Evaluating Extreme Hierarchical Multi-label Classification

Enrique Amigó
UNED
Madrid, Spain
enrique@lsi.uned.es

Agustín D. Delgado
UNED
Madrid, Spain
agustin.delgado@lsi.uned.es

Abstract

Several natural language processing (NLP) tasks are defined as a classification problem in its most complex form: Multi-label Hierarchical Extreme classification, in which items may be associated with multiple classes from a set of thousands of possible classes organized in a hierarchy and with a highly unbalanced distribution both in terms of class frequency and the number of labels per item. We analyze the state of the art of evaluation metrics based on a set of formal properties and we define an information theoretic based metric inspired by the Information Contrast Model (ICM). Experiments on synthetic data and a case study on real data show the suitability of the ICM for such scenarios.

1 Introduction

Many natural language processing (NLP) problems involve classification, such as sentiment analysis, entity linking, etc. However, the adequacy of evaluation metrics is still an open problem. Different metrics such as Accuracy, F-measure or Macro Average Accuracy (MAAC) may differ substantially, seriously affecting the system optimization process. For example, assigning all elements to the majority class may be very effective according to Accuracy and score low according to MAAC.

In addition, in many scenarios such as tagging in social networks (Coope et al., 2018) or topic identification (Yu et al., 2019), the classifier must assign several labels to each item (multi-label classification). This greatly complicates the evaluation problem since, in addition to the class specificity (frequency), other variables appears such as the distribution of labels per item in the gold standard, the excess or absence of labels in the system output, etc.

The evaluation problem becomes even more complicated if we consider hierarchical category structures, which are very common in NLP. For example, toxic messages are divided into different types of toxicity (Fortuna et al., 2019), named entities could be organized in nested categories (Sekine and Nobata, 2004), etc. In these scenarios, the category proximity in the hierarchical structure is an additional variable.

Even, the problem can be further complicated. *Extreme Classification* scenarios address with thousands of highly unbalanced categories (Gupta et al., 2019), where a few categories are very frequent and others completely infrequent (Almagro et al., 2020). In addition, some items have no category at all and some have many. An example scenario that we will use as a case study in this article is the labelling of adverse events in medical documents.

In this paper, we analyse the state of the art on metrics for multi-label, hierarchical and extreme classification problems. We characterize existing metrics by means of a set of formal properties. The analysis shows that different metric families satisfy different properties, and that satisfying all of them at the same time is not straightforward.

Then, propose an information-theoretic based metric inspired by the Information Contrast Model similarity measure (ICM), which can be particularized to simpler scenarios (e.g. flat, single labeled) while keeping its formal properties. Later, we define a set of five tests on synthetic data to compare empirically ICM against existing metrics. Finally, we explore a case study with real data which shows the suitability of ICM for such extreme scenarios. The paper ends with some conclusions and future work.

Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 5809 - 5819
May 22-27, 2022 ©2022 Association for Computational Linguistics

2 Background

In this section, we analyze the literature on the two main evaluation problems tackled in this paper: multi-labeling and class hierarchies, keeping the focus on extreme scenarios (numerous and unbalanced classes).

2.1 Multi-Label Classification

There are three main ways of generalizing effectiveness metrics to the multi-label scenario (Zhang and Zhou, 2014). The first one consists in modeling the problem as a ranking task, i.e. the system returns an ordered label list for each item according to their suitability. Some specific ranking metrics applied in multi-label classification displayed in (Wu and Zhou, 2017) are: *Ranking Loss*, which is a ordinal correlation measure, *one-error* which is based on Precision at 1, or *Average Precision*. Although these metrics are very common, they do not take into account the specificity of (unbalanced) classes. Jain et al. proposed the *propensity* versions of ranking metrics (Precision@k, nDCG) in order to weight classes according to their frequency in the data set (Jain et al., 2016).

Reducing the classification to a ranking problem is specially appropriate in extreme classification scenarios and simplifies the definition of metrics. However, it also has several disadvantages. First, it requires the output of the classifier to be in ranking format, and that does not fit many scenarios. For example, annotating posts in social networks requires predicting the amount of tags to be assigned to the post. For this reason, we focus on classification outputs, so ranking based metrics are out of our scope.

Apart from ranking metrics, multi-label effectiveness metrics have been categorized into label- and example-based metrics (Tsoumakas et al., 2010; Zhang and Zhou, 2014). **Label-based evaluation measures** assess and average the predictive performance for each category as a binary classification problem, where the negative category corresponds with the other categories. The most popular are the label-based Accuracy (LB-ACC) and F-measure (LB-F)[1]. The label-based metrics have some drawbacks. First, they do not consider the distribution of labels per item. Hits are rewarded independently of how many labels are associated to the item. Second, while items are supposed to be random samples, classes are not, so the idea of averaging results across classes is not always consistent. That is, the metric scores can vary substantially depending on how the category space is configured. Finally, if there are a large number of possible categories (extreme classification), the score contribution of any label has an upper limit of $\frac{1}{|\mathcal{C}|}$, being $\mathcal{C}$ the set of categories. This limit can be problematic, specially when labels are unbalanced and numerous.

