IN SUPREMÆ DIGNITATIS

1343

UNIVERSITÀ DI PISA

Corso di Laurea in Informatica Umanistica

TESI DI LAUREA MAGISTRALE

From Classification to Quantification: New Methods for Estimating Cause-of-Death Prevalence from Verbal Autopsies

Relatori

Felice Dell'Orletta
Andrea Esuli
Alejandro Moreo
Fabrizio Sebastiani

Candidata

Ilaria Salogni

ANNO ACCADEMICO 2023/2024

Abstract

Verbal autopsies are standardized textual questionnaires that gather information about the symptoms experienced by the deceased in the days preceding a fatality. Verbal autopsies were developed to address the need for fundamental registration of deaths also for countries with weak registration systems. These documents pertain to the field of epidemiology, in which the aim is obtaining estimates of disease prevalence across different geographical regions, time periods, and age groups. In this context, the focus shifts from individual data to aggregate data, and the main goal is that of estimating (by means of supervised learning) the prevalence of each class accurately also when the distribution of the classes in the training data may be very different from the distribution of the unlabelled data that is the object of estimation. Therefore, epidemiological objectives, such as those just mentioned, align more with the task of quantification than that of classification, understood as minimizing errors while accurately assigning the correct pathology to each death.

This study focuses on examining methods for estimating prevalence in mortality statistics datasets, aiming to explore cause-of-death assignment from a novel perspective. We developed innovative text quantification techniques that leverage the dataset's intrinsic hierarchical nature.

The results obtained from the four classes of solutions we implemented diverged from our expectations.

Surprisingly, many of the supposedly more sophisticated quantification methods failed to outperform a simple Classify-and-Count approach. However, utilizing a hierarchical classification algorithm instead of a flat one yielded interesting outcomes, as it produced the same results with a shorter training time.

By testing hierarchical quantification algorithms in various settings, this work introduces the concept of hierarchical quantification for the first time. Nevertheless, our results show no significant improvement compared to algorithms that disregard the hierarchical nature of the labels.

Acknowledgments

We would like to express our gratitude to the organizations that provided the invaluable datasets for this research. Specifically, we thank Estelle McLean and Clara Calverts at The London School of Hygiene & Tropical Medicine for the Karonga HDSS dataset and for the invaluable support. We equally would like to thank Peter Rodriguez, Prabhat Jha, Richard Wen at University of Toronto for the Million-Death Study dataset and for their contributions, that made this study possible.

Contents

Chapter 1

Introduction

The global landscape exhibits significant disparities in the caliber and efficacy of national health information systems, closely linked to levels of economic and infrastructural advancement. Nations that presently manage streamlined and comprehensive health information systems, founded on exhaustive individual data, often started from fundamental registration of fatalities and their origins (Byass et al., 2019). *Verbal Autopsies* (VAs) (King and Lu, 2008) are indeed standardised textual questionnaires that gather information about the condition and symptoms experienced by the deceased in the days preceding a fatality, developed to address the need of reliable mortality statistics also for countries with weak death registration systems. As a practical illustration, verbal autopsies have been employed in the recent past to assess mortality attributed to HIV/AIDS and malaria (Deressa et al., 2007) in Ethiopia's rural areas, as well as to evaluate the effectiveness of anti-retroviral therapy (ART) interventions (Araya et al., 2011).

The purpose of this tool is to facilitate the assignment of the most probable cause of death to each document, based on the informations collected during an interview with caregivers, and thus obtain reliable statistics, intended as mortality rates within a specific population over a defined period. Since the assignable causes of death are a finite and predefined set, this inference is a classification task, which in previous works have been automated using ad hoc algorithms. However, epidemiological objectives such as those just mentioned, align more with the task of *quantification* than classification, understood as the process of estimating the prevalence of each class in a dataset (Esuli et al., 2023), also when classes in the training data may be very different from the distribution in the future, unlabelled data.

Quantification algorithms can also be used to predict the prevalence distribution of data with an intrinsic hierarchical nature, where data are organised in a tree-like structure. There is a range of diverse applications in which this peculiar structure can be leveraged (Silla and Freitas, 2010), including the hierarchical taxonomy of the International Statistical Classification of Diseases (ICD) classification framework (Steindel, 2010).

This work, which centers on examining methods for estimating prevalence in mortality statistics datasets, aims to explore *cause of death* (CoD) assignment from a new perspective, developing novel methodologies that leverage both flat and hierarchical quantification techniques.

Within the realm of machine learning, this scenario embodies a supervised learning task, where the definitive cause of death, determined by a group of trained physicians during dataset creation, acts as the ground truth label.

Training estimators for class prevalence via supervised learning is called quantification, hence we will refer to the set objective as flat and hierarchical quantification. While computational approaches to assigning causes of death have been in existence for several years, the ones adopted in this thesis are innovative, as our objective is not to achieve accurate classification but rather accurate quantification.

To this goal, we will leverage the textual part of the VAs dataset, that has been not employed in the computer algorithms specifically designed for VAs, proving that this information can be very useful to the final cause-of-death assignment (Blanco et al., 2021).

1.1 Subject and Outline of this Thesis

In this thesis, we tackle the problem of obtaining accurate estimators of class prevalence values via machine learning in a verbal autopsies dataset, leveraging the hierarchical structure of the cause-of-death codeframe. We propose three research questions to guide this work.

The first one (i) is whether exploiting algorithms for direct estimation of the distribution of causes of death leads to a benefit compared to following a Classify and Count approach. As previously mentioned, our objective is indeed not to achieve accurate classification but rather accurate quantification, in a focus in line with that of epidemiology. This work is among the first ones to apply quantification specifically to verbal autopsies.

The second key idea (ii) is that in order to solve a hierarchical quantification application, it may be useful to have quantification algorithms that are themselves hierarchical in nature. With this purpose, we present experiments in which we test the hierarchical quantification algorithms against baseline methods on data generated by a robust experimental protocol. As the third and last research question (iii), we propose to assess whether hierarchical quantification may lead to an improvement in execution time, when compared to flat quantification.

This marks the initial contribution to hierarchical quantification, which involves conducting quantification while leveraging the branched structure of data, through a hierarchically implemented algorithm.

The experimentation has yielded negative results, revealing that, although it is often more advantageous to follow a *divide et impera* approach (i.e., to break down a large multiclass quantification problem into several smaller multiclass quantification problems), flat quantification techniques perform well also when the set of classes is structured as a hierarchy.

In order to ensure the reproducibility of our experiments, we make available the code[1] and the data (when possible) on which they are based. The rest of the thesis is structured as follows. We first discuss preliminaries and the problem setting in Chapter 2. This discussion is followed, in Chapter 3, by a thorough presentation of previous works that address the problem of predicting causes of death in different settings. We then move to describing our proposed methods in Chapter 4, and the results of our experiments in Chapter 5. In Chapter 7 we will outline some conclusions about our work and outline possible avenues for future research.

[1] https://github.com/IlariaSalogni/Quantification_VAs/

Chapter 2

Preliminaries

2.1 Notation

In this work we use the following notation. By $\mathbf{x}$ we indicate a datapoint drawn from a domain $\mathcal{X}$ of datapoints, by y we indicate a class belonging to a finite, predefined set of classes (also known as a *codeframe*) $\mathcal{Y}$, while by $(\mathbf{x}, y)$ we denote a datapoint with its true class label. By L we denote a collection of labelled datapoints, that we typically use for training our model, while by U we denote a collection of unlabelled datapoints (i.e., datapoints whose label is unknown to our model), that we typically use for testing purposes.

We define a *hierarchical codeframe* as a tree-shaped codeframe $\mathcal{Y}_0 = \{y_0, y_1, ..., y_n, y_{n+1}, ..., y_{n+m}\}$, where y_0 is the "root" class, $y_1, ..., y_n$ are the "leaf" classes, and $y_{n+1}, ..., y_{n+m}$ are the "internal node" classes.[1]

We will take $\uparrow(y_i)$ to denote the parent class of y_i, $\downarrow(y_i)$ to denote the set of children classes of y_i, $\Uparrow(y_i)$ to denote the set of ancestor classes of y_i, $\Downarrow(y_i)$ to denote the set of descendant classes of y_i, and $\leftrightarrow(y_i)$ to denote the set of sibling classes of y_i. We restrict our attention to single-label problems, i.e., to scenarios in which each datapoint $\mathbf{x}$ belongs to exactly one leaf class in $\{y_1, ..., y_n\}$.

When $\mathbf{x}$ belongs to a leaf class y_i, we consider it to also belong to all of its ancestor classes in $\Uparrow(y_i)$. (Classes corresponding to internal nodes are thus seen as mere aggregations of leaf classes.) A training set will thus be a set $L = L_1 \cup ... \cup L_n$, where L_i indicates the set of training examples of leaf class y_i, while a test set will be a set $U = U_1 \cup ... \cup U_n$, with the analogous meaning.[2]

[1] The "0" index in $\mathcal{Y}_0$ is there to signify that the tree has class y_0 as its root. This will come in handy when illustrating recursive algorithms, where by $\mathcal{Y}_i$ we will usually indicate a codeframe with class y_i as its root and corresponding to a subtree of $\mathcal{Y}_0$.

