs with different supercategories than for pairs with the same supercategories. A similar approach was proposed by Choi and Rhee (2019) for non-hierarchical classification to simply keep class representations distinct from one another.

Formally, let X be the set of instances, and $s(x)$ be the supercategory of instance x. We consider the set of instance triplets where the first and second member share a supercategory and the third has a separate one, and measure the extent to which the distance across supercategories exceeds the distance within the supercategory:

$$d^{\text{diff}} = \sum_{\substack{x,y,z \in X \\ s(x)=s(y) \\ s(x) \neq s(z)}} \max(0, \text{dist}(e(x), e(y)) - \text{dist}(e(y), e(z))) \tag{11}$$

We then regularize the learning objective as:

$$\mathcal{L} = \mathcal{L}_{\text{main}} + \alpha \cdot d^{\text{diff}} \tag{12}$$

ID	Label	f	#sub	mean f.sub
1xx	Controlling Migration	998	16	62 ± 46.2
2xx	Residency	726	18	40 ± 41.2
3xx	Integration	475	15	31 ± 35.5
4xx	Domestic Security	230	9	25 ± 17.9
5xx	Foreign Policy	689	9	76 ± 17.8
6xx	Economy	194	12	16 ± 13.1
7xx	Society	749	19	39 ± 37.9
8xx	Procedures	676	20	33 ± 37.7
	Overall	4737	118	

Table 1: Subcategory distribution by supercategories in DebateNet dataset: *ID*; *Label*; frequency (*f*); number of subcategories (*#sub*); mean subcategory frequency with standard deviation (*mean f.sub*).

where $\alpha \geq 0$ controls the regularization strength. Since using the complete set of triples is computationally demanding, it may be necessary to sample instead. In this paper, we create triples from each mini-batch by combining its instances, which is an approximation to uniform sampling (cf. Sections 4.2 and 5.2).

Assessment: IRR also imposes soft constraints, adding one hyperparameter. IRR requires each instance to belong to a single supercategory.

4 Experiment 1: Newspapers

4.1 Dataset

Our first experiment adopts a monolingual multi-label statement classification task. We work with an extended version of *DebateNet-mig15* (Lapesa et al., 2020), a German corpus of migration-related *claims*, statements targeting a specific action to be taken in a policy field.[3] The corpus comprises 1361 articles from the 2015 issues of the German quality newspaper *taz*. The corpus, referred to in what follows as *DebateNet*,is annotated manually according to a two-level ontology (Table 1) for the migration domain, comprising 8 supercategories with 118 subcategories. There is a total of 3827 annotated textual spans that can be assigned subcategories if the statements touch on several policy issues. For example, the following sentence:

Eine weitere massive Verfahrensbeschleunigung ist bei vorübergehenden Grenzkontrollen vor der Einreise vorgesehen

(A further massive acceleration of procedures is envisaged for temporary border controls prior to entry)

is assigned to the subcategories Border Controls (supercategory Controlling Migration) as well as Accelerated Procedure (supercategory Procedures).

4.2 Experimental Setup

Given these properties, we model statement classification on DebateNet as multi-label classification. Furthermore, we remove 46 extremely infrequent subcategories with less than 20 instances each. For each supercategory, we merge these infrequent subcategorie into the pre-existing 'catch-all' subcategory x99. We acknowledge that that makes the catch-all subcategories are presumably challenging to learn, given their inhomogeneous nature, but we believe that this strategy is reasonable, since no instances are discarded in this manner, and they still retain the supercategory signal that we are interested in. This results in a final count of 72 subcategories.

We experiment with eight model variations: **Base**; **ILP**, **HLE** and **CRR**; and the combinations **HLE+ILP**, **HLE+CRR**,**CRR+ILP** and **HLE+CRR+ILP**. Recall that IRR is not applicable to multi-label classification. We use Euclidean distance as dist in CRR.

We adopt the 90/10 train/test split of Dayanik et al. (2021) and perform grid search by cross-validation on the training set to optimize hyperparameters, including mini-batch size. We report weighted-averaged Precision, Recall and F1 scores on the whole dataset and three equal-sized frequency bands of categories. Details on the bands and the training method are given in Appendix A.

4.3 Results

Does hierarchical information improve overall performance? Table 2 summarizes the results, with the Overall results in the first row. The Base model achieves the lowest overall F1 score among the others (47 points), indicating the general efficacy of integrating hierarchical information into the classifier. However, different extensions of the base model show different effects in terms of Precision vs. Recall: ILP (2nd column) improves Recall only (+8) while both Precision and Recall benefit from HLE (+14/+10) and CRR (+9/+7). The combination HLE+ILP yields the best Recall (+17), and the combination of HLE and CRR is the best overall model (F1=61: +14 F1, +15 Pr, +14 R). We slightly outperform the results of the best model from our previous study (Dayanik et al., 2021), namely, HLE-only, by 1% overall F1 and on two of

[3]The corpus is available at `mardy-spp.github.io`.

Freq band	Base			ILP			HLE			CRR			HLE+ILP			HLE+CRR			CRR+ILP			HLE+CRR+ILP		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Overall	61.2	41.9	47.0	56.0	49.7	50.4	75.2	52.2	59.0	70.4	49.0	55.2	65.8	59.0	60.5	76.5	54.3	**60.8**	66.0	55.4	57.8	64.3	57.3	58.6
Low	10.2	9.7	9.6	18.3	14.5	14.8	58.3	30.6	37.4	31.2	16.1	18.7	48.1	30.6	34.8	54.8	29.0	35.8	35.5	19.4	21.9	52.2	33.9	**38.3**
Mid	58.0	36.0	41.8	65.0	47.4	50.4	77.4	55.3	62.2	75.8	49.1	55.8	71.5	63.2	65.1	85.1	58.8	**66.2**	74.3	58.8	61.5	71.9	62.3	64.0
High	73.1	50.8	56.7	60.5	57.9	57.9	77.8	55.6	62.3	76.4	55.9	62.6	67.3	63.3	**64.0**	77.7	57.9	**64.0**	69.1	61.6	63.8	63.9	60.3	60.8

Table 2: Experiment 1 (multi-label statement classification): Precision, Recall, F-Scores for the DebateNet Dataset (Overall and broken down by category frequency bands).