On the other hand, the **example-based metrics** compute for each object, the proximity between predicted and true label sets ($s(d) = \{c_1^s, .., c_n^s\}$ and $g(d) = \{c_1^g, .., c_n^g\}$). Some popular ways to match category sets in multi-label classification evaluation are the Jaccard similarity (EB-JACC) which is computed as $\frac{|s(d) \cap g(d)|}{|s(d) \cup g(d)|}$ (Godbole and Sarawagi, 2004), or the precision $\left(\frac{|s(d) \cap g(d)|}{|s(d)|}\right)$, recall $\left(\frac{|s(d) \cap g(d)|}{|g(d)|}\right)$ and their F combination (EB-F). Another example-based metric is the *Hamming Loss* (EB-HAMM) (Zhang et al., 2006) which matching function is defined as: $\frac{|s(d)\ \mathrm{XOR}\ g(d)|}{|C_g|}$ where C_g represents the set of categories annotated in the gold standard. *Subset Accuracy* (EB-SUBACC) (Ghamrawi and McCallum, 2005) is a more strict measure due to it requires exact matching between both category sets. Notice that all example-based multi-label metrics converge to Accuracy in the single-label scenario. On the other hand, there are some situations in which these metrics are undefined. If both the gold standard and the system output label sets are empty, the maximum score is usually assigned to the item.

The main drawback of these approaches is that they do not take into account the specificity of classes (i.e. unbalanced classes in extreme classification). The label propensity applied over precision and recall for single items can solve this lack. Each accurate class in the intersection is weighted according to the class *propensity* p_c (Jain et al., 2016):

$$\mathrm{Prop}_P(i) = \frac{\sum_{c \in s(i) \cap g(i)} \frac{1}{p_c}}{|s(i)|}$$

$$\mathrm{Prop}_R(i) = \frac{\sum_{c \in s(i) \cap g(i)} \frac{1}{p_c}}{|g(i)|}$$

The propensity factor p_c for each class is computed as: $p_c = \frac{1}{1+Ce^{-A\log_2(N_c+B)}}$ where N_c is

[1] In the single label scenario, the label-based F-measure converges to the traditional F and the label-based accuracy is proportional to the traditional ACC.

the number of data points annotated with label c in the observed ground truth data set of size N and A, B are application specific parameters and $C = (logN - 1)(B + 1)^A$. In our experiments, we set the recommended parameter values $A = 0.55$ and $B = 1.5$.

However, propensity precision and recall values are not upper bounded as $\frac{1}{p_c}$ tends to infinite when p_c tends to zero. In order to solve this issue, in our experiments we replace the normalization factors $|s(i)|$ and $|g(i)|$ with the accumulation of inverse propensities in the system output or the gold standard. We also add the empty class $c_\emptyset$ in both the system output and the gold standard in order to capture the specificity of classes in the mono-label scenario:

$$\text{Prop}_P(i) = \frac{\sum_{c \in s'(i) \cap g'(i)} \frac{1}{p_c}}{\sum_{c \in s'(i)} \frac{1}{p_c}}$$

$$\text{Prop}_R(i) = \frac{\sum_{c \in s'(i) \cap g'(i)} \frac{1}{p_c}}{\sum_{c \in g'(i)} \frac{1}{p_c}}$$

where $s'(i) = s(i) \cup \{c_\emptyset\}$ and $g'(i) = g(i) \cup \{c_\emptyset\}$. Propensity F-measure (PROP-F) is computed as the harmonic mean of these values.

2.2 Hierarchical Classification

There are different taxonomies of hierarchical classification metrics (Costa et al., 2007; Kosmopoulos et al., 2013). Kosmopoulos et al. distinguish between pair and set-based metrics. **Pair-based metrics** weight hits or misses according to the distance between categories in the hierarchy. This distance depends on the number of intermediate nodes (Wang et al., 1999; Sun and Lim, 2001), with the disadvantage that the specificity of the categories is not taken into account. Depth-based distance metrics include the class depth in the metric (Blockeel et al., 2002). However, the depth of the node is not sufficient to model its specificity since depending on their frequency, leaf nodes at the first levels may be more specific than leaf nodes at deeper levels.

It is possible to compare the predicted and true single labels by means of standard ontological similarity measures such as Leackock and Chodorow (path-based) (Leacock and Chodorow, 1998), Wu and Palmer (Wu and Palmer, 1994), Resnik (depth-based) (Resnik, 1999), Jiang and Conrath (Jiang and Conrath, 1997) or Lin (Lin, 1998) similarities. The last two are based on the notion of Information Content (IC) or category specificity, i.e., the amount of items belonging to the category or any of its descendants.

However, extending pair-based hierarchical metrics to the multi-label scenario is not straightforward. Sun and Lim extended Accuracy, Precision and Recall measures for ontological distance based metrics (Sun and Lim, 2001). This method has two drawbacks. First, it requires defining a neutral hierarchical distance, i.e., an acceptable distance threshold for range normalization purposes. The second drawback is that it inherits the weaknesses of label-based metrics (see previous section). Blockeel et al. proposed computing a kernel and thus define a Euclidean distance metric between sums of class values (Blockeel et al., 2002). The drawback is that they assume a previously defined distance metric between categories and the origin and between different categories. Information based ontological similarity measures such as Jiang and Conrath or Lin's similarity do not have an upper bound which is necessary for the calculation of accuracy and coverage.

On the other hand, **set-based metrics** (also called hierarchical-based) consider the ancestor overlap (Kiritchenko et al., 2004; Costa et al., 2007). More concretely, hierarchical precision and recall are computed as the intersection of ancestor divided by the amount of ancestors of the system output category and of the gold standard respectively[2]. Their combination is the Hierarchical F-measure (HF). Since these metrics are based on category set overlap, they can be applied as example based multi-label classification by joining ancestors and computing the F measure. Their drawback is that the specificity of categories is not strictly captured since they assume a correspondence between specificity and hierarchical deepness. However, this correspondence is not necessarily true. Categories in first levels can be infrequent whereas leaf categories can be very common in the data set.