[2] Note that the fact that a datapoint belongs to a leaf node and, at the same time, to all of its ancestor nodes, does not make the problem a multi-label one. To us, a multi-label hierarchical classification problem is one where a datapoint can belong to zero, one, or several *leaf* classes at the same time.

By symbol σ we denote a *sample*, i.e., a non-empty set of (labelled or unlabelled) datapoints drawn from $\mathcal{X}$. By $p_\sigma(y)$ we indicate the *true* prevalence of class y in sample σ, by $\hat{p}_\sigma(y)$ we indicate an *estimate* of this prevalence, and by $\hat{p}^q_\sigma(y)$ we indicate the estimate of this prevalence obtained by means of quantification method q. What we have said above concerning hierarchical codeframes entails that the prevalence of an internal node class is the sum of the prevalence values of its descendant leaf classes; in particular, the prevalence of the root class is 1.

From what we have said above it also derives that vector $\mathbf{p} = (p_1, \ldots, p_n)$, where we shorten $p_\sigma(y_i)$ as p_i, represents a distribution; the goal of a hierarchical quantifier is indeed to estimate this distribution, i.e., to infer an estimated distribution $\hat{\mathbf{p}} = (\hat{p}_1, \ldots, \hat{p}_n)$. By $d(\mathbf{p}, \hat{\mathbf{p}})$ we denote an evaluation measure for SLQ; these measures are typically *divergences*, i.e., functions that measure the amount of discrepancy between two probability distributions.

We define a *single-label hard classifier* for a codeframe $\mathcal{Y}$ as a function $h : \mathcal{X} \rightarrow \mathcal{Y}$, i.e., a predictor of the class attributed to a datapoint. We instead take a *probabilistic classifier* for a codeframe $\mathcal{Y}$ to be a function $s : \mathcal{X} \rightarrow \Delta^{n-1}$, with Δ^{n-1} the unit (n-1)-simplex (aka *probability simplex* or *standard simplex*) defined as $\Delta^{n-1} = \{(p_1, \ldots, p_n) \mid p_i \in [0, 1], \sum_{i=1}^n p_i = 1\}$, i.e., as the domain of all vectors representing probability distributions over $\mathcal{Y}$. We use the notation $\Pr(y_i|\mathbf{x})$ to denote a *posterior probability*, i.e., the probability that classifier s attributes to the fact that datapoint $\mathbf{x}$ belongs to class y_i; our probabilistic classifiers indeed map datapoints $\mathbf{x}$ into vectors $(\Pr(y_1|\mathbf{x}), ..., \Pr(y_n|\mathbf{x}))$, where $\sum_{i=1}^n \Pr(y_i|\mathbf{x}) = 1$. We will typically assume that our probabilistic classifiers are *calibrated.*

We define a *single-label quantifier* as a function $q : 2^{\mathcal{X}} \rightarrow \Delta^{n-1}$, i.e., a function mapping samples drawn from $\mathcal{X}$ into probability distributions over $y_1, ..., y_n$. Note that, despite the fact that the codomains of probabilistic classifiers and quantifiers are the same, in the former case the i-th component of $s(\mathbf{x})$ denotes the posterior probability $\Pr(y_i|\mathbf{x})$, i.e., the probability that $\mathbf{x}$ belongs to class y_i as estimated by s, while in the latter case it denotes the class prevalence value $p_\sigma(y_i)$ as estimated by q.

2.2 Causes of Death and Verbal Autopsies

2.2.1 What Verbal Autopsies are

In 2015, the World Health Organization (WHO) estimated that approximately 50% of the 56 million global deaths lacked registered cause information (Nichols et al., 2018). In other terms, more than a half of the yearly fatalities in lower and middle-income nations take place outside of the healthcare system, and due to inadequate death registration systems, this does not make reliable mortality statistics available.

Furthermore, the practice of gross pathology autopsy is neither practicable nor accepted in numerous developing nations (Gemechu et al., 2009), further complicating the process of gathering reliable data, and the available information often exhibit bias as they predominantly stem from hospital statistics in non-rural environments.

Consequently, the available data frequently fail to accurately represent the broader experiences of the general population, and may lead to a distorted understanding of health priorities within the country. The issue is especially concerning at present, when highly reactive epidemiological tools are needed more than ever.

Utilizing validated verbal autopsy procedures in mortality surveillance systems and *demographic surveillance sites* (DSSs) is being proposed as a cost-effective alternative method for determining causes of death and establishing sustainable medium-term solutions (Misganaw et al., 2012). For instance, the Million-Death Study (Ke et al., 2021), an ongoing research initiative conducted in India, tracked nearly 14 million individuals from 1998 to 2014. As reported by its administrators, its findings have already impacted priority setting and altered global estimates of disease burdens, for example modelling malaria mortality (Jana et al., 2022) and diseases associated with particulate matter (Brown et al., 2022).

Although the instrument first originated in 1956 within the Narangwal Project in India (Biraud, 1956), the WHO is the organism that first developed standards and attempted to make this practice more organic and systematised (Bailo et al., 2022), and is engaged in ongoing revisions of these protocols. The latest iteration of these protocols is embodied in the WHO 2016 verbal autopsy instrument (Nichols et al., 2018)

In concrete terms, the basic principle of VAs is to ask standardised questions to a respondent who was with the deceased during the final illness, and that can recall symptoms and chronological sequence of events that occurred around that period, while the interview is conducted. For instance, a query could inquire about the deceased person having had a fever in the days preceding the death, or whether the deceased individual smoked. Households for conducting interviews can be identified either through referrals from the health district or through burial surveillance (Misganaw et al., 2012). This process is based on the assumption that most of the death causes can be recognised accurately by these reported signs, and that there is at least one main cause of death, that can be identified as the most likely via processing of the collected data.

Conducting the interviews are usually field workers, non-health professionals who undergo specific training and are locally recruited to ensure a shared cultural background with the community, but in a few studies also nurses or doctors were employed (Mahesh et al., 2022).

These field workers schedule interviews with sensitivity to the families' mourning period (Mahesh et al., 2022), and carry out interviews in the local language, back-translating the form into English at a later time. The most widespread tools for VAs (Mahesh et al., 2022) are the World Health Organization (Nichols et al., 2018) questionnaires, followed by Population Health Metrics Research Consortium (PHMRC) (Murray et al., 2011a) questionnaire, but many researches use a form adapted from a standardised version specifically for the purposes of the search, usually based on the just mentioned most common ones. The questionnaire can also be (and usually is) further modified to align with the local culture and overall context. For instance, in a study conducted in specific regions of India, the question regarding cigarette smoking was restated to determine if the deceased had smoked *bidi*, a locally used tobacco product (Wu et al., 2013). Also as a consequence of this, a number of questionnaires with overlapping but not identical question can be found. Recently, there have been studies (Blanco et al., 2021) regarding the impact of the collection of data directly on electronic devices, instead of paper questionnaires. Subsequently, the information from the verbal autopsy questionnaires is used by a panel of hospital-assigned reviewers (that are required to be specialised healthcare personnel, unlike the previous phase) to independently assign up to two causes of death. Agreement by at least two clinical officers is required for a diagnosis to be accepted as the underlying cause of death. If consensus can not be reached, the cause of death is considered undefined.

Finally, a research assistant with a health background assigns cause-of-death codes based on the international classification of diseases or a study-specific coding system. As described in Section 3.1, the computational methods to assign cause of death, avoiding the need for a trained physician to manually review each document, have been object of study for at least two decades now.
Performance varies between physicians and computer-coded algorithms across different settings (Bailo et al., 2022; Tunga et al., 2021): CCVA methods are faster and consistent, and excels in reliably established, properly documented hospital settings, but as models trained in-hospital data may not replicate well in community deaths, while PCVA remains reliable due to uncertainties in CCVA accuracy (De Souza et al., 2020) and effective for community CoD that rely on caretaker observations. A number of studies (Mahesh et al., 2022) relies on both methods.