Supercategory	Fi			De			Hu			Tr			En		
	f	#sub	mean f.sub	f	#sub	mean f.sub	f	#sub	mean f.sub	f	#sub	mean f.sub	f	#sub	mean f.sub
External Relations	1599	10	159 ± 159	5727	10	572 ± 665	2288	9	254 ± 268	3721	10	372 ± 435	3071	10	307 ± 302
Freedom, Democracy	758	4	189 ± 209	5672	4	1418 ± 1547	3553	4	888 ± 705	5211	4	1302 ± 1443	2091	4	522 ± 509
Political System	1129	5	225 ± 226	5661	5	1132 ± 1012	4040	5	808 ± 423	3299	5	659 ± 405	2530	5	506 ± 553
Economy	4556	15	303 ± 395	15185	16	949 ± 1082	10380	16	648 ± 773	17899	16	1118 ± 1557	6753	15	450 ± 499
Welfare, Quality of Life	7787	7	1112 ± 927	16592	7	2370 ± 1965	15121	7	2160 ± 1567	11120	7	1588 ± 1414	10246	7	1463 ± 1431
Fabric of Society	2677	8	334 ± 203	6095	8	761 ± 452	5500	8	687 ± 582	5555	8	694 ± 721	3328	8	416 ± 448
Social Groups	2113	6	352 ± 523	5865	6	977 ± 1102	3625	6	604 ± 635	5157	5	1031 ± 988	2075	6	345 ± 422
Overall	20619			60797			44507			51962			30094		

Table 3: Subcategory distribution by supercategories in the complete (100%) Manifesto dataset: frequency (*f*); number of subcategories (*#sub*); mean subcategory frequency with SD (*mean f.sub*). Total: instances per language.

the three frequency bands (low, mid +1% F1), with a tie on the third one (high), which we attribute to the addition of class level regularization through the CRR component. We obtain the best results for $\alpha \in [0.005, 0.01]$ and $\beta = 0.01$: thus, a very mild regularization already has a substantial effect.

How do hierarchical structure and category frequency interact? The results by frequency band enable us to analyze classification performance depending on frequency. We observe that the Base model fails badly in the low frequency band (F1=10) while doing a fair job in the mid-frequency and high-frequency bands (F1=42 and 57). The inclusion of hierarchical information leads to the most substantial improvements for the low-frequency band (+28 F1 for HLE+CRR+ILP). Improvements are generally correlated with (in)frequency: the best overall model, HLE+CRR, improves the mid-frequency band by 20 points F1 and the high-frequency band by 7 points F1. Figure 2 shows the subcategories with the highest improvement: four belong to the mid-frequency and three to the low-frequency band.

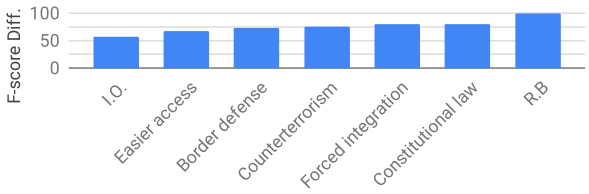

Figure 2: Experiment 1: Seven subcategories with highest F1 increase for best model compared to base model. I.O:Integration Offers, R.B:Reducing Bureaucracy

5 Experiment 2: Party Manifestos

5.1 Dataset

Our second (single-label classification) experiment targets political statements in party manifestos, official documents issued by parties to summarize their political program. We build on the Comparative Manifesto Project (Volkens et al., 2019) which collected and manually coded manifestos from multiple countries and languages. Considering the availability of language specific transformer based models and large annotated data, we focus on 5 countries with one language each: Finland (Fi), Germany (De), Hungary (Hu), Turkey (Tr) and United Kingdom (En).Note that this is not a parallel corpus, and the amount of annotated data available for each language varies greatly (cf. Table 3). Coding uses a two-level ontology of 7 policy areas as supercategories "designed to be comparable between parties, countries, elections, and across time", and 56 subcategories (Table 3).[4] Sentences are split into segments if they discuss unrelated topics or different aspects of a larger policy, so each segment is assigned a single subcategory.

[4] https://manifesto-project.wzb.eu/coding_schemes/mp_v5

Lang	Plain			CRR			IRR			CRR + IRR		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Fi	39.0	38.4	37.4	40.6	40.0	39.3	41.5	39.2	38.6	42.2	40.8	**40.1**
De	33.3	31.3	31.4	35.4	34.1	34.2	34.6	34.7	34.3	36.8	34.8	**34.9**
Hu	41.1	38.8	38.7	41.7	39.8	39.7	42.2	39.0	39.2	43.7	39.3	**39.8**
Tr	45.6	42.5	42.4	47.9	41.7	43.0	48.9	42.4	43.3	49.0	42.5	**43.6**
En	31.5	30.8	30.5	34.6	32.5	32.3	32.7	32.7	32.1	34.4	32.5	**32.8**

Table 4: Experiment 2 (single-label statement classification): Macro-averaged Precision, Recall, F1 scores for the Manifesto dataset (Overall, trained on 25% of the data).