In this paper, we propose an information theoretic similarity measure called Information Contrast Model (ICM). ICM is an example-based metric as it is computed per item. Just like HF, ICM is a set-based multi-label metric as it computes the similarity between category sets. Unlike HF, ICM takes into account the statistical specificity of categories.

[2] In our experiments, when computing the ancestor overlap we consider the common empty label (root class) in order to avoid undefined situations

3 Formal Properties

In order to define the set of desirable properties, we formalize both the gold standard g and the system output s as sets of item/category assignments $(i, c) \in \mathcal{I} \times \mathcal{C}$, where $\mathcal{I}$ and $\mathcal{C}$ represent the set of items and categories respectively. We will denote as $P(c_j)$ the probability of items to be classified as c_j in the gold standard $(P((i, c_j) \in g | i \in \mathcal{I}))$. We also assume that the categories in the hierarchical structure are subsumed. For instance, items in a *PERSON_NAMED_ENTITY* category are implicitly labeled with the parent category *NAMED_ENTITY*. The common ancestor with maximum depth is denoted as $\texttt{lso}(c_1, c_2)$ and the descendant categories are denoted as $\texttt{Desc}(c)$ including itself.

Note that we do not claim that all properties are necessary in any scenario. The purpose of this article is to provide at least one metric that is capable of capturing all aspects simultaneously when necessary.

The first property is related to hits. In order to make this aspect independent from the ability of the metrics to capture hierarchical relationships or multi-labeling, we define monotonicity over hits in the simplest case (flat single label scenario):

Property 1 *[Strict Monotonicity] A hit increases effectiveness. Given a flat single label category structure, if* $(i, c) \in g \setminus s$, *then*[3] $\texttt{Eff}(s \cup \{(i, c)\}) > \texttt{Eff}(s)$

The next two properties state that the specificity of both the predicted and the true category affects the metric score. That is, an error or a hit in an infrequent category should have more effect than in the majority category. For instance, identifying a rare symptom in a medical report should be rewarded more than identifying a common malady present in the vast majority of patients. In addition, both the specificity of the actual category and the specificity of the category predicted by the system must be taken into account. Again, we make this aspect independent of hierarchical structures and multi-labeling.

Property 2 *[True Category Specificity] Given a flat single label category distribution, if* $P(c_1) < P(c_2)$ *and* $(i, c_1), (i, c_2) \in g \setminus s$, *then* $\texttt{Eff}(s \cup \{(i, c_1)\}) > \texttt{Eff}(s \cup \{(i, c_2)\})$.

Property 3 *[Wrong Category Specificity] Given a flat single label category distribution, if* $P(c_1) < P(c_2)$ *and* $(i, c_1), (i, c_2) \notin g \cup s$, *then* $\texttt{Eff}(s \cup \{(i, c_1)\}) < \texttt{Eff}(s \cup \{(i, c_2)\})$.

The following property captures the effect of the hierarchical category structure. A common element of any hierarchical proximity measure is that it is monotonic with respect to the common ancestor. That is, our brother is always closer to us than our cousin, regardless of which family proximity criterion is applied.In this property we do not consider multi-labelling.

Property 4 *[Hierarchical Proximity] Under equiprobable categories* $(P(c_1) = P(c_2) = P(c_3))$, *the deepness of the common ancestor affects similarity. Given a single label hierarchical category structure, if* $s(i) = \emptyset$, $g(i) = c_1$ *and* $\texttt{lso}(c_1, c_2) \in \texttt{Desc}(\texttt{lso}(c_1, c_3))$ *then* $\texttt{Eff}(s \cup \{(i, c_2)\}) > \texttt{Eff}(s \cup \{(i, c_3)\})$.

The last two properties are related with the multi-labeling problem. Property 5 rewards the amount of predicted categories per item.

Property 5 *[Multi-label Monotonicity] The amount of predicted categories increases effectiveness. Given a flat multi-label category structure, if* $(i, c) \in g \setminus s$, *then* $\texttt{Eff}(s \cup \{(i, c)\}) > \texttt{Eff}(s)$

Property 6 rewards hits on multiple items regarding a single item with multiple categories. To understand the motivation for this property, we can consider an extreme case. Identifying 1000 symptoms in one patient report is of less health benefit than identifying one symptom in 1000 patients.

Property 6 *[Label vs. Item Quantity] n hits on different items are more beneficial than n labels assigned to one item. Given a flat multi-label category distribution, if* $\forall j = 1..n((j, c_j) \in g \setminus s)$ *and* $\forall j = 1..n, i > n((i, c_j) \in g \setminus s)$ *then* $\texttt{Eff}(s \cup \{(1, c_1), .., (n, c_n)\}) > \texttt{Eff}(s \cup \{(i, c_1), .., (i, c_n)\})$.