2.2.2 Cause-of-Death Codeframes

The International Classification of Diseases (ICD), regularly updated by the WHO, stands as the predominant system for globally coding causes of death; however, numerous verbal autopsy studies either utilise a personalised version of ICD or develop entirely bespoke codeframes (Glynn et al., 2014). Like most of the other codeframes for VAs, often derived from it, the ICD employs a three-levels hierarchical structure with sections dedicated to general conditions, each contain-

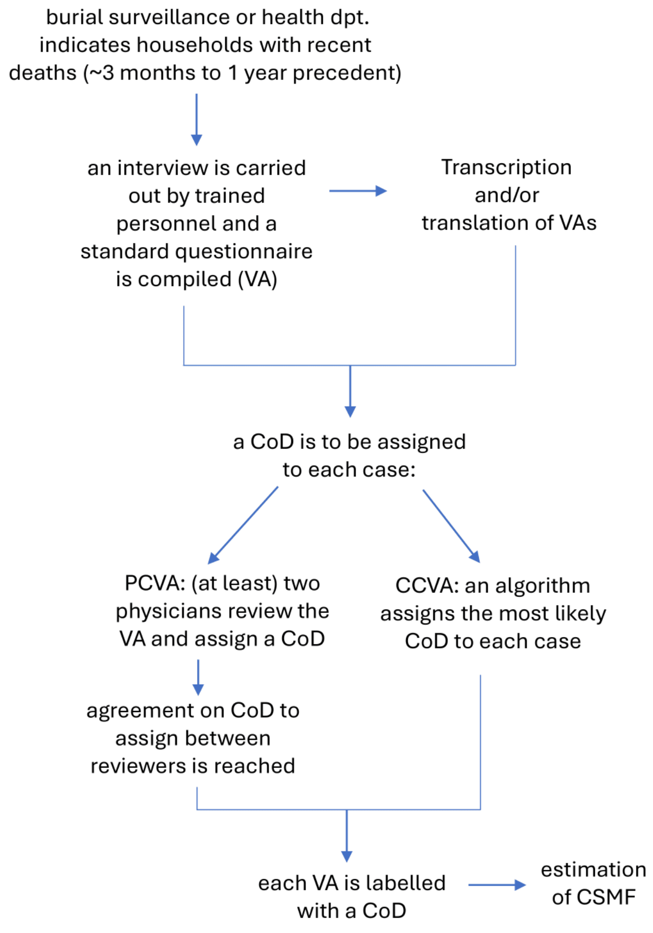

burial surveillance or health dpt. indicates households with recent deaths (~3 months to 1 year precedent)
an interview is carried out by trained personnel and a standard questionnaire is compiled (VA)
Transcription and/or translation of VAs
a CoD is to be assigned to each case:
PCVA: (at least) two physicians review the VA and assign a CoD
CCVA: an algorithm assigns the most likely CoD to each case
agreement on CoD to assign between reviewers is reached
each VA is labelled with a CoD
estimation of CSMF

Figure 2.1: Steps in the cause-of-death assignment process for verbal autopsies.

ing a subtree that outlines more specific diseases. This classification hierarchy serves the purpose of aggregation rather than defining inheritance, making the hierarchical relationship more aligned with (member-of) rather than the conventional (parent–child) structure (Steindel, 2010). Usually the first level is indicated by a letter, and the others by numbers from 0 to 9.

2.2.3 Validity of the Verbal Autopsy Instrument

Despite this may seem to be in contrast with the previously discussed notion that VAs are beneficial in contexts characterised by a significant number of deaths outside hospital settings, assessing the performance of the method—meaning the establishment of a usable reference standard—requires a full diagnostic autopsy

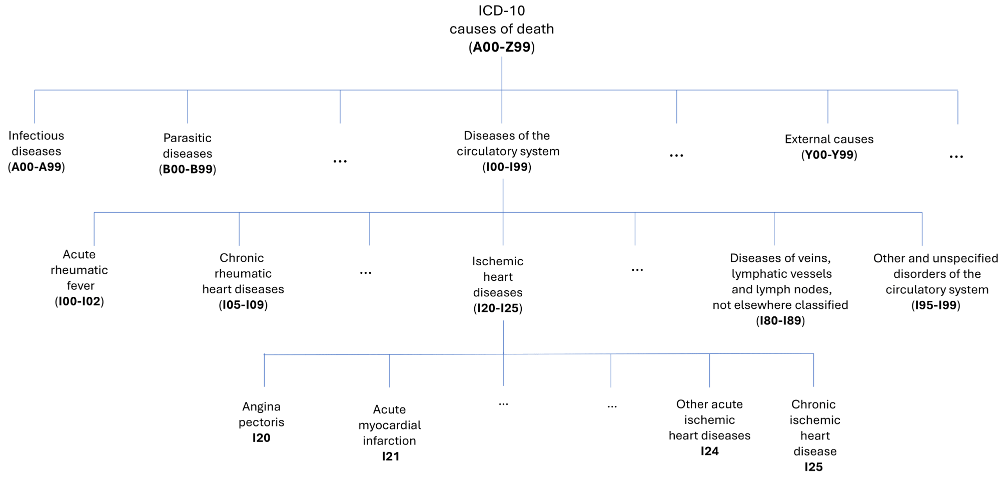


Figure 2.2: A segment of the hierarchical structure of the ICD-10 codeframe, illustrating the nested classification of diseases. Each node represents a category, subcategory, or specific condition, showcasing the detailed organization used to systematically classify medical diagnoses.

and medically-assigned cause of death from deaths occurring in hospital settings (Tunga et al., 2021).

Numerous interventions involve the validation process of an algorithm against clinician interpretation of verbal autopsy data, including this thesis. However, many assessments of the VA method do not prioritise the accuracy of the diagnosis of the reference standard (Leitao et al., 2014; Mahesh et al., 2022). While it may be easily identifiable, for example, in the case of external causes like injuries, other instances may require diagnostic imaging or laboratory findings. Very few (Bailo et al., 2022) instances exist in which a verbal autopsy validation process has been conducted using a full diagnostic autopsy as the gold standard (Hart et al., 2022; Menéndez et al., 2021). Nonetheless, it would surpass the scope of this thesis to inquire into how closely our reference aligns with the cause of death that could have been determined through the use of medical imaging or a complete autopsy.

Additionally, the distribution of classes in the training set may significantly differ from that in the unlabelled data, leading to distribution shift (Esuli et al., 2023) and making generalization considerably complex. To give a practical example, the fact that some Causes of Death are much rarer than others poses a challenge: AIDS is not only ubiquitous among datasets, but also causes an extremely higher number of fatalities than some specific type of cancer.

Besides that, the fact that certain causes of death, such as maternal and perinatal mortality diseases, are distributed only within specific age groups or genders of the deceased underscores the need for datasets tailored to investigate specific conditions.

Furthermore, datasets may be created by merging data from different geographical areas (an extensive comparison is given in Mahesh et al. (2022)) which, however, may contain local environmental conditions (such as the prevalence of malaria, or the presence of conditions linked to specific industries adjacency). This implies that, even if we undertake the considerable effort to create a VAs dataset by collecting data in areas of India and Tanzania, we might not be able to reuse the trained tool effectively when applying it to data from Brazil, for example.

In addition to the absence of standardised datasets (Tunga et al., 2021), several other elements can impact the validity and reliability of verbal autopsies. Factors such as the quality and standardization of the verbal autopsy tool (including the questionnaire, diagnostic procedures, and mortality classification) and the data collection process play significant roles (Misganaw et al., 2012); (Chandramohan et al., 1998). It must be noted that there is a notable degree of variability between studies concerning field procedures, questionnaires utilised, causes of death assessed, respondent recall, and performance metrics, among other considerations (Leitao et al., 2014), and that this poses a limit to this instrument.

However, the occurrence of misclassification does not inherently indicate inaccuracies in the verbal autopsy's estimation of *cause-specific mortality fractions* (CSMF) (Anker, 1997) and, as already said, the aim of this thesis is to obtain a good quantifier and not necessarily a good classifier. In conclusion, in settings with limited resources, the tool still demonstrates substantial potential in health and demographic surveillance (Van Eijk et al., 2008).

2.3 Classification

Classification involves a supervised machine learning process in which a model endeavours to anticipate the accurate label of an input data point. This method relies on training the model on a dataset containing known labels for a collection of data points, in our case the cause of death assigned by a trained physician to each death report. As we already said, causes of death are organised in a standardised hierarchical taxonomy, the most common being ICD, and the information coming from this structure can potentially be leveraged or disregarded.

2.3.1 Flat Classification

The straightforward flat classification approach, considered the most basic for addressing hierarchical classification problems, involves entirely disregarding the class hierarchy. This method predicts only the classes located at the leaf nodes, just like a traditional classification algorithm during both training and testing phases, but it indirectly resolves the hierarchical classification issue by implicitly assigning all ancestor classes to an instance when a leaf class is assigned. However, this simplicity comes at a significant drawback – the need to construct a classifier for discriminating among a multitude of classes (all leaf classes), and giving up on exploiting information about parent-child class relationships. A flat classifier can handle both tree and DAG taxonomies, but it is incapable of handling non-mandatory leaf node prediction problems.

2.3.2 Hierarchical Classification

Hierarchical classification can be viewed as a specific form of structured classification problem, wherein the classification algorithm's output is determined within a class taxonomy. From this perspective, hierarchical classification can be regarded as a subset of structured classification (Astikainen et al., 2008; Seeger, 2008), which is, in any case, a broader category encompassing classification problems characterised by some form of structure among the classes, whether hierarchical or not. The literature addressing hierarchical classification problems is often dispersed among various application domains that lack strong interconnections, in which also VAs are included.