Lang	Freq band	Base			CRR			IRR			CRR + IRR		
		P	R	F1	P	R	F1	P	R	F1	P	R	F1
Fi	Low	18.4	15.2	13.7	20.7	17.7	16.7	22.6	16.6	15.4	25.5	19.6	**19.5**
	Mid	42.1	42.2	41.5	42.5	42.6	41.9	44.4	42.7	42.7	43.9	43.9	**43.0**
	High	56.6	57.8	57.0	58.7	59.8	**59.2**	57.4	58.4	57.7	57.3	58.9	57.9
De	Low	16.1	9.0	10.6	19.7	14.7	16.4	18.6	17.7	17.8	23.1	16.2	**18.0**
	Mid	36.9	38.3	37.4	38.3	40.3	38.7	37.3	40.8	38.5	38.7	40.5	**38.9**
	High	48.7	48.9	48.5	49.9	49.4	49.3	49.6	47.6	48.4	50.1	49.7	**49.7**
Hu	Low	24.5	15.4	17.3	26.4	18.4	19.9	28.4	16.9	19.1	33.6	17.5	**21.1**
	Mid	41.5	43.7	41.7	41.5	43.8	**42.1**	41.0	43.5	41.6	40.1	42.7	40.9
	High	57.3	57.2	57.0	57.2	57.2	57.0	57.3	56.7	56.7	57.3	57.7	**57.4**
Tr	Low	29.2	19.6	20.2	37.4	20.8	24.2	40.4	22.2	**24.9**	38.0	21.0	23.8
	Mid	46.4	47.3	**46.6**	45.8	43.2	44.1	46.0	44.1	44.8	48.8	44.9	46.4
	High	61.1	60.6	**60.7**	60.4	61.0	60.6	60.1	60.8	60.1	60.3	61.5	**60.7**
En	Low	13.3	8.3	9.7	20.1	10.8	12.9	14.6	10.7	11.9	17.2	11.3	**13.3**
	Mid	30.5	31.7	30.6	32.1	34.7	32.5	32.0	34.9	32.8	33.7	33.1	**32.9**
	High	50.7	52.4	51.3	51.7	52.0	51.6	51.6	52.3	51.6	52.3	53.2	**52.2**

Table 5: Experiment 2 (single-label statement classification): Macro-averaged Precision, Recall, F1 scores for the Manifesto dataset (by category frequency band, trained on 25% of the data).

5.2 Experimental Setup

We model statement classification in the Manifesto corpus at the segment level as a single-label classification task. Unlike in Section 4.1, we do not apply any pre-processing to merge very infrequent subcategories, since all categories in the Manifesto corpus are frequent enough. For example, there is only one subcategory with less instances than the threshold (20) in the DE portion.

Since HLE and ILP are only useful for multi-label classification, we experiment with the following model variations: **Base**; **CRR**, **IRR**; and **CRR+IRR**. As distance metric, we use L_1 distance in CRR and Cosine distance in IRR. (Other choices led to worse results.)

We split the dataset into train (65%), validation (15%), and test (20%) portions. With several hundred thousand sentences after years of annotation, the Manifesto corpus is one of the largest CSS datasets available and its size is arguably larger than typical for CSS projects (annotation of the 4k DebateNet instances took more than a year). For this reason, we introduce a further experimental variable, namely the amount of the training data. This allows us to simulate the application of these methods to scenarios in which smaller amounts of training are available. Specifically, we use random draws of percentages (25%, 50% and 100%) of the full training set, keeping the test set constant. Due to space constraints, we will discuss only the 25% case in detail and provide an overview of the 50% and 100% cases, whose details can be found in the appendix. We perform hyperparameter search for each language separately and adopt the same

evaluation setup as in Experiment 1 (Section 4.2).

5.3 Results

Does hierarchical information improve performance? Table 4 shows the results for 25% training data of each language. The results are surprisingly similar across all languages, despite the typological differences and varying amounts of training data. The Base model consistently yields the worst results, in line with the findings of Experiment 1.

The use of hierarchical structure, both through CRR and IRR, leads to improvements for all languages, with no clear winner between the two. However, as was the case in Experiment 1 for CRR+HLE, the two methods can be beneficially combined: CRR+IRR yields the highest F-Score for each language: the gains over Base are between 1.1 points (Hu) and 2.3 points (En). The improvements are substantially smaller than in Experiment 1, which we attribute to the larger amount of data available, both overall and per subcategory. We obtained the best results for $\alpha = 0.1$ and $\beta \in [0.1, 0.2]$ indicating that the CMP data profits from a bit more but still mild regularization. Our setup is not exactly comparable to previous work, but our 100% condition (cf. Appendix A) matches or exceeds the results of the closest study by Subramanian et al. (2018).

How do hierarchical structure and category frequency interact? As in Experiment 1, we analyze the impact of hierarchical structure on three equal-sized subcategory frequency bands, shown in Table 5, for the 25% condition. Similar to Experiment 1, the Plain model fails badly on the low frequency band with F1 between 9.7 (En) and 20.2 (Tr). The combination CRR+IRR yields the highest improvements for this frequency band, between 3 and 7 points F1. (Turkish is an exception with the highest F1 for IRR without CRR.) CRR and IRR also generally improve the results for the two other bands, but (again in line with Experiment 1) the gains are more modest, up to 2.5% F1 for the mid-frequency and 1.0% F1 for the high-frequency abdn. Indeed, a correlation analysis shows a significant negative correlation between subcategory size and the F1 improvement of CRR+IRR over Base, $r = -0.19$. In the higher frequency bands, the variance is also higher, with some wins for CRR (Fi, Hu), IRR (Tr), or the Base model (Tr).

Corpus size and hierarchical structure. As stated above, our main results use the 25% condition. To assess the behavior for larger datasets, Figure 3 summarizes the mean improvement in F1 between Base and IRR+CRR for the 25%, 50% and 100% conditions. The improvement is largest for the 25% setting, further supporting our observations that incorporating hierarchical information into the models is especially important in a low data regime. That being said, we still obtain consistent improvements for the 50% condition. For 100%, we still gain 1-2 points F1 for De, En, and Fi. In contrast, Tr and Hu lose slightly on the full dataset (100%). Further analysis (Appendix B.3) shows that in Tr and Hu, the high-frequency band – where we see the least improvement – account for 76% and 79% of the data, respectively, while it only makes up, e.g., 73% of the German data.