4 Metric Analysis

In this section, we analyze existing metrics on the basis of the proposed formal properties (Table 1). Most of metrics satisfy Strict Monotonicity in single label scenarios. The label-based metric LB-F captures the true and wrong category specificity via the recall component. The example-based metric PROP-F (modified as described in Section 2) captures these properties via the propensity factor. Notice that the original propensity F-measure does not capture the wrong category specificity (Property 3) given that the p_c factor is applied only to

[3] Notice that $x \in X \setminus Y \equiv x \in X \wedge x \notin Y$

Table 1: Metric and Formal Properties

Family	Metrics	Constraints					
		Strict Monotonicity	True Category Specificity	Wrong Category Specificity	Hierarchical Proximity	Multi-label Monotonicity	Label vs. Item Quantity
Label Based	Accuracy (LB-ACC)	✓	-	-	-	✓	-
	F measure (LB-F)	✓	✓	✓	-	✓	-
Example Based	Jaccard (EB-JACC)	✓	-	-	-	✓	✓
	Hamming (EB-HAMM)	✓	-	-	-	✓	-
	Subset Acc. (EB-SUBACC)	✓	-	-	-	-	✓
	F-measure (EB-F)	✓	-	-	-	✓	✓
	Propensity F (PROP-F)	✓	✓	✓	-	✓	✓
Set Based	Hierarchical F (HF)	✓	-	-	✓	✓	✓
Ontological Similarity Measures (single-label classification)	Leacock and Chodorows	✓	-	-	✓	-	-
	Wu and Palmer	✓	-	-	✓	-	-
	Resnik	✓	✓	-	✓	-	-
	Jiang and Conrath	✓	✓	✓	✓	-	-
	Lin's similarity	✓	✓	✓	✓	-	-
	ICM	✓	✓	✓	✓	✓	-

hits. In addition, both kind of metrics do not capture hierarchical structures. The contribution of example regarding label-based metrics is that, as label-based metrics are computed item by item, the property Label vs. Item Quantity is satisfied (Property 6). The exception is EB-HAMM which does not normalize the results with respect to the amount of labels assigned to the item.

Unlike previous metrics, the set based F-measure (HF) captures the hierarchical structure (Property 4). However, it does not capture the category specificity (properties 2 and 3). Some information-based ontological similarity measures, (Lin and Jiang & Conrath) capture both the category specificity and the hierarchical structure. However, they are not defined for multi-label classification (properties 5 and 6). In sum, different metric families satisfy different properties, and that satisfying all of them at the same time is not straightforward. The properties of ICM are described in the next section.

5 Information Contrast Model

The *Information Contrast Model* (ICM) is a similarity measure that unifies measures based on both object feature sets and Information Theory (Amigó et al., 2020). Given two feature sets A and B, ICM is computed as:

$$\text{ICM}(A,B) = \alpha_1 IC(A) + \alpha_2 IC(B) - \beta IC(A \cup B)$$

Where $IC(A)$ represents the information content $(-log(P(A))$ of the feature set A. In our scenario, objects are items to be classified and features are categories. The intuition is that the more the category sets are unlikely to occur simultaneously (large $IC(A \cup B)$), the less they are similar. Given a fixed joint IC, the more the category sets are specific ($IC(A)$ and $IC(B)$), the more they are similar. ICM is grounded on similarity axioms supported by the literature in both information access and cognitive sciences. In addition, it generalizes the Pointwise Mutual Information and the Tversky's linear contrast model (Amigó et al., 2020).

5.1 Computing Information Content

The IC of a single category corresponds with the probability of items to appear in the category or any of its descendant. It can be estimated as follows:

$$IC(c) = -log_2(P(c)) \simeq -log_2\left(\frac{|\bigcup_{c' \in \{c\} \cup \text{Desc}(c)} \mathcal{I}_{c'}|}{|\bigcup_{c' \in \mathcal{C}} \mathcal{I}_{c'}|}\right)$$

where $\mathcal{I}_{c'}$ represent the set of items assigned to the category c' and $\text{Desc}(c)$ represents the set of descendant categories. In order to estimate the IC of category set, we state the following considerations. The first one is that, given two categories A and B the common ancestor represents their intersection in terms of feature sets:

$$\{c_i\} \cap \{c_j\} = \texttt{lso}(c_i, c_j) \qquad (1)$$

The second consideration is that we assume *Information Additivity*, i.e. the IC of the union of two sets is the sum of their IC's minus the IC of its intersection:

$$IC(\{c_i\}\cup\{c_j\}) = IC(c_i)+IC(c_j)-I(\{c_i\}\cap\{c_j\}) \quad (2)$$

Equations 1 and 2 are enough to compute ICM in the single label scenario. Generalizing for category sets:

$$IC(\{c_1, c_2, .., c_n\}) = IC\left(\bigcup_i \{c_i\}\right) =$$
$$IC(c_1) + IC(\{c_2, .., c_n\}) - IC(\{c_1\} \cap \{c_2, .., c_n\})$$

where, according to the transitivity property;

$$\{c_1\} \cap \{c_2, .., c_n\} = \bigcup_{i=2..n} (\{c_1\} \cap \{c_i\})$$

and according to Equation 1, it is equivalent to $\bigcup_{i=2..n}\{\mathtt{lso}(c_1, c_i)\}$. Then, we finally obtain a recursive function to compute the IC of a category set:

$$IC(\{c_1, c_2, .., c_n\}) =$$
$$IC(c_1) + IC\left(\bigcup_{i=2..n} \{c_i\}\right) - IC\left(\bigcup_{i=2..n} \{\mathtt{lso}(c_1, c_i)\}\right)$$

In the case of ICM, it is possible the need for estimating the IC of classes that do not appear in the gold standard. Therefore, we have not evidence about its frequency or probability. We apply a smoothing approach by considering the minimum probability $\frac{1}{|\mathcal{I}|}$.