Silla and Freitas (2010) distinguish hierarchical classification problems where datapoints can only belong to leaf nodes (a scenario they call *mandatory leaf node prediction*), and problems where a datapoint can instead belong to an internal node and to none of its descendants. However, Cheng et al. (2001) argue that this distinction is inessential, showing that it is always possible to map a problem of the latter type into one of the former type;[3] in this thesis we will thus assume that datapoints can only belong to leaf nodes.

Most HC methods follow a *divide et impera* approach, i.e., use the hierarchical structure to decompose the classification problem into a set of smaller such problems, one for each node in the hierarchy. Classifiers local to each node are trained, often by using a "local" subset of the entire training set, and are used at classification time to route the unlabelled datapoint through a specific root-to-leaf path. This *divide et impera* approach goes back at least to (Koller and Sahami, 1997), and has been the dominant approach ever since.

[3] Without loss of generality, given a hierarchical codeframe $\mathcal{Y}_0$ in which internal nodes can indeed have their "own" datapoints, we can map $\mathcal{Y}_0$ into an "extended" hierarchical codeframe $\mathcal{Y}_0'$ by adding to every internal node y_i of $\mathcal{Y}_0$ an additional child (leaf) node y_i', and by moving into y_i' all datapoints that belonged to y_i but to none of its descendant classes. This simple mapping, indeed proposed by Cheng et al. (2001), produces a hierarchy $\mathcal{Y}_0'$ semantically equivalent to $\mathcal{Y}_0$ in which all datapoints are contained in leaf classes only. Note that many real-world classification schemes (e.g. the ACM Classification Scheme) are of this latter type, since their internal nodes often have a special child leaf class (called General, or Other) which contains all datapoints belonging to the node but to none of its descendants.

A fundamental difference among HC problems is that between single-label HC problems and multi-label HC problems. In *single-label HC problems* each datapoint belongs to exactly one leaf. To solve these problems, a typical approach (and one that we will follow in this thesis) is that of associating a single-label classifier to each nonleaf node y_i, where the classes addressed by this classifier are the children of y_i; the classifier thus acts as a "router", i.e., it receives a datapoint from node $\uparrow(y_i)$ and routes it to exactly one of y_i's children, and to its associated subtree. The process of training these classifiers is often sensitive of the presence of a hierarchical structure; for instance, feature selection is usually carried out locally to each node rather than globally to the entire tree (Koller and Sahami, 1997). The first authors to propose this approach were Koller and Sahami (1997), and many followed suit. In *multi-label HC problems* each datapoint may instead belong to zero, one, or several leaves. Here, a typical approach is that of associating a binary classifier to each nonroot node y_i; this classifier acts as a filter, i.e., it receives a datapoint $\mathbf{x}$ from $\uparrow y_i$ (the parent node of y_i) and decides whether $\mathbf{x}$ belongs to the subtree associated to y_i (in which case $\mathbf{x}$ is routed to all the children nodes of y_i) or not (in which case $\mathbf{x}$ is blocked from proceeding further down the tree). Example works in this direction are Esuli et al. (2008); Ruiz and Srinivasan (2002).

2.4 Quantification

The process of training class prevalence estimators through supervised learning is termed quantification, as coined by Forman (2005) or as "learning to quantify".

In literature, two main categories of quantification methods are recognised: aggregative methods, which involve classifying individual data items as an intermediate step, and non-aggregative methods, that address the quantification problem in its entirety, bypassing the need to classify individual items. In this section we will start from the more obvious method, *Classify and Count*, then moving to direct methods for getting the aggregate level, that are expected to perform better in prior probability shift settings (Moreo et al., 2023).

2.4.1 Classify and Count

Classify and Count (**CC**), already hinted at in the introduction, is the naïve quantification method, and the one baseline that all genuine quantification methods are supposed to beat. Given a hard classifier h and a sample σ, CC is formally defined as

$$\hat{p}_{\sigma}^{\mathrm{CC}}(y_i) = \frac{|\{\mathbf{x} \in \sigma | h(\mathbf{x}) = y_i\}|}{|\sigma|} \tag{2.1}$$

In other words, the prevalence of a class y_i is estimated by classifying all unlabelled datapoints, counting the number of datapoints that have been assigned to

y_i, and dividing the result by the total number of datapoints. It may yield poor results due to prior probability shift, impacting its effectiveness (Forman, 2006).

2.4.2 Adjusted Classify and Count

The *Adjusted Classify and Count* (**ACC**) method (see (Fernandes Vaz et al., 2019; Forman, 2008)) attempts to correct the estimates returned by CC by relying on the law of total probability, according to which

$$p(h(\mathbf{x}) = y_i) = \sum_{y_j \in \mathcal{Y}} p(h(\mathbf{x}) = y_i \mid y_j) \cdot p(y_j) \tag{2.2}$$

which can be more conveniently rewritten using matrix notation as

$$\mathbf{p}_\sigma^{\mathrm{CC}} = \mathbf{M}_h \cdot \mathbf{p}_\sigma^{\mathrm{ACC}} \tag{2.3}$$

where $\mathbf{p}_\sigma^{\mathrm{CC}}$ is the vector representing the distribution across $\mathcal{Y}$ of the datapoints as estimated via CC, and matrix $\mathbf{M}_h$ contains the misclassification rates of h, i.e., m_{ij} is the probability that h will assign class y_i to a datapoint whose true label is y_j. Matrix $\mathbf{M}_h$ is unknown, but can be estimated via k-fold cross-validation, or on a validation set. Vector $\mathbf{p}_\sigma^{\mathrm{ACC}}$ is the true distribution; it is unknown, and the ACC method consists of estimating it by solving the system of linear equations of Equation 2.3 (see Bunse (2022) for more on the multiclass version of ACC).

2.4.3 Probabilistic Classify and Count

While CC and ACC rely on the integer counts returned by a hard classifier h, it is possible to define variants of them that use instead the expected counts computed from the posterior probabilities returned by a calibrated probabilistic classifier s. This is the core idea behind *Probabilistic Classify and Count* (**PCC**) and *Probabilistic Adjusted Classify and Count* (**PACC**) (Bella et al., 2010). PCC is defined as

$$\begin{aligned}\hat{p}_\sigma^{\mathrm{PCC}}(y_i) &= \frac{1}{|\sigma|}\sum_{\mathbf{x}\in\sigma}[s(\mathbf{x})]_i \\ &= \frac{1}{|\sigma|}\sum_{\mathbf{x}\in\sigma}\Pr(y_i|\mathbf{x})\end{aligned} \tag{2.4}$$

while PACC is defined as

$$\mathbf{p}_\sigma^{\mathrm{PCC}} = \mathbf{M}_s \cdot \mathbf{p}_\sigma^{\mathrm{PACC}} \tag{2.5}$$

Equation 2.5 is identical to Equation 2.3, but for the fact that the leftmost part is replaced by the prevalence values estimated via PCC, and for the fact that the misclassification rates of the soft classifier s (i.e., the rates computed as expectations using the posterior probabilities) are used.

Methods CC, ACC, PCC, PACC, are sometimes collectively referred to as the "CC variants", and are all (as it is easy to verify) aggregative quantification methods. Although more sophisticated quantification systems have been proposed in the literature, the CC variants have recently been found to be competitive contenders when properly optimised (Moreo and Sebastiani, 2021). This, along with their simplicity, has motivated us to focus on the CC variants as a first step towards devising hierarchical quantifiers.

2.4.4 The SLD Quantification Algorithm

A further, very popular (aggregative) quantification method is the one proposed by Saerens et al. (2002), which is often called **SLD** from the names of its proposers, and which was called EMQ in Gao and Sebastiani (2016). SLD was the best performer in a recent data challenge centred on quantification (Esuli et al., 2022a), and consists of training a probabilistic classifier and then using the EM algorithm

1. to re-estimate the class prevalence values of sample σ, according to the update rule

$$\hat{p}_i^{(s)} \leftarrow \frac{1}{|\sigma|}\sum_{\mathbf{x}\in\sigma}\Pr{}^{(s-1)}(y_i|\mathbf{x}) \tag{2.6}$$

   where the (s) superscript indicates the s-th iteration;

2. to update the posterior probabilities of the unlabelled datapoints $\mathbf{x}$ that the classifier returns, according to the update rule

$$\Pr{}^{(s)}(y_j|\mathbf{x}) \leftarrow \frac{\dfrac{\hat{p}_i^{(s)}}{\hat{p}_i^{(0)}}\cdot\Pr{}^{(0)}(y_j|\mathbf{x})}{\displaystyle\sum_{y_i\in\mathcal{Y}}\dfrac{\hat{p}_i^{(s)}}{\hat{p}_i^{(0)}}\cdot\Pr{}^{(0)}(y_i|\mathbf{x})} \tag{2.7}$$

Steps 1 and 2 are carried out in an iterative, mutually recursive way, until mutual consistency, defined as the situation in which

$$\hat{p}_i \approx \frac{1}{|\sigma|}\sum_{\mathbf{x}\in\sigma}\Pr(y_i|\mathbf{x}) \tag{2.8}$$

is achieved for all $y_i \in \mathcal{Y}$.