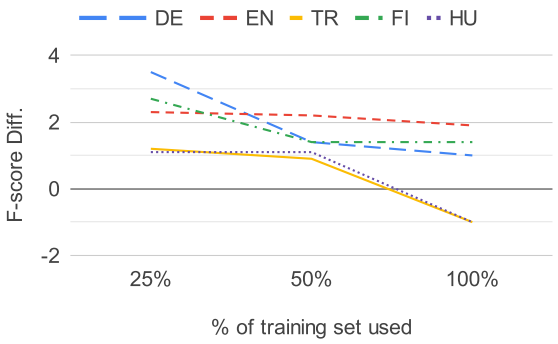

Figure 3: Experiment 2: F1 difference between the CRR+IRR and Base models across training data sizes.

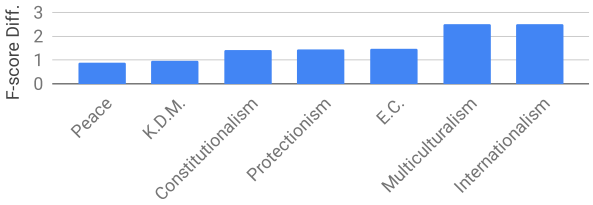

Figure 4: Experiment 2: Seven subcategories with highest F1 increase for best model compared to base model. Peace, E.C and Protectionism belong to mid frequency class. The other four subcategory belong to low band. K.D.M: Keynesian Demand Management, E.C: European Community/Union.

Qualitative Analysis. Table 6 shows some English examples which were classified incorrectly by the Base model and correctly by the IRR+CRR model. All involve arguably related subcategories, illustrating the benefit of hierarchical modeling to counteract the substitution, among related categories, of the more frequent by the less frequent one. This pattern is bolstered by Figure 4, which

Input	Base Pred. (incorrect)	CRR+IRR Pred. (correct)
Our long-term economic plan is turning around Britain's economy.	Economic growth (Mid)	Economic planning (Low)
Face coverings such as these are barriers to integration.	National way of life (Mid)	Multiculturalism (Low)
Fairer corporate governance, built on new rules for takeovers executive pay and worker representation on company boards.	Market regulation (High)	Corporatism (Low)
This sent out terrible signals: if you did the right thing, you were penalised — and if you did the wrong thing, you were rewarded, with the unfairness of it all infuriating hardworking people.	Equality (High)	Welfare limitation (Low)

Table 6: Examples from Manifesto dataset correctly classified only by CRR+IRR. Mid, Low, High indicates frequency band of predicted subcategories.

shows the 7 subcategories with the largest improvement in F1: Three of them belong to the mid-frequency band, four to the low-frequency band, and none to the high-frequency band.

6 Conclusion

This paper addresses the task of political statement classification focussing on the challenge of class imbalance. We have argued that the hierarchically structured codebooks developed by political science projects are a source of domain knowledge that can be integrated in classification models. We extend state-of-the-art transformer models with lightweight modules that implement this intuition in different ways. We evaluate on two datasets, covering two codebooks, single-label and multi-label classification, and various languages. Our main findings are robust across the different setups: inclusion of hierarchical information virtually always improves classification, and the methods we consider are sufficiently complementary that their benefits combine. We obtain improvements even for fairly large datasets, with diminishing benefits for very large datasets – which is plausible, given that performance improves particularly for low-frequency categories.

The latter finding – strong improvements for low-frequency categories – is arguably important with regard to algorithmic fairness (Dayanik and Padó, 2020; Jacobs and Wallach, 2021), since in the case of rare categories, a small number of prediction errors is sufficient to substantially impact the reliability of downstream analyses. Indeed, multiple causes of low frequency categories exist. As one example, in analyses over time, statement frequencies co-vary naturally with topic prominence, and analyses like the (semi-)automatic extraction of network representations to assess dynamics of political debates (Haunss et al., 2020) may misrepresent the contribution of infrequent categories. As another example, work on the framing of immigration discourse on Twitter (Mendelsohn et al., 2021) has shown that employing issue-specific categories (e.g., "victim:war", "victim: discrimination", "threat:jobs", "threat:public order") reveal ideological and regional patterns which would be missed by the commonly employed generic frames such "economy" or "morality" (Card et al., 2015) – but at the cost of introducing many fine-grained categories which are sparse and attested with widely different frequencies. Our work demonstrates that a well designed hierarchical codebook, combined with the right computational devices, can go a long way towards redressing the challenges that arise from this situation. An more detailed assessment of the impact of our methods on downstream tasks remains future work.

Acknowledgments

We acknowledge funding by Deutsche Forschungsgemeinschaft (DFG) for project MARDY 2 (375875969) within SPP RATIO and by Bundesministerium für Bildung und Forschung (BMBF) through E-DELIB (Powering up e-deliberation: towards AI-supported moderation).

Ethics Statement

The research reported in this paper is concerned with fundamental aspects of machine learning models by enabling machine learning models to better represent data and improve the representation of low-frequency categories, ideally improving fairness. The methods we propose for this purpose do not introduce additional ethical risks on top of the previous work we build upon.

References

Rami Aly, Steffen Remus, and Chris Biemann. 2019. Hierarchical multi-label classification of text with capsule networks. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 323–330, Florence, Italy. Association for Computational Linguistics.

André Blessing, Nico Blokker, Erenay Dayanik, Sebastian Haunss, Jonas Kuhn, Gabriella Lapesa, and Sebastian Padó. 2019. *Modeling ARgumentation DYnamics in political discourse. Codebook. Topic: immigration in Germany (2015).*

Thang D Bui, Sujith Ravi, and Vivek Ramavajjala. 2018. Neural graph learning: Training neural networks using graphs. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, pages 64–71.

Dallas Card, Amber E. Boydstun, Justin H. Gross, Philip Resnik, and Noah A. Smith. 2015. The media frames corpus: Annotations of frames across issues. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 2: Short Papers*, pages 438–444. The Association for Computer Linguistics.