5.2 Parameterization and Formal Properties

On the basis of five general similarity axioms, in (Amigó et al., 2020) it is stated that the ICM parameters should satisfy $\alpha_1, \alpha_2 < \beta < \alpha 1 + \alpha 2$. We propose the parameter values $\alpha_1 = \alpha_2 = 2$ an $\beta = 3$. This parameterization leads to the following instantiations for each particular classification scenario. In the hierarchical mono-label scenario, it becomes into (equations 1 and 2):

$$\mathrm{ICM}(c_1, c_2) = -IC(c_1) - IC(c_2) + 3IC(\mathtt{lso}(c_1, c_2)) \quad (3)$$

which is similar to the Jiang and Conrath ontological similarity measure. In the flat multi-label scenario, it becomes into:

$$\mathrm{ICM}(C, C') = \sum_{c \in C \cap C'} IC(c) - \sum_{\substack{c \in C \setminus C' \\ \cup C' \setminus C}} IC(c) \quad (4)$$

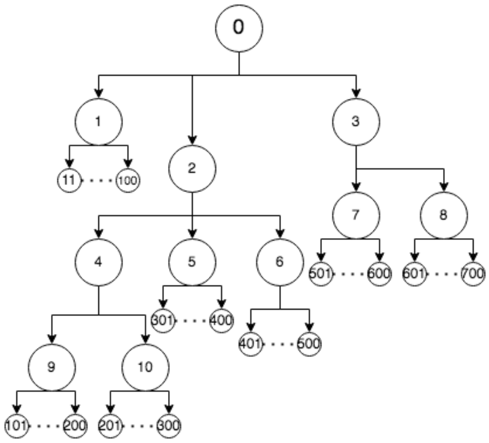

Figure 1: Category hierarchy for experiments on synthetic data.

which is an information additive example-based metric. That is, the information content of the common categories minus the differences. Finally, in the traditional flat mono-label scenario, it becomes into:

$$\mathrm{ICM}(c_1, c_2) \simeq \begin{cases} IC(c_1) & \text{if } c_1 = c_2 \\ -IC(c_1) - IC(c_2) & \text{i.o.c.} \end{cases} \quad (5)$$

which corresponds with Accuracy weighted according to the information content of categories.

According to the flat mono-label instantiation (Equation 5) $\mathrm{ICM}_{\alpha_1=\alpha_2=2,\beta=3}$ satisfies the properties 1 2 and 3. According to the single label hierarchical instantiation (Equation 3) Property 4 is satisfied. According to the flat multi-label instantiation (Equation 4), Property 5 is satisfied. Unfortunately, the label vs item quantity property is not strictly satisfied given that the gain per hit is additive in non hierarchical scenarios (Property 6). However, in the experiments we will see that the hit gain on items with many categories is smoothed out if the categories are related to each other by a hierarchical structure.

6 Experiments on Synthetic Data

Different evaluation aspects such as error rate, category specificity, hierarchical structures, etc., may have more or less weight depending on the scenario. These aspects correspond to the formal properties defined in the previous section. We perform a set of tests in order to quantify the suitability of metrics with respect to each property or evaluation aspect.

Table 2: Experiments over synthetic data. Ratio of cases in which the *best* synthetic output outperforms the *worst*.

Metrics	Metric Test				
	Sensitivity to error rate	True Category Specificity	Wrong Category Specificity	Hierarchical Proximity	Item Specificity
Accuracy (LB-ACC)	100%	50%	50%	50%	50%
F-measure(LB-F)	84.98%	100%	100%	52.65%	26.38%
Jaccard (EB-JACC)	86.59%	50%	50%	50%	100%
Hamming (EB-HAMM)	100%	50%	50%	50%	50%
Subset Accuracy (EB-SUBACC)	91.79%	50%	50%	50%	96.80%
Example Based F-measure (EB-F)	79.43%	50%	50%	50%	100%
Hierarchical F-measure (HF)	81.03%	46.55%	42.04%	100%	99.90%
Propensity F-Measure (PROP-F)	85.64%	100%	100%	53.15%	100%
ICM	96.10%	100%	100%	100%	74.77%

First, we generate the following synthetic data set. First, we definea hierarchical structure structure of 700 categories exposed in Figure 1. Note that categories $\{1..10\}$ are parent categories spread throughout the hierarchy, and categories $\{11..700\}$ are leaf categories. Secondly, We distributed 100 items across all categories. We generate assignments for each pair item/category (i, c) with a probability of $p_i \cdot p_c$ where $p_i = \max\left(\frac{51-i}{2225}, \frac{1}{2225}\right)$ with $i = 1..1000$ and $p_c = \frac{\max\left(\frac{512}{c},1\right)}{1713}$ where $c = 1..700$. We repeat this 1000 times. The result is a distribution $(300, 150, 40, .., 0.6, 0.6)$ items per category and $(22.5, 22, 21.6, 21.1, ..., 0.5, 0.5)$ labels per item. The purpose is to ensure unbalanced assignments across items and classes. We generate 1000 gold standards by reordering the category identifiers c each time in the p_c computation in order to alter the distribution of items in the hierarchical structure.