2.4.5 The ReadMe Quantification Algorithm

It is a text quantification method, proposed by King and Lu (2008) specifically for verbal autopsies and later renamed *ReadMe* in (King et al., 2010). It is based on the idea of estimating class prevalence values directly via equation

$$p(\mathbf{x}_i) = \sum_{y_j \in \mathcal{Y}} p(i \mid y_j) p(y_j) \tag{2.9}$$

where $p(\mathbf{x}_i)$ represents the probability that a document drawn randomly from U has $\mathbf{x}_i$ as its vectorial representation. *ReadMe2* has been proposed by Jerzak et al. (2023) aiming at improving the performance of the original system, moving away from the sparse representation of the feature space and the subsampling procedure in favour of a dense representation based on word embeddings (Esuli et al., 2023).

Chapter 3

Related Work

In the following sections, we disclose some of the main works of the past years that concerned verbal autopsies, focusing first on classification algorithms proposed in the past years for cause-of-death assignment.

Quantification has been studied extensively in some scenarios, like the binary setting (when the codeframe only contains two classes), the single-label multiclass setting (when the set of classes are three or more and mutually exclusive) and ordinal (when the classes are three or more and in a total order), or more marginally in other settings, like the multi-label setting (each example can be assigned one, several or no classes) (Moreo et al., 2023). However, the hierarchical setting, although widely explored for classification (Silla and Freitas, 2010), has never been considered for quantification. Besides that, it is worth noting that quantification has rarely been performed over a large number of classes, with the widest (in a single-label, multiclass setting) being 28 classes, experimented in Esuli et al. (2022b).

3.1 Classification of Verbal Autopsies

3.1.1 Computerised Coding of Verbal Autopsies (CCVA)

As discussed earlier, CoD assignment is traditionally done by physician and traditionally the stages described above constituted the pipeline for CoD assignment. PCVA specifically refers to the process where human reviewers, usually two, certify the cause of death of a verbal autopsy. In case of disagreement, a third physician decides on CoD, in a slow, expensive and not always consistent approach (Byass et al., 2019).

However, there is also a relatively long history of the instruments computerised coding of verbal autopsy (CCVA) models to interpret the data and generate a likely underlying cause of death. For example, the inception of InterVA models dates back to Byass et al. (2003), undergoing several iterations since, but Boulle et al. (2001) already used neural networks for cause-of-death classification.

The principal CCVA methods (Bailo et al., 2022; Leitao et al., 2014) are: InterVAs, Tariff/SmartVA, InSilicoVA, King-Lu, Simplified Symptom Pattern (SSP), and Random Forest. Bayes rule is the most used logic (Bailo et al., 2022) in the CoD prediction, with a number of models, i.e., InterVA-4, InsilicoVA, King-Lu, and SSP relying on the Bayes rule to calculate the probability of a set of CoD on presence of circumstances, signs and symptoms.

Next, we will briefly illustrate the methodologies, further elaborated in Tunga et al. (2021) and in Li et al. (2022). We will consider $\mathcal{X}$ deaths, each with $\mathcal{F}$ binary indicators of symptoms. Let f_{ij} denote the indicator for the presence of j-th symptom in the i-th death $\mathbf{x_i}$, which can take values 0, 1, or NA (for missing data). We consider a pre-defined set of causes of size $\mathcal{Y}$. For the i-th death $\mathbf{x}_i$, denote the CoD by $y_i \in \{1...\mathcal{Y}\}$ and the probability of dying from cause k is denoted by P_{ik}. For the population, the CSMF of cause k is denoted as p_k, with $\sum_{k=1}^{\mathcal{Y}} p_k = 1$.

1. **InterVA**. InterVA-4 (Byass et al., 2012) and InterVA5 (Byass et al., 2019), designed to correspond to the WHO-2016 standard (Pham et al., 2023), calculate the chance for each cause of death using the formula:

$$P_{ik} = \frac{p_k^{(0)} \prod_{j=1}^{F} P(f_{ij} = 1 | y_i = k) 1_{f_{ij}=1}}{\sum_{k'=1}^{\mathcal{Y}} p_{k'}^{(0)} \prod_{j=1}^{F} P(f_{ij} = 1 | y_i = k') 1_{f_{ij}=1}} \tag{3.1}$$

   where both the prior distribution of each of the causes, $p_k^{(0)}$ and the conditional probabilities $P(f_{ij}{=}1|y_i{=}k)$ are fixed values provided in the InterVA software, as assigned by physicians considering local context (Menéndez et al., 2021). The formula does not follow the standard Bayes' rule as it omits the probability that any symptom is absent (McCormick et al., 2016). Once the individual cause of death distributions are computed, InterVA-4 employs a set of predefined rules to identify the top three most probable CoD assignments. It then sets the probabilities for the remaining CoDs to zero and includes an "undetermined" category to ensure the probabilities sum up to 1 (Li et al., 2022). Then the population-level CSMFs are calculated as the aggregation of individual CoD distributions, such that

$$p_k = \sum_{i=1}^{\mathcal{X}} P_{ik'}^{*} \tag{3.2}$$

   where $P*_{ik}$ denotes the individual CoD distribution after introducing the undetermined category.

2. **Naive Bayes Classifier (NBC)**. The application to VAs of the NBC algorithm was proposed by Miasnikof et al. (2015). It shares similarities

with the InterVA algorithm but, unlike InterVA, also incorporates symptoms that are absent, and calculates conditional probabilities of symptoms given causes based on training data rather than relying on physician given input.

$$p_{ik} = \frac{p_k^{(0)} \prod_{j=1}^{F} P(f_{ij} = 1 | y_i = k) 1_{f_{ij}=1} + P(f_{ij} \neq 1 | y_i = k) 1_{f_{ij} \neq 1}}{\sum_{k'=1}^{\mathcal{Y}} p_{k'}^{(0)} \prod_{j=1}^{F} P(f_{ij} = 1 | y_i = k') 1_{f_{ij}=1} + P(f_{ij} \neq 1 | y_i = k) 1_{f_{ij} \neq 1}}$$

3. **InSilicoVA**. InSilicoVA (McCormick et al., 2016) is an extension of InterVA, that estimates both individual cause-assignment probabilities and CSMF at the population level. InSilicoVA is the only algorithm (Li et al., 2022) that distinguishes between negative and unknown responses in verbal autopsy data. Similar to InterVA, InSilicoVA utilises the same probabilistic framework to link indicators with causes, employing identical interview items and a matrix of conditional probabilities (Byass et al., 2019).

4. **Tariff/SmartVA**. The Tariff method was first proposed in 2011 by James et al. (2011), and subsequently refined and renamed as SmartVA (for Health Metrics and application., 2024). The idea at the base of the method is to identify signs or symptoms in a VA questionnaire that are highly indicative of a particular CoD.

   Differing from the previous three methods, this approach does not calculate an explicit probability distribution of the cause of death for each death, but instead, it employs a weighted sum system known as a "tariff", which aggregates scores from various indicators such as symptoms or risk factors for each death.

   Specifically, a tariff score is computed for each CoD k so that

$$\text{Score}_{ik} = \sum_{j=1}^{F} \tau_{kj} 1_{f_{ij}=1} \tag{3.3}$$

   where the symptom-specific tariff score τ_{kj} is defined as

$$\tau_k = \frac{\mathbf{x}_{kj} - \text{med}(\mathbf{x}_{1j}, \mathbf{x}_{2j}, ..., \mathbf{x}\mathcal{Y}j)}{\text{IQR}(\mathbf{x}_{1j}, \mathbf{x}_{2j}, ..., \mathbf{x}\mathcal{Y}j)} \tag{3.4}$$

   where $\mathbf{x}_{kj}$ is the count of how many deaths from cause k contain symptom j in the training data.

   The Tariff scores are transformed into rankings through comparison with a reference distribution obtained by resampling the training dataset to achieve uniformity and consequently, the algorithm assigns CoD distributions based on these rankings. Therefore, calculation of cause-specific mortality fractions is based on the highest-ranked cause for each death, i.e.,

$$p_k = \frac{\sum_{i=1}^{\mathcal{X}} 1_{y_i=k}}{\mathcal{X}} \tag{3.5}$$

The model became more precise by considering fewer symptoms/signs (Serina et al., 2015), but it necessitates a training dataset from a prior validation study for implementation (Kalter et al., 2016), and missing indicators are still assumed to be equivalent to missing data, like in InterVA and NBC.

5. **King-Lu**. The model, discussed in detail in Section 2.4.5 provides a holistic distribution of Causes of Death (CoD) rather than individual occurrences by leveraging conditional probability distributions to calculate the overall CoD distribution (King and Lu, 2008) and therefore is the only one among the CCVA methods that can be considered to perform quantification. For each category, such as age, sex, or condition, King-Lu necessitates a distinct sample. The method, unbiased and adjustable by experts for symptom sampling, exhibits limitations in effectiveness with small sample sizes (McCormick et al., 2016).