Ilias Chalkidis, Emmanouil Fergadiotis, Prodromos Malakasiotis, and Ion Androutsopoulos. 2019. Large-scale multi-label text classification on EU legislation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6314–6322, Florence, Italy. Association for Computational Linguistics.

Daeyoung Choi and Wonjong Rhee. 2019. Utilizing class information for deep network representation shaping. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):3396–3403.

Erenay Dayanik, Andre Blessing, Nico Blokker, Sebastian Haunss, Jonas Kuhn, Gabriella Lapesa, and Sebastian Padó. 2021. Using hierarchical class structure to improve fine-grained claim classification. In *Proceedings of the 5th Workshop on Structured Prediction for NLP (SPNLP 2021)*, pages 53–60, Online. Association for Computational Linguistics.

Erenay Dayanik and Sebastian Padó. 2020. Masking actor information leads to fairer political claims detection. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4385–4391, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Jacob Eisenstein, Noah A. Smith, and Eric P. Xing. 2011. Discovering sociolinguistic associations with structured sparsity. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1365–1374, Portland, Oregon, USA. Association for Computational Linguistics.

Frank Fischer and John Forester, editors. 1993. *The Argumentative turn in policy analysis and planning*. Duke University Press.

Goran Glavaš, Federico Nanni, and Simone Paolo Ponzetto. 2017a. Cross-lingual classification of topics in political texts. In *Proceedings of the Second Workshop on NLP and Computational Social Science*, pages 42–46, Vancouver, Canada. Association for Computational Linguistics.

Goran Glavaš, Federico Nanni, and Simone Paolo Ponzetto. 2017b. Unsupervised cross-lingual scaling of political texts. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 688–693, Valencia, Spain. Association for Computational Linguistics.

Goran Glavaš and Ivan Vulić. 2019. Generalized tuning of distributional word vectors for monolingual and cross-lingual lexical entailment. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4824–4830, Florence, Italy. Association for Computational Linguistics.

Sebastian Haunss, Jonas Kuhn, Sebastian Pado, Andre Blessing, Nico Blokker, Erenay Dayanik, and Gabriella Lapesa. 2020. Integrating manual and automatic annotation for the creation of discourse network data sets. *Politics and Governance*, 8(2).

Ivana Clairine Irsan and Masayu Leylia Khodra. 2019. Hierarchical multi-label news article classification

with distributed semantic model based features. *International Journal of Advances in Intelligent Informatics*, 5(1):40–47.

Abigail Z. Jacobs and Hanna Wallach. 2021. Measurement and fairness. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, page 375–385, New York, NY, USA. Association for Computing Machinery.

Mladen Karan, Jan Šnajder, Daniela Širinić, and Goran Glavaš. 2016. Analysis of policy agendas: Lessons learned from automatic topic classification of Croatian political texts. In *Proceedings of the 10th SIGHUM Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*, pages 12–21, Berlin, Germany. Association for Computational Linguistics.

Stefan Kennedy, Niall Walsh, Kirils Sloka, Andrew McCarren, and Jennifer Foster. 2019. Fact or factitious? contextualized opinion spam detection. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 344–350, Florence, Italy. Association for Computational Linguistics.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar. Association for Computational Linguistics.

Ruud Koopmans. 2002. *Codebook for the Analysis of Political Mobilisation and Communication in European Public Spheres*.

Gabriella Lapesa, Andre Blessing, Nico Blokker, Erenay Dayanik, Sebastian Haunss, Jonas Kuhn, and Sebastian Padó. 2020. DEbateNet-mig15:tracing the 2015 immigration debate in Germany over time. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 919–927, Marseille, France. European Language Resources Association.

Yingjie Li and Cornelia Caragea. 2019. Multi-task stance detection with sentiment and stance lexicons. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6299–6305, Hong Kong, China. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2020. RoBERTa: A robustly optimized BERT pretraining approach. *ArXiv*, abs/1907.11692.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization.

Philipp Mayring. 2019. Qualitative content analysis: Demarcation, varieties, developments. *Forum Qualitative Sozialforschung / Forum: Qualitative Social Research*, 20(3).

Julia Mendelsohn, Ceren Budak, and David Jurgens. 2021. Modeling framing in immigration discourse on social media. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2219–2263, Online. Association for Computational Linguistics.

Nicolas Merz, Sven Regel, and Jirka Lewandowski. 2016. The manifesto corpus: A new resource for research on political parties and quantitative text analysis. *Research & Politics*, 3(2):2053168016643346.

Shervin Minaee, Nal Kalchbrenner, Erik Cambria, Narjes Nikzad, Meysam Chenaghlu, and Jianfeng Gao. 2021. Deep learning–based text classification: A comprehensive review. *ACM Comput. Surv.*, 54(3).

Azad Naik and Huzefa Rangwala. 2015. A ranking-based approach for hierarchical classification. In *2015 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, pages 1–10.

Federico Nanni, Simone Paolo Ponzetto, and Laura Dietz. 2017. Building entity-centric event collections. In *2017 ACM/IEEE Joint Conference on Digital Libraries (JCDL)*, pages 1–10.

Frank Nullmeier, Roland Lhotta, Dominika Biegoń, Jennifer Gronau, Sebastian Haunss, Achim Hurrelmann, Zuzana Krell-Laluhová, Falk Lenke, Martin Nonhoff, Tanja Pritzlaff, and et al. 2015. *Project B1: Legitimating States, International Regimes, and Economic Orders. Codebook – Final Version*.

Sebastian Padó, Andre Blessing, Nico Blokker, Erenay Dayanik, Sebastian Haunss, and Jonas Kuhn. 2019. Who sides with whom? towards computational construction of discourse networks for political debates. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2841–2847, Florence, Italy. Association for Computational Linguistics.