We consider in this experiment the metrics label-based Accuracy and F-measure (LB-ACC and LB-F), the example-based metrics Hamming (EB-HAMM), Jaccard (EB-JACC), Subset Accuracy (EB-SUBACC), F-measure (EB-F) and Propensity F-measure (PROP-F), the Hierarchical F-measure (HF) and ICM. The ontological similarity metrics are discarded given that they are not defined for the multi-label case. Ranking based metrics are discarded as the synthetic data set does not include graded assignments.

After this, we perform the following tests by comparing two noisy versions of the gold standard. The test result is the percentage of cases in which the hypothetically worse noised output is outscored by the best noised output (Table 2). Ties count 0.5.

In the first experiment referred in Table 2 as **Sensitivity to Error Rate**, We ran an error insertion procedure 1000 times on the goldstandard, with a probability of 0.09 and 0.1 for the best and worst output respectively. On average we will have 9 and 10 errors respectively. Each error consists of randomly choosing one of the 1000 assignments (i, c) of the goldstandard and removing it. For all metrics the best output outperforms the worst output in more that 50% of cases. LB-ACC and EB-HAMM seems to be specially sensitive to the error rate. This is due to the fact that they do not consider other aspects such as the category specificity or the hierarchical proximity. Surprisingly, ICM achieves a relatively high error rate sensitivity although it also consider other aspects. We do not have a clear explanation for this.

The second experiment is the **True Category Specificity** test. The intuition is that a gap in a frequent category should have less effect than a gap in an infrequent category. With an error rate of 0.05, for the best output, we remove a single label assignment randomly selected from all the goldstandard. For the worst output, we first select randomly a category and then we remove an assignment from this category. The result is that the best output tends to concentrate the gaps in frequent categories to a greater extent than the worst output. At the table shows, the metrics that satisfy the corresponding property achieve high scores (LB-F, PROP-F and ICM).

The third experiment is the **Wrong Category Specificity** test. The intuition is that a wrong assignment in a frequent category should have less effect than a wrong assignment in an infrequent category. With an error rate of 0.05, we select an

Table 3: Experimental results over real data. Metrics values for each baseline. The normalised value with respect to the maximum and minimum of the five baseline scores is shown in brackets.

Metrics	Baselines: Metric result (normalization)				
	ALL NONE	**MOST FREQ.**	**MATCH 75%**	**SVM DESCR.**	**SVM CODES**
Accuracy (LB-ACC)	0.9999 (1.00)	0.9997 (0.00)	0.9998 (0.50)	0.9999 (1.00)	0.9999 (1.00)
F-measure(LB-F)	0.9248 (0.79)	0.9248 (0.79)	0.9005 (0.00)	0.9273 (0.88)	0.9309 (1.00)
Jaccard (EB-JACC)	0.8395 (0.97)	0.0055 (0.00)	0.7209 (0.83)	0.8409 (0.97)	0.8644 (1.00)
-Hamming$\times 10^3$ (EB-HAMM)	-0.0507 (0.98)	-0.254 (0.00)	-0.117 (0.66)	-0.0506 (0.98)	-0.0472 (1.00)
Subset Accuracy (EB-SUBACC)	0.8395 (0.97)	0.0027 (0.00)	0.7205 (0.83)	0.8392 (0.97)	0.8573 (1.00)
Example Based F (EB-F)	0.8395 (0.96)	0.0066 (0.00)	0.7210 (0.83)	0.8416 (0.97)	0.8670 (1.00)
Hierarchical F (HF)	0.8902 (0.97)	0.2750 (0.00)	0.8054 (0.83)	0.8913 (0.97)	0.9080 (1.00)
Propensity F (PROP-F)	0.8893 (0.96)	0.5024 (0.00)	0.7742 (0.67)	0.8903 (0.96)	0.9030 (1.00)
ICM Average	-2.2062 (0.92)	-8.6158 (0.00)	-5.5761 (0.43)	-2.1107 (0.94)	-1,700 (1.00)

assignment (i, c) randomly from items with a single label. For the best output we replace c with the most frequent class different than c. For the worst output, we replace c with a randomly selected category different than c. We obtain the same result than in the previous experiment.

The fourth experiment is the **Hierarchical Similarity** test. The intuition is that the more a wrong assignment is far away from the correct category, the more it has effect in the effectiveness score. Again, with an error rate of 0.05, we select an assignment (i, c) randomly from single labeled items with leaf categories. For the best output we replace c with a sister wrong category. For the worst output, we replace c with a randomly selected wrong category. Again, the metrics that satisfy the corresponding property achieve high scores.

The last test is **Item Specificity**. The intuition is that a wrong assignment in an item with many labels should have more effect than an error in an item with one or a few labels. For the best output, for each error insertion iteration, we randomly select an assignment (i, c) (with the same error rate 0.05). For the worst output, we randomly select an item i, and we take one of its assignments (i, c). In both cases, the category is replaced with a randomly selected wrong label. In other words, we distribute errors uniformly across item/category assignments in the best output and we distribute errors uniformly across items in the worst output. The effect is that the best output concentrates errors in items with many labels. Again, those metrics that satisfy the corresponding metric achieve high performance. The label-based F-measure tends to reward the worst output. The reason is that items with many labels tend to concentrate diverse labels. Therefore, the label-based F measure penalizes the best output. As discussed in the previous section, although ICM does not satisfy the property, the hit gain on items with many categories is smoothed out if the categories are related to each other by a hierarchical structure.