6. **Simplified Symptom Pattern (SSP)**. This hybrid model (Leitao et al., 2014) integrates elements from King-Lu and InterVA, utilizing Bayesian logic to adapt the symptom pattern. Drawing on King-Lu's direct CSMF approximation and the probability of item responses based on the true cause in verbal autopsy data, SP provides pre-distribution probabilities. It demonstrates accuracy in both individual cause-of-death determination and CSMF outcomes, although its implementation for single CoD analysis requires significant computational resources compared to other methods (Lozano et al., 2011).

7. **Random Forest**. It is a machine learning approach that predicts the cause of death generating numerous decision trees trained to differentiate between cause pairs, and then combines outcomes via an innovative ranking approach (James et al., 2011). Due to the large codeframes of CoDs, this technique demands substantial computational power and poses challenges in interpretation (Tunga et al., 2021).

3.1.2 Quantification for Verbal Autopsies

It should be noted that, being the aggregate data the main focus of epidemiology, the inference target comprises two components (Li et al., 2022): individual cause-of-death assignments and population-level CSMF. CSFM is intended as the estimates of the distribution of CoDs at the population level (Jeblee et al., 2019), and is measured by CSFM accuracy (Flaxman et al., 2015), defined as one minus the sum of all absolute CSMF errors across causes divided by the maximum total error (Murray et al., 2011b). We will not consider CSFM estimation as quantification as, except for King-Lu method, in the models listed here it is done by counting the output classes of the classifier, and our baseline is already a classify and count method.

3.1.3 Text-Leveraging Methods for Verbal Autopsies

As already mentioned, VAs questionnaires are made from closed-ended questions (symptoms that can be present, absent or in some cases missing) and a free narrative, that provides a verbatim account of the interviewee's answer. Despite the considerable wealth of information contained in this last part, it has not been fully utilised for most of the history of VAs, due to the intrinsic logic of the employed models, which only consider the presence of symptoms.

In 2014, the review by Danso et al. (2013b) stated that computational approaches up to that point had predominantly utilised structured data, except for a solution described by Murray et al. (2011a) which involved preprocessing and storing textual data as part of a coded database. This solution converted free text into dichotomous variables by retaining only the most relevant keywords and eliminating syntactic structure.

In (Danso et al., 2013b), several classifiers including SVM, naïve Bayes, and random forest were employed with word frequency counts and TF-IDF scores to analyze narratives, resulting in moderate F_1 scores. In another study (Danso et al., 2013a) by the same authors, linguistic features from narratives of infant deaths in Ghana were utilised to classify causes of death, demonstrating improved performance with combined narrative and questionnaire features despite limitations in dataset size and the accuracy of the part-of-speech tagger.

Hence, traditionally, the narratives were not utilised for CCVA, and the majority of contributions still refrain from doing so today (Blanco et al., 2021), because of data limitations such as incomplete textual input, with many samples being non-informative (i.e., "*respondent has nothing to say*"), the presence of non-informative samples and lengthy responses, and high number of out-of-vocabulary words in the test set due to lexical variability in the clinical field (Bailo et al., 2022; Blanco et al., 2021; Jeblee et al., 2019).

However, even short narratives may carry meaningful information (i.e., *"he drowned")* and new technologies such as transformer-based models (Vaswani et al., 2017) are able to overcame such problems, so that some methods either exclusively use them (Jeblee et al., 2019; Manaka et al., 2022) or integrate their information with the one derived from closed questions (Blanco et al., 2021; Danso et al., 2013a). All the studies just cited agree that using both the information coming from structured data and that coming from textual narrative, exploiting a transformer-based model, leads to an improvement in accuracy.

Finally, the idea of exploiting the hierarchical structure of the ICD-10 codeframe for classification has been implemented in (Blanco et al., 2022), considering that leveraging the relationship between a label and its descendants, through a multitask BERT-based transfer learning approach, would be beneficial to consistency between predictions.

Chapter 4

A Quantification-Based Approach for Estimating the Distribution of Causes of Death from Verbal Autopsies

In the previous chapters, we described the hierarchical structure of CoD codeframes (Section 2.2.2) and stated that quantification, a process of estimating the prevalence of categories within datasets, is more aligned with epidemiological goals than mere classification.

The proposed task for this work is to predict the distribution of CoD in verbal autopsies textual datasets, with the aim of improving the estimation of mortality cause prevalence.

It is a supervised learning task, with physician-determined causes of death serving as ground truth labels.

Before moving further, we will briefly focus on the three approaches that come to mind when framing a quantification task on a hierarchical dataset.

The first approach will be to ignore the hierarchical relations among the classes and use a both a classifier and a quantifier that treat the set of classes as the set of leaf nodes. This step (Single-label Multiclass Non-Hierarchical Classify and Count; Single-label Multiclass Non-Hierarchical ACC, PCC, PACC, SLD) is further covered in Section 5.5 (1) and 5.5 (2).

The second approach will then be to use a quantifier that relies on a hierarchical classifier (Single-label Multiclass Hierarchical Classifier, Non-Hierarchical Classify and Count; Single-label Multiclass Hierarchical Classifier, Non-Hierarchical ACC), further detailed again in Section 5.5(3).

The third and final approach is to leverage the branching structure with a hierarchical quantifier.

This is the most innovative approach, as (i) we propose the direct estimation of cause-of-death prevalence rather than using a classify-and-count approach, and (ii) we exploit the hierarchical structure of causes of death, comparing its performance with algorithms that treat the codeframes as a flat structure. The methodology employed for the implementation of these new algorithms is covered in this section.

4.1 A Hierarchical Quantification System

4.1.1 Hierarchical ACC, a Method for Hierarchical Quantification

We here describe the *Hierarchical Adjusted Classify and Count* (**H(ACC)**) method for hierarchical quantification (see Algorithms 1 and 2). As the name implies, H(ACC) is a hierarchical version of the ACC "flat" multiclass quantification method described by Equations 2.2 and 2.3. The basic pattern that H(ACC) follows is that of (a) applying ACC to the root node and, recursively, to its subtrees, in a top-down fashion, and (b) propagating down, from the root and the internal nodes to the leaf nodes, the corrections to the prevalence estimates that ACC computes.

We train H(ACC) (see Algorithm 1) by recursively visiting, in a top-down fashion, tree $\mathcal{Y}_0$. Essentially, this visit consists of training, for each internal node y_i, a single-label multiclass classifier h_i that uses as codeframe the set $\downarrow(y_i)$ of the children of y_i, and of estimating, via k-fold cross-validation, the misclassification rates of classifier h_i; these misclassification rates will be used at quantification time in order to perform ACC for node y_i.

At quantification time, we apply H(ACC) (see Algorithm 2) by recursively (a) performing, for each internal node y_0, Classify and Count on the set $\downarrow(y_0)$ of the children of y_0, (b) correcting the estimates $\hat{p}_i$ computed by Classify and Count for each $y_i \in \downarrow(y_0)$ by invoking multiclass ACC, and (c) conditioning (see Line 8) the corrected estimates thus computed on the estimate of p_0. Step (c) is needed since the estimated prevalence values $\hat{p}_i$ computed in Step (b) are "local" to $\mathcal{Y}_0$, i.e., these estimated prevalence values $\hat{p}_i$ sum to 1; however, when H(ACC) is recursively called on a *subtree*, this sum should not be 1 but p_0, which is the reason why p_0 is an input parameter (in the "outermost" call of H(ACC) this parameter is set to 1) and why in Step (c) (see Line 7 of Algorithm 3) we multiply all the $\hat{p}_i$'s by p_0 .

Note that only the datapoints assigned to node y_i by the classifier associated with its parent node $\uparrow y_i$ are presented as input to classifier h_i; in other words, classifiers at internal nodes act as "filters", that prevent datapoints from descending through subtrees they are deemed not to belong to. Step (a) relies on the single-label multiclass classifiers generated at training time, while Step (b) relies on the misclassification rates estimated at training time.

One important difference between the main (outermost) call of Algorithm 2

```
Input  : Hierarchical codeframe $\mathcal{Y}_0 = \{y_0, y_1, ..., y_n, y_{n+1}, ..., y_{n+m}\}$
         Training set $L = L_1 \cup ... \cup L_n$
Output: Set $\mathcal{H}_0 = \{h_i\}_{i \in \{0,n+1,...,n+m\}}$ of trained single-label classifiers
        $h_i : \mathcal{X} \rightarrow \downarrow(y_i)$
         Set $\mathcal{R}_0 = \{R_i\}_{i \in \{0,n+1,...,n+m\}}$ of sets $R_i = \{r_{(y',y'')}\}_{y',y'' \in \downarrow(y_i)}$ of
misclassification rates

// Main body
if NonLeaf($y_0$) then // Nothing needs to be done if $y_0$ is a leaf
    Train a single-label classifier $h_0 : \mathcal{X} \rightarrow \downarrow(y_0)$
    for $y', y'' \in \downarrow(y_0)$ do
        use $k$-fold cross-validation to compute
          the misclassification rates
          $r_{(y',y'')} = \frac{|\{\mathbf{x} \in L \ : \ \mathbf{x} \in y', h_0(\mathbf{x}) = y''\}|}{|\{\mathbf{x} \in L \ : \ \mathbf{x} \in y'\}|}$
    end
    for $y_i \in \downarrow(y_0)$ do
        TrainHACC($\mathcal{Y}_i, L_i$) // Recursively repeat for all subtrees
    end
end
```

Algorithm 1: The TrainHACC($\mathcal{Y}_0, L$) algorithm.

and its recursive (innermost) calls is that in the former the input parameters σ_0 and p_0 are not affected by uncertainty (i.e., we know for certain that σ_0 is the set of datapoints that belong to y_0, and we know for certain that $p_0 = 1$), while in the latter they are indeed affected by uncertainty (i.e., σ_0 and p_0 represent *estimates*, since σ_0 is the set of datapoints that have been attributed to y_0 by the classifier associated with $\uparrow y_0$, and p_0 is thus an estimate of the true prevalence of class y_0). The fact that the recursive innermost calls operate on estimates is, of course, a possible source of quantification error, but a source that unfortunately cannot be eliminated, since for these calls the true values of σ_0 and p_0 are unknown.