Vasin Punyakanok, Dan Roth, Wen-tau Yih, and Dav Zimak. 2004. Semantic role labeling via integer linear programming inference. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, pages 1346–1352, Geneva, Switzerland. COLING.

Sebastian Riedel and James Clarke. 2006. Incremental integer linear programming for non-projective dependency parsing. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 129–137, Sydney, Australia. Association for Computational Linguistics.

Anthony Rios and Ramakanth Kavuluru. 2018. Few-shot and zero-shot multi-label learning for structured label spaces. In *Proceedings of the 2018 Conference*

on Empirical Methods in Natural Language Processing, pages 3132–3142, Brussels, Belgium. Association for Computational Linguistics.

Prasanna Sattigeri and Jayaraman J. Thiagarajan. 2016. Sparsifying word representations for deep unordered sentence modeling. In *Proceedings of the 1st Workshop on Representation Learning for NLP*, pages 206–214, Berlin, Germany. Association for Computational Linguistics.

Sonse Shimaoka, Pontus Stenetorp, Kentaro Inui, and Sebastian Riedel. 2017. Neural architectures for fine-grained entity type classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1271–1280, Valencia, Spain. Association for Computational Linguistics.

Yangqiu Song and Dan Roth. 2014. On dataless hierarchical text classification. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence*, AAAI'14, page 1579–1585. AAAI Press.

Otilia Stretcu, Krishnamurthy Viswanathan, Dana Movshovitz-Attias, Emmanouil Platanios, Sujith Ravi, and Andrew Tomkins. 2019. Graph agreement models for semi-supervised learning. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Shivashankar Subramanian, Trevor Cohn, and Timothy Baldwin. 2018. Hierarchical structured model for fine-to-coarse manifesto text analysis. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1964–1974, New Orleans, Louisiana. Association for Computational Linguistics.

Suzan Verberne, Eva D'Hondt, Antal van den Bosch, and Maarten Marx. 2014. Automatic thematic classification of election manifestos. *Information Processing & Management*, 50(4):554–567.

Andrea Volkens, Tobias Burst, Werner Krause, Pola Lehmann, Theres Matthieß, Nicolas Merz, Sven Regel, Bernhard Weßels, and Lisa Zehnter. 2020. *The Manifesto Project Dataset - Codebook*. Manifesto Project (MRG/CMP/MARPOR). Version 2020b.

Andrea Volkens, Werner Krause, Pola Lehmann, Theres Matthieß, Nicolas Merz, Sven Regel, and Bernhard Weßels. 2019. The Manifesto data collection, version 2019b.

Hendrik Wagenaar. 2011. *Meaning in Action. Interpretation and Dialogue in Policy Analysis*. M.E. Sharpe.

Annika Werner, Onawa Lacewell, and Andrea Volkens. 2011. *Manifesto coding instructions*, 4th fully revised edition edition. Available at: https://manifestoproject.wzb.eu/down/papers/handbook_2011_version_4.pdf.

Lin Xiao, Xin Huang, Boli Chen, and Liping Jing. 2019. Label-specific document representation for multi-label text classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 466–475, Hong Kong, China. Association for Computational Linguistics.

Jie Zhou, Chunping Ma, Dingkun Long, Guangwei Xu, Ning Ding, Haoyu Zhang, Pengjun Xie, and Gongshen Liu. 2020. Hierarchy-aware global model for hierarchical text classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1106–1117, Online. Association for Computational Linguistics.

Cäcilia Zirn, Goran Glavaš, Federico Nanni, Jason Eichorts, and Heiner Stuckenschmidt. 2016. Classifying topics and detecting topic shifts in political manifestos. In *Proceedings of PolText 2016: The International Conference on the Advances in Computational Analysis of Political Text*, pages 88–93, Zagreb. University of Zagreb.

A Details on Experiment 1

A.1 Dataset Details

We split the fine-grained categories into three equal-sized frequency bands using following threshold values: high-frequency (265≥f≥ 67), mid-frequency (65≥f≥ 40) and low-frequency (20≥f≥ 39). Table 7 shows the category frequency band assignments in the DebateNet dataset.

Band	Label					
Low-frequency	111	199	201	209	213	214
	406	408	499	502	505	508
	602	603	605	701	706	707
	708	801	802	807	811	814
Mid-frequency	106	107	109	204	211	212
	215	301	302	303	307	401
	402	405	503	509	601	699
	702	711	715	803	804	808
High-frequency	101	102	104	105	108	110
	190	202	203	207	299	309
	399	501	504	507	703	705
	709	712	799	805	812	899

Table 7: Lists of the categories in the frequency bands

A.2 Training Details

We use use a cased BERT variant that was trained specifically for the target language. We split DebateNet into to a train set (90%) and a test set

(10%) and perform grid search by cross validation on the training set to optimize hyperparameters. All models are trained using cross entropy loss with the sigmoid activation function and AdamW (Loshchilov and Hutter, 2019) optimizer. We perform grid search for hyperparameter optimization and use the hyperparameters leading highest average F1 score during 5-Fold cross validation. Following lower and upper bounds have been applied during search for each hyperparameter: learning rate: [1e-5, 5e-2], epoch: [5, 25], mini-batch size: [16, 32], dropout: [0.2,0.8], α: [0.005,0.6], β: [0.01,0.6]. The best hyperparameters for the best model (HLE+IRR+ILP) are shown in Table 8.