7 A Case Study

The problem addressed is the automatic encoding of discharge reports (Dermouche et al., 2016; Bampa and Dalianis, 2020) from a Spanish hospital to detect adverse events (AEs) from CIE-10-ES[4], the Spanish version of the tenth revision of the International Classification of Diseases (ICD-10).

AEs detection fits to the scenario tackled in this article due to the following reasons: (i) **Extreme**: CIE-10-ES contains 4816 codes related to AEs, which probability follows a power-law distribution since most of them rarely appear in health records or even they do not appear; (ii) **Hierarchical**: CIE-10-ES is a hierarchy with six levels: an empty root ($c_\emptyset$ such that $IC(c_\emptyset) = 0$), and then a level composed by three-character-codes categories which can be divided into successive nested subcategories adding characters until seven-character-codes at most; and (iii) **Multi-label classification**: Each discharge report could have associated with several AEs codes.

We have used a corpus composed of 36264 real anonymized discharge reports (Almagro et al., 2020) annotated with AEs codes by experts. The corpus has been divided into three data sets, training, development and test, following the proportion 50%-30%-20% respectively. The corpus includes only 671 AEs codes of 4816 and 84% of the discharge reports have no AEs, so the data is highly biased and unbalanced.

[4] https://eciemaps.mscbs.gob.es/ecieMaps/

We have applied five simple baselines in order to analyze the behaviour of the metrics: (i) **ALL NONE** does not assign any code to each item; (ii) **MOST FREQ.** assigns the most frequent AE code in the training data set (T45.1X5A) to each item, which just appears in 68 items of 7253; (iii) **MATCH 75%** divides each item into sentences and assigns a code if a sentence contains 75% of the words of the code description avoiding stop-words; (iv) **SVM DESCR.** creates a binary classifier for each AE code in the training set using the presence of words of the AEs codes descriptions in the items as features, excepting stop-words; (v) **SVM CODES**: similar to the previous one but using as features the annotated non-AEs codes in order to check if AEs codes are related to non-AEs codes. Note that MATCH 75% is able to assign any AE, but the SVM baselines are only able to assign AEs appearing in the training data set.

Table 3 shows the metrics results obtained by each baseline. Unfortunately, with only five systems it is difficult to find differences in terms of system ranking. Therefore, we have normalised the values for each metric between the maximum and the minimum obtained across the 5 systems in order to study the relative differences of scores (values in brackets). LB-ACC, LB-F and EB-HAMM reward the absence of most of the labels in the corpus, so they are not suitable in this scenario. The rest of the metrics sort systems in the same way. The particularity of ICM is that, as shows the normalized results, the baseline MATCH 75% is penalized with respect to ALL NONE to a greater extent than in other metrics, since MATCH 75% assigns many codes incorrectly, whereas ALL NONE does not provide any information. Another slight particularity of ICM is that the system SVM CODES is rewarded against the rest of baselines to a greater extent. Notice that SVM CODES achieves 269 hits while SVM DESCR achieves 77 hits.

8 Conclusions and Future Work

The definition of evaluation metrics is an open problem for extreme hierarchical multi-label classification scenarios due to the role of several variables, for instance, a huge number of labels, unbalanced and biased label and item distributions, proximity between classes into the hierarchy, etc. Our formal analysis shows that metrics from different families (label, example, set-based, ontological similarity measures etc.) satisfy different properties and capture different evaluation aspects. The information-theoretic metric ICM proposed in this paper, combines strengths from different families. Just like example-based multi-label metrics, it computes scores by items. Just like set-based metrics, it compares hierarchical category sets. Just like some ontological similarity measures (Lin or Jiang and Conrath), it considers the specificity of categories in terms of Information Content. Our experiments using synthetic and real data show the suitability of ICM with respect to existing metrics.

ICM does not strictly hold the label vs. item quantity property. We propose to adapt ICM in order to guarantee all the formal properties as future work.

Acknowledgments

Research cooperation between UNED and the Spanish Ministry of Economy and Competitiveness, ref. C039/21-OT and in the framework of DOTT-HEALTH project (MCI/AEI/FEDER, UE) under Grant PID2019-106942RB-C32.

References

Mario Almagro, Raquel Martínez, Víctor Fresno, and Soto Montalvo. 2020. ICD-10 coding of spanish electronic discharge summaries: An extreme classification problem. *IEEE Access*, 8:100073–100083.

Enrique Amigó, Fernando Giner, Julio Gonzalo, and Felisa Verdejo. 2020. On the foundations of similarity in information access. *Inf. Retr. J.*, 23(3):216–254.

Maria Bampa and Hercules Dalianis. 2020. Detecting adverse drug events from Swedish electronic health records using text mining. In *Proceedings of the LREC 2020 Workshop on Multilingual Biomedical Text Processing (MultilingualBIO 2020)*, pages 1–8, Marseille, France. European Language Resources Association.

Hendrik Blockeel, Maurice Bruynooghe, Saso Dzeroski, Jan Ramon, and Jan Struyf. 2002. Hierarchical multi-classification. *Workshop Notes of the KDD'02 Workshop on Multi-Relational Data Mining*, pages 21–35.

Sam Coope, Yoram Bachrach, Andrej Zukov Gregoric, José Rodríguez, Bogdan Maksak, Conan McMurtie, and Mahyar Bordbar. 2018. A neural architecture for multi-label text classification. In *Intelligent Systems and Applications - Proceedings of the 2018 Intelligent Systems Conference, IntelliSys 2018, London, UK, September 6-7, 2018, Volume 1*, volume 868 of *Advances in Intelligent Systems and Computing*, pages 676–691. Springer.