4.1.2 Hierarchical PACC, a Probabilistic Variant of H(ACC)

Hierarchical Probabilistic Adjusted Classify and Count (**H(PACC)** – see Algorithms 3 and 4) is, as the name implies, a hierarchical algorithm built on top of the PACC quantification method discussed in Section 2.4.3. As such, it can be defined along lines similar to those of H(ACC); in the present section we will thus only focus on the differences between H(ACC) and H(PACC).

The main difference between the two methods is that the classifiers trained by H(PACC) are probabilistic, i.e., for datapoint $\mathbf{x}$ they do not return a class but a set of class-specific confidence scores, in the form of posterior probabilities

```
Input  : Hierarchical codeframe $\mathcal{Y}_0 = \{y_0, y_1, ..., y_n, y_{n+1}, ..., y_{n+m}\}$
         Set $\mathcal{H}_0 = \{h_i\}_{i \in \{0,n+1,...,n+m\}}$ of trained single-label classifiers
         $h_i : \mathcal{X} \rightarrow \downarrow(y_i)$
         Set $\mathcal{R}_0 = \{R_i\}_{i \in \{0,n+1,...,n+m\}}$ of sets $R_i = \{r_{(y',y'')}\}_{y',y'' \in \downarrow(y_i)}$ of
         misclassification rates
         Sample $\sigma_0$
         Prevalence $p_0$ of $y_0$ in $\sigma_0$
Output: Estimated distribution $\hat{\mathbf{p}} = (\hat{p}_1, \ldots, \hat{p}_n)$ over the leaf classes
         $y_1, ..., y_n$

// Main body
1  if NonLeaf($y_0$) then // Nothing needs to be done if $y_0$ is a leaf
2      Apply $h_0$ to $\sigma_0$ ;                                  // Classify ...
3      for $y_i \in \downarrow(y_0)$ do
4          $\sigma_i \leftarrow \{\mathbf{x} \in \sigma_0 \;:\; h_0(\mathbf{x}) = y_i\}$
5          $\hat{p}_i \leftarrow |\sigma_i|/|\sigma_0|$ ;                          // ...  and count
6      end
7      Apply multiclass ACC, using $\mathcal{R}_0$ to re-compute $\hat{p}_i$ for all $y_i \in \downarrow(y_0)$
8      $\hat{p}_i \leftarrow \hat{p}_i \cdot p_0$  ;            // Propagate downwards the estimated
        prevalence values
9      for $y_i \in \downarrow(y_0)$ do
10         ApplyHACC($\mathcal{Y}_i, \mathcal{H}_i, \mathcal{R}_i, \sigma_i, \hat{p}_i$)  // Recursively repeat for all
               subtrees
11     end
12 end
```

Algorithm 2: The ApplyHACC($\mathcal{Y}_0, \mathcal{H}_0, \mathcal{R}_0, \sigma_0, p_0$) algorithm. In its main (outermost) call, parameter p_0 must be set to 1.

$\Pr(y_i|\mathbf{x})$. As a consequence, at quantification time there is no such a thing as σ_i (the set of datapoints assigned to class y_i by the classifier associated to y_0 – see Line 4 of Algorithm 2). This notion is replaced by a probability mass $\sum_{\mathbf{x} \in \sigma} \Pr(y_i|\mathbf{x})$ assigned to class y_i (see Line 4 of Algorithm 4), and it is this probability mass that forms the basis for the initial estimate $\hat{p}_i$. This estimate is then corrected by using PACC (instead of ACC) and, as for the method of Section 4.1.1, a recursive call is issued. The only difference between the recursive calls to H(ACC) and H(PACC) is that, in this latter, the same sample σ as used in the outer call is passed as a parameter, exactly because there is no such a thing as "the set of datapoints assigned to class y_i".

Note that by omitting the computation of the misclassification rates (Lines 4-7 in Algorithm 3) and the call to PACC (Line 5 in Algorithm 4) we obtain a hierarchical version of the PCC method (see Equation 2.4), that we might dub H(PCC).

```
Input  : Hierarchical codeframe $\mathcal{Y}_0 = \{y_0, y_1, ..., y_n, y_{n+1}, ..., y_{n+m}\}$
         Training set $L = L_1 \cup ... \cup L_n$
Output: Set $\mathcal{S}_0 = \{s_i\}_{i \in \{0,n+1,...,n+m\}}$ of trained probabilistic classifiers
        $s_i : \mathcal{X} \rightarrow \Delta^{t_i - 1}$
        Set $\mathcal{R}_0 = \{R_i\}_{i \in \{0,n+1,...,n+m\}}$ of sets $R_i = \{r_{(y',y'')}\}_{y',y'' \in \downarrow(y_i)}$ of
            probabilistic misclassification rates

// Main body
if NonLeaf($y_0$) then // Nothing needs to be done if $y_0$ is a leaf
    Define $t_0 = |\downarrow y_0|$
    Train a probabilistic classifier $s_0 : \mathcal{X} \rightarrow \Delta^{t_0 - 1}$
    for $y', y'' \in \downarrow(y_0)$ do
        use $k$-fold cross-validation to compute the
          probabilistic misclassification rates
          $r_{(y',y'')} = \sum_{\mathbf{x} \in L, \mathbf{x} \in y'} \frac{\Pr(y''|\mathbf{x})}{|\mathbf{x} \in L, \mathbf{x} \in y'|}$
    end
    for $y_i \in \downarrow(y_0)$ do
        TrainHPACC($\mathcal{Y}_i, L_i$)          // Recursively repeat for all
            subtrees
    end
end
```

Algorithm 3: The TrainHPACC($\mathcal{Y}_0, L$) algorithm.

Note though that H(PACC) is much less efficient than H(ACC). To see this note that, at quantification time, in H(ACC) each classifier needs to classify only sample σ_i, a subset of σ, while in H(PACC) each classifier needs to classify the entire sample σ. In other words, by recursively descending the tree, in H(ACC) we remove exponentially many unlabelled datapoints from consideration, which makes H(ACC) substantially more efficient than H(PACC).

4.1.3 Hierarchical SLD

Hierarchical SLD (**H(SLD)** – see Algorithms 5 and 6) is instead a hierarchical variant of the SLD multiclass quantification method (see Equations 2.7 and 2.6 in Section 2.4.3). H(SLD) has a number of similarities with H(PACC), especially since it also relies on probabilistic classifiers, and can thus be defined along lines similar to those of H(PACC); in the present section we will thus only dwell on the differences between H(PACC) and H(SLD).

One of the differences between these two methods is that in the latter, unlike in the former, there is no computation of misclassification rates, since the use of these rates is specific to "prevalence correction" methods such as ACC and PACC.

```
Input  : Hierarchical codeframe $\mathcal{Y}_0 = \{y_0, y_1, ..., y_n, y_{n+1}, ..., y_{n+m}\}$
         Set $\mathcal{S}_0 = \{s_i\}_{i \in \{0,n+1,...,n+m\}}$ of trained probabilistic classifiers
         $s_i : \mathcal{X} \rightarrow \Delta^{t_i - 1}$
         Set $\mathcal{R}_0 = \{R_i\}_{i \in \{0,n+1,...,n+m\}}$ of sets $R_i = \{r_{(y',y'')}\}_{y',y'' \in \downarrow(y_i)}$ of
           probabilistic misclassification rates
         Sample $\sigma$
         Prevalence $p_0$ of $y_0$ in $\sigma$
Output: Estimated distribution $\hat{\mathbf{p}} = (\hat{p}_1, \ldots, \hat{p}_n)$ over the leaf classes
         $y_1, ..., y_n$
// Main body
if NonLeaf($y_0$) then // Nothing needs to be done if $y_0$ is a leaf
    Apply $s_0$ to $\sigma$
    for $y_i \in \downarrow(y_0)$ do
        $\hat{p}_i \leftarrow \frac{1}{|\sigma|} \sum_{\mathbf{x} \in \sigma} \Pr(y_i|\mathbf{x})$ ;    // Initialise class prevalence
          estimates
    end
    Apply multiclass PACC, using $\mathcal{R}_0$ to re-compute $\hat{p}_i$ for all $y_i \in \downarrow(y_0)$
    $\hat{p}_i \leftarrow \hat{p}_i \cdot p_0$ ;    // Propagate downwards estimated prevalence
      values
    for $y_i \in \downarrow(y_0)$ do
        ApplyHPACC($\mathcal{Y}_i, \mathcal{S}_i, \mathcal{R}_i, \sigma, \hat{p}_i$) // Recursively repeat for all
          subtrees
    end
end
```

Algorithm 4: The ApplyHPACC($\mathcal{Y}_0, \mathcal{S}_0, \mathcal{R}_0, \sigma, p_0$) algorithm. In its main (outermost) call, parameter p_0 must be set to 1.