Lang	Train	lr	α_{CRR}	β	α_{IRR}	dp
DebateNet		5e-5	0.01	0.01	-	0.3
Fi	25%	3e-5	0.1	0.1	0.1	0.4
	50%	2e-5	0.05	0.05	0.1	0.2
	100%	2e-5	0.05	0.05	0.1	0.2
De	25%	2e-5	0.2	0.2	0.4	0.2
	50%	2e-5	0.05	0.01	0.2	0.2
	100%	2e-5	0.1	0.2	0.1	0.1
Hu	25%	2e-5	0.4	0.05	0.1	0.2
	50%	2e-5	0.1	0.1	0.1	0.2
	100%	2e-5	0.01	0.01	0.05	0.2
Tr	25%	2e-5	0.2	0.2	0.4	0.2
	50%	2e-5	0.2	0.4	0.05	0.2
	100%	2e-5	0.01	0.01	0.1	0.2
En	25%	3e-5	0.05	-0.05	0.1	0.4
	50%	3e-5	0.2	0.2	0.4	0.4
	100%	3e-5	0.05	0.05	0.4	0.4

Table 8: Hyperparameters of HLE+IRR+ILP (Experiment 1, DebateNet) and CRR+IRR (Experiment 2, remaining rows) models. $\alpha_{\text{CRR/IRR}}$: α parameter of CRR/IRR method.

B Details on Experiment 2

B.1 Dataset Details

Similar to Experiment 1, we split the categories into three equal-sized frequency bands. Table 9 shows threshold values for each band in the Manifesto dataset and category-frequency band assigments for Experiment 2 can be found at https://github.com/repo4supp/data_splits.

B.2 Training Details

In our experiments, for each language (Fi[5], De[6], Hu[7], Tr[8] and En[9]), we use a cased BERT variant that was trained specifically for the target language. We split the dataset into train (65%), validation (15%), and test (20%) sets and perform hyperparameter search on the development set for Experiment 2. We again use AdamW as the optimizer and cross-entropy as the loss function. We perform grid search for hyperparameter optimization and use the hyperparameters leading highest average F1 score on the development set. Following lower and upper bounds have been applied during search for each hyperparameter: learning rate:[1e-5, 5e-2], epoch:[5, 30], mini-batch size:[16, 32], dropout:[0.1,0.6], α_{CRR}:[0.01,0.6], α_{IRR}:[0.01,0.6] β:[0.01,0.6]. The hyperparameters for the best model (CRR+IRR), for each language and training set, are listed in Table 8.

B.3 Results Details

As the Manifesto corpus is one of the largest CSS datasets available and its size is arguably beyond the scope of typical CSS projects, we train each model variant multiple times using incrementally larger percentages (25%, 50% and 100% of the full training set) of the training data, keeping the test set constant.

Table 10 and Table 11 show the results for the 50% condition. We observe similar patterns as in 25% case: While the gap between performance of the Base model and the CRR+IRR model becomes less pronounced, CRR+IRR always yields better F1-Scores than the Plain model under 50% training data case. Furthermore, a comparison of the columns CRR and IRR with the column Base in Table 10 reveals that in most of the languages we considered, these extensions still able to outperform plain model when they are used stand-alone. Next, we investigate impact of hierarchical structure on three equal sized category frequency bands for the 50% case. Table 11 shows the results. We find that stand-alone CRR and stand-alone IRR yields the highest improvements for low frequency band in Hu and Tr and CRR+IRR achieves best results in Fi, De and En. Results in Mid and High rows of Table 11 also indicate that the extension

[5]https://github.com/TurkuNLP/FinBERT
[6]https://deepset.ai/german-bert
[7]https://hlt.bme.hu/en/resources/hubert
[8]https://github.com/dbmdz/berts
[9]https://huggingface.co/bert-base-cased

Lang	Freq.	25% Threshold	50% Threshold	100% Threshold
Fi	Low	1 ≥f≥ 12	1 ≥f≥ 23	2≥f≥ 52
	Mid	14≥f≥ 55	24≥f≥ 110	53≥f≥ 215
	High	57≥f≥ 417	111≥f≥ 867	221≥f≥ 1666
De	Low	3 ≥f≥ 56	5 ≥f≥ 98	6≥f≥ 201
	Mid	59≥f≥ 196	99≥f≥ 391	202≥f≥ 764
	High	204≥f≥ 951	401 ≥f≥ 1866	785 ≥f≥ 3655
Hu	Low	1 ≥f≥ 31	1 ≥f≥ 63	2≥f≥ 124
	Mid	37≥f≥ 147	69≥f≥ 276	133 ≥f≥ 560
	High	168 ≥f≥ 772	357 ≥f≥ 1541	697 ≥f≥ 3046
Tr	Low	1 ≥f≥ 33	1 ≥f≥ 67	1≥f≥ 130
	Mid	34≥f≥ 166	68≥f≥ 316	137≥f≥ 628
	High	187 ≥f≥ 937	380 ≥f≥ 1862	739 ≥f≥ 3720
En	Low	2 ≥f≥ 22	4 ≥f≥ 42	4≥f≥ 91
	Mid	23≥f≥ 84	49≥f≥ 180	97≥f≥ 356
	High	101 ≥f≥ 536	188 ≥f≥ 1122	368≥f≥ 2315

Table 9: Experiment 2 (single-label statement classification): Threshold values for frequency bands.

Lang	Base			CRR			IRR			CRR + IRR		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Fi	43.8	43.4	42.5	44.3	42.7	42.5	43.7	42.5	42.2	45.8	43.8	**43.9**
De	37.7	37.8	37.1	39.4	37.9	38.1	38.6	37.7	37.7	40.0	38.0	**38.5**
Hu	42.1	40.0	40.1	43.4	40.8	41.1	43.0	39.4	39.9	44.9	40.7	**41.2**
Tr	50.9	46.5	47.1	49.9	46.9	47.2	52.9	48.6	**49.2**	51.8	47.7	48.0
En	33.4	31.9	32.0	34.9	33.8	33.8	33.0	32.6	32.1	35.4	34.9	**34.2**

Table 10: Experiment 2 (single-label statement classification): Macro-averaged Precision, Recall, F1 scores for the Manifesto dataset (Overall, trained on 50% of the data)

methods boost the performance of the Base model on mid and high frequency bands as well.