Eduardo P. Costa, Ana C. Lorena, Andre C.P.L.F. Carvalho, and Alex A. Freitas. 2007. A review of performance evaluation measures for hierarchical classifiers. *AAAI Workshop - Technical Report*.

Mohamed Dermouche, Julien Velcin, Rémi Flicoteaux, Sylvie Chevret, and Namik Taright. 2016. Supervised topic models for diagnosis code assignment to discharge summaries. In *Computational Linguistics and Intelligent Text Processing - 17th International Conference, CICLing 2016, Konya, Turkey, April 3-9, 2016, Revised Selected Papers, Part II*, volume 9624 of *Lecture Notes in Computer Science*, pages 485–497. Springer.

Paula Fortuna, João Rocha da Silva, Juan Soler-Company, Leo Wanner, and Sérgio Nunes. 2019. A hierarchically-labeled Portuguese hate speech dataset. In *Proceedings of the Third Workshop on Abusive Language Online*, pages 94–104, Florence, Italy. Association for Computational Linguistics.

Nadia Ghamrawi and Andrew McCallum. 2005. Collective multi-label classification. In *Proceedings of the 14th ACM International Conference on Information and Knowledge Management*, CIKM '05, page 195–200, New York, NY, USA. Association for Computing Machinery.

Shantanu Godbole and Sunita Sarawagi. 2004. Discriminative methods for multi-labeled classification. In *Advances in Knowledge Discovery and Data Mining*, pages 22–30, Berlin, Heidelberg. Springer Berlin Heidelberg.

Vivek Gupta, Rahul Wadbude, Nagarajan Natarajan, Harish Karnick, Prateek Jain, and Piyush Rai. 2019. Distributional semantics meets multi-label learning. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, Honolulu, Hawaii, USA*, pages 3747–3754. AAAI Press.

Himanshu Jain, Yashoteja Prabhu, and Manik Varma. 2016. Extreme multi-label loss functions for recommendation, tagging, ranking & other missing label applications. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 935–944, New York, NY, USA. Association for Computing Machinery.

Jay J. Jiang and David W. Conrath. 1997. Semantic similarity based on corpus statistics and lexical taxonomy. In *Proc. of the Int'l. Conf. on Research in Computational Linguistics*, pages 19–33.

Svetlana Kiritchenko, Stan Matwin, and Fazel Famili. 2004. Hierarchical text categorization as a tool of associating genes with gene ontology codes. *Proceedings of the 2nd European Workshop on Data Mining and Text Mining in Bioinformatics*.

Aris Kosmopoulos, Ioannis Partalas, Eric Gaussier, Georgios Paliouras, and Ion Androutsopoulos. 2013. Evaluation measures for hierarchical classification: a unified view and novel approaches. *Data Mining and Knowledge Discovery*, 29.

Claudia Leacock and Martin Chodorow. 1998. Combining local context and wordnet similarity for word sense identification. *WordNet: An electronic lexical database*, 49(2):265–283.

Dekang Lin. 1998. An information-theoretic definition of similarity. In *Proceedings of the Fifteenth International Conference on Machine Learning (ICML 1998), Madison, Wisconsin, USA, July 24-27, 1998*, pages 296–304. Morgan Kaufmann.

Philip Resnik. 1999. Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language. *J. Artif. Intell. Res.*, 11:95–130.

Satoshi Sekine and Chikashi Nobata. 2004. Definition, dictionaries and tagger for extended named entity hierarchy. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, Lisbon, Portugal. European Language Resources Association (ELRA).

Aixin Sun and Ee-Peng Lim. 2001. Hierarchical text classification and evaluation. In *Proceedings of the 2001 IEEE International Conference on Data Mining, 29 November - 2 December 2001, San Jose, California, USA*, pages 521–528. IEEE Computer Society.

Grigorios Tsoumakas, Ioannis Katakis, and Ioannis P. Vlahavas. 2010. Mining multi-label data. In *Data Mining and Knowledge Discovery Handbook, 2nd ed*, pages 667–685. Springer.

Ke Wang, Senqiang Zhou, and Shiang Chen Liew. 1999. Building hierarchical classifiers using class proximity. In *VLDB'99, Proceedings of 25th International Conference on Very Large Data Bases, September 7-10, 1999, Edinburgh, Scotland, UK*, pages 363–374. Morgan Kaufmann.

Xi-Zhu Wu and Zhi-Hua Zhou. 2017. A unified view of multi-label performance measures. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 3780–3788. JMLR.org.

Zhibiao Wu and Martha S. Palmer. 1994. Verb semantics and lexical selection. In *32nd Annual Meeting of the Association for Computational Linguistics, 27-30 June 1994, New Mexico State University, Las Cruces, New Mexico, USA, Proceedings*, pages 133–138. Morgan Kaufmann Publishers / ACL.

Dongjin Yu, Dengwei Xu, Dongjing Wang, and Zhiyong Ni. 2019. Hierarchical topic modeling of twitter data for online analytical processing. *IEEE Access*, 7:12373–12385.

Min-Ling Zhang and Zhi-Hua Zhou. 2014. A review on multi-label learning algorithms. *IEEE Transactions on Knowledge and Data Engineering*, 26(8):1819–1837.

Yi Zhang, Samuel Burer, and W. Nick Street. 2006. Ensemble pruning via semi-definite programming. *Journal of Machine Learning Research*, 7:1315–1338.