Finally, Table 12 and Table 13 present results for the 100% condition. Unlike in the 25% and 50% cases, we see that all of the extended models are outperformed by the Base model in terms of overall F1-Score for Hungarian and Turkish, which indicates that incorporating hierarchical information into the models does not always lead to better results in a high data regime. When we look at per frequency band performance, however, we see that it is still useful to include hierarchical information into the models: the CRR+IRR model yields the best F-score for low frequency band in four languages out of five.

Lang	Freq band	Base			CRR			IRR			CRR + IRR		
		P	R	F1	P	R	F1	P	R	F1	P	R	F1
Fi	Low	26.6	28.8	25.8	27.9	23.5	23.9	22.8	24.0	21.8	29.6	28.4	**27.1**
	Mid	44.4	39.6	41.2	44.7	43.7	43.6	45.9	42.7	43.7	48.4	42.5	**44.9**
	High	61.3	62.6	61.5	61.1	61.9	61.2	63.5	62.0	**62.4**	60.4	61.4	60.7
De	Low	23.1	22.8	21.8	26.3	22.4	23.4	25.1	22.6	23.0	28.1	24.1	**25.4**
	Mid	39.9	42.0	40.5	42.4	41.1	**41.3**	41.1	42.1	41.2	43.1	39.6	40.9
	High	51.5	50.2	50.7	51.0	52.0	**51.2**	51.0	50.2	50.4	50.3	51.6	50.7
Hu	Low	25.7	19.4	20.2	27.9	21.8	**22.4**	28.6	18.5	20.9	30.5	19.6	21.0
	Mid	43.8	43.9	43.4	46.0	42.8	44.1	42.7	43.6	42.6	45.9	44.5	**44.8**
	High	57.9	57.9	57.8	57.0	59.1	57.7	58.5	57.3	57.3	59.0	59.2	**58.8**
Tr	Low	37.9	24.9	27.0	34.0	24.9	26.4	41.9	28.8	**31.2**	41.0	27.1	29.2
	Mid	51.7	49.2	50.1	52.3	50.7	51.2	52.2	51.1	**51.4**	50.7	51.3	50.6
	High	63.2	65.4	64.1	63.4	65.2	64.1	64.4	65.9	**65.1**	63.7	64.8	64.1
En	Low	15.0	10.0	11.5	17.3	13.4	14.4	13.1	8.8	9.9	19.2	16.1	**16.8**
	Mid	33.8	33.6	33.1	35.0	34.3	34.2	35.3	34.8	**34.3**	33.9	34.3	32.4
	High	51.4	52.2	51.5	52.5	53.6	52.8	50.7	**54.2**	52.1	53.0	54.4	53.3

Table 11: Experiment 2 (single-label statement classification): Macro-averaged Precision, Recall, F1 scores for the Manifesto dataset (by frequency band, trained on 50% of the data)

Lang	Base			CRR			IRR			CRR + IRR		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Fi	47.0	48.1	46.7	48.1	48.7	47.8	47.1	48.3	46.9	47.6	51.2	**48.1**
De	40.4	40.9	40.2	41.3	41.2	40.9	41.8	40.0	40.2	42.4	40.8	**41.2**
Hu	47.8	43.9	**44.6**	45.0	41.4	42.3	47.1	42.8	43.8	43.4	45.0	43.6
Tr	56.7	55.7	**55.5**	56.4	54.2	54.3	55.6	53.9	53.6	55.9	54.6	54.5
En	38.5	35.7	35.9	40.2	36.3	37.2	37.8	36.1	36.5	38.4	38.2	**37.8**

Table 12: Experiment 2 (single-label statement classification): Macro-averaged Precision, Recall, F1 scores for the Manifesto dataset (Overall,trained on 100% of the data)

Lang	Freq band	Base			CRR			IRR			CRR + IRR		
		P	R	F1	P	R	F1	P	R	F1	P	R	F1
Fi	Low	29.1	34.8	30.3	31.6	34.2	31.6	27.8	31.5	28.2	30.0	39.3	**31.7**
	Mid	49.4	46.7	47.4	50.5	48.8	49.3	50.6	49.3	49.2	49.7	50.3	**49.6**
	High	63.4	63.6	63.2	63.0	63.8	63.3	63.9	65.0	**64.2**	64.0	64.7	64.0
De	Low	24.1	26.0	24.2	28.4	26.8	27.1	29.2	25.2	24.9	30.6	27.7	**28.3**
	Mid	44.9	45.4	**44.8**	43.8	45.5	44.4	44.4	43.8	43.3	44.4	43.0	43.4
	High	54.0	53.1	**53.5**	53.1	53.0	52.9	53.3	52.8	52.9	53.5	53.3	53.2
Hu	Low	35.6	24.8	27.6	30.4	20.5	23.6	33.2	23.4	26.3	27.2	31.2	**28.4**
	Mid	47.6	47.2	**46.9**	47.0	45.4	45.5	48.6	45.8	46.3	42.9	46.2	43.8
	High	60.8	60.8	**60.2**	58.4	59.6	58.7	60.1	60.1	59.9	61.0	58.3	59.5
Tr	Low	41.3	40.9	**39.7**	41.8	37.6	37.6	38.9	35.1	34.3	41.5	37.9	37.9
	Mid	59.0	55.9	**57.1**	58.2	55.5	56.2	58.0	56.2	56.6	57.1	56.7	56.5
	High	70.5	71.1	70.7	70.0	70.3	69.9	70.8	71.5	**71.0**	70.0	70.2	69.9
En	Low	23.2	13.3	15.7	25.2	16.7	19.2	17.7	15.6	16.1	22.4	21.5	**21.2**
	Mid	38.0	39.3	37.9	40.8	36.9	37.9	41.6	36.6	**38.5**	38.4	36.6	37.0
	High	55.0	55.6	55.1	55.5	56.6	55.7	55.3	57.4	**56.1**	55.2	57.5	**56.1**

Table 13: Experiment 2 (single-label statement classification): Macro-averaged Precision, Recall, F1 scores for the Manifesto dataset (by frequency band, trained on 100% of the data)