In the training phase, instead, H(SLD) needs to compute, from the training data (see Line 5 of Algorithm 5), initial estimates $\hat{p}_i$ of the class prevalence values, since they will form the basis of the iteration in the quantification phase. In the latter phase, the main aspect that characterises H(SLD) is the iterative update, until convergence, of the posterior probabilities (see Line 2 of Algorithm 6) and of the class prevalence estimates (Line 4).

```
Input  : Hierarchical codeframe $\mathcal{Y}_0 = \{y_0, y_1, ..., y_n, y_{n+1}, ..., y_{n+m}\}$
         Training set $L = L_1 \cup ... \cup L_n$
Output: Set $\mathcal{S}_0 = \{s_i\}_{i \in \{0,n+1,...,n+m\}}$ of trained probabilistic classifiers
        $s_i : \mathcal{X} \rightarrow \Delta^{t_i - 1}$
        Set $\mathcal{P}_0 = \{\hat{p}_i\}_{i \in \{1,...,n+m\}}$ of initial class prevalence estimates

// Main body
if NonLeaf($y_0$) then // Nothing needs to be done if $y_0$ is a leaf
    Define $t_0 = |\downarrow y_0|$
    Train a probabilistic classifier $s_0 : \mathcal{X} \rightarrow \Delta^{t_0 - 1}$
    for $y_i \in \downarrow(y_0)$ do
        Compute $\hat{p}_i = \frac{|L_i|}{|L|}$ ;       // Initialise class prevalence
         estimates
        TrainHSLD($\mathcal{Y}_i, L_i$)  // Recursively repeat for all subtrees
    end
end
```

Algorithm 5: The TrainHSLD($\mathcal{Y}_0, L$) algorithm.

```
Input  : Hierarchical codeframe $\mathcal{Y}_0 = \{y_0, y_1, ..., y_n, y_{n+1}, ..., y_{n+m}\}$
         Set $\mathcal{S}_0 = \{s_i\}_{i \in \{0,n+1,...,n+m\}}$ of trained probabilistic classifiers
         $s_i : \mathcal{X} \rightarrow \Delta^{t_i - 1}$
         Set $\mathcal{P}_0 = \{\hat{p}_i\}_{i \in \{1,...,n+m\}}$ of initial class prevalence estimates
         Sample $\sigma$
         Prevalence $p_0$ of $y_0$ in $\sigma$
Output: Estimated distribution $\hat{\mathbf{p}} = (\hat{p}_1, \ldots, \hat{p}_n)$ over the leaf classes
         $y_1, ..., y_n$
// Main body
if NonLeaf($y_0$) then // Nothing needs to be done if $y_0$ is a leaf
    Apply $s_0$ to $\sigma$
    Apply multiclass SLD (i) to update $\Pr(y_i|\mathbf{x})$ for all $y_i \in \downarrow(y_0)$ and for
      all $\mathbf{x} \in \sigma$
                         (ii) to re-estimate $\hat{p}_i$ for all $y_i \in \downarrow(y_0)$
    $\hat{p}_i \leftarrow \hat{p}_i \cdot p_0$ ;      // Propagate downwards estimated prevalence
      values
    for $y_i \in \downarrow(y_0)$ do
        ApplyHSLD($\mathcal{Y}_i, \mathcal{S}_i, \mathcal{P}_i, \sigma, \hat{p}_i$)    // Recursively repeat for all
          subtrees
    end
end
```

Algorithm 6: The ApplyHSLD($\mathcal{Y}_0, \mathcal{S}_0, \mathcal{P}_0, \sigma, p_0$) algorithm. In its main (outermost) call, parameter p_0 must be set to 1.

Chapter 5

Experiments

5.1 Datasets

Our interest lies in exploring quantification approaches that can effectively handle a pre-defined class hierarchy. This structure can be generally found in VAs datasets labels, as the most commonly used codeframe ICD has a branching structure of causes of death into more specific pathologies or conditions, as described in Section 2.2.2.

Although numerous datasets of verbal autopsies are available, obtaining the narratives section is not at all easy, for privacy reasons (for example, the texts must be anonymised), or because having a digital copy would require the transcription and possible translation of the paper form by of the structure in possession of the data.

The VAs datasets that include a textual part that we obtained and used, are Million-Death Study (MDS) Ke et al. (2021) and The Karonga Health and Demographic Surveillance System INDEPTH dataset (Karonga HDSS) Crampin and Dube (2017).

The aforementioned hierarchical structure necessary for these experiments is actually widespread in a different number of applications, and therefore we decided to also use the well-known 20newsgroup dataset, that comprises around 18,000 newsgroups posts on 20 topics, imported through the Scikit-learn (Pedregosa et al., 2011) library. Although it is not a VAs dataset, we decided to employ it to get further experimental confirmation of the functioning of hierarchical quantification algorithms.

	first-level classes	total classes	total VAs	VAs with narratives
Karonga HDSS	5	57	5k+	3366
MDS Adults subset	21	174	90k+	7884

Table 5.1: Main characteristics of the VA datasets we consider.

As some previously cited works (Blanco et al., 2021; Danso et al., 2013b) claimed an improvement in CoD assignment accuracy when leveraging both the closed questions and the textual narrative, we will test our models both on a narratives-only version and a concatenated version of both MDS and Karonga HDSS datasets.

5.1.1 Million-Death Study

Full details of this dataset history can be found in Aleksandrowicz et al. (2014); Brown et al. (2022); Dikshit et al. (2012); Jana et al. (2022); Ke et al. (2021) among others. The documents for which a narrative is available only cover a small portion of the MDS records, as they were manually transcribed from images by Jeblee et al. (2019), and translated to English when necessary. The causes of death are coded according to the ICD-10 codeframe. The dataset is further divided into three subsets, according to the ages of the deceased:

- Adults (12 years or older). It is the largest subset, counting more than 8400 deaths with a narrative available, besides other about 120 dichotomous symptoms. There are 482 different labels assigned, but without taking into account classes with 5 or fewer examples each, we arrive at 174 unique labels for CoD, distributed unevenly among 7884 single-death reports. As can be observed in 5.1 (c), the first-level labels are 21, coded by letters according to the ICD standard.

- Children (28 days to 11 years). It contains 1299 documents and 42 labels, after removing uncommon CoDs (again, 5 or less deaths each). The main 5 CoDs (namely fever, epilepsy, pneumonia, infectious gastroenteritis and sepsis) are responsible for half of all deaths.

- Neonates (under 28 days). This is the smallest with 412 documents and 11 unique labels, after removing the reports assigned with the 37 rare CoDs. The most common 5 CoDs (all conditions specifically originating in the perinatal period) are responsible for 64% of all deaths.

From a textual point of view, the spelling of the narratives from this dataset was corrected by the transcribers, that also removed stop words, lowercased and stemmed the text.

Although not being able to fully exploit the dataset is unfortunate, it is not possible to merge the three subsets into a single, larger one, because they diverge in they were created after three different questionnaires, and therefore leverage different symptoms and questions. In example, the children and adult datasets only share a third of the total variables. Therefore, the experiments will be run only on the adults subset, as it is the largest, given that the results on a smaller data collection with fewer labels are already investigated using the Karonga HDSS dataset, illustrated in detail shortly.

To avoid data spilling, the columns containing other version alternative to the finally assigned CoD were removed from the data.

The fact that in this dataset, although larger, only registers 26 deaths related to AIDS, while they are responsible of almost a quarter of the total Karonga HDSS dataset, is a good example of how difficult it can be for CCVA tools to generalise over different geographical regions.

5.1.2 Karonga HDSS

The Karonga HDSS dataset was assembled in the southern part of Karonga District, around Lake Malawi (Malawi), collected by Malawi Epidemiology and Intervention Research Unit (MEIRU) in collaboration with London School of Hygiene and Tropical Medicine. The dataset contains more than 5000 documents with a narrative, but once those without a complete label that we can use as golden labels have been removed or without a narrative, it narrows down to 3366 reports. The fact that the dataset contains reports from all ages could likely be the reason for the large number of closed-answer questions included, around 280, more than twice the ones contained in the MDS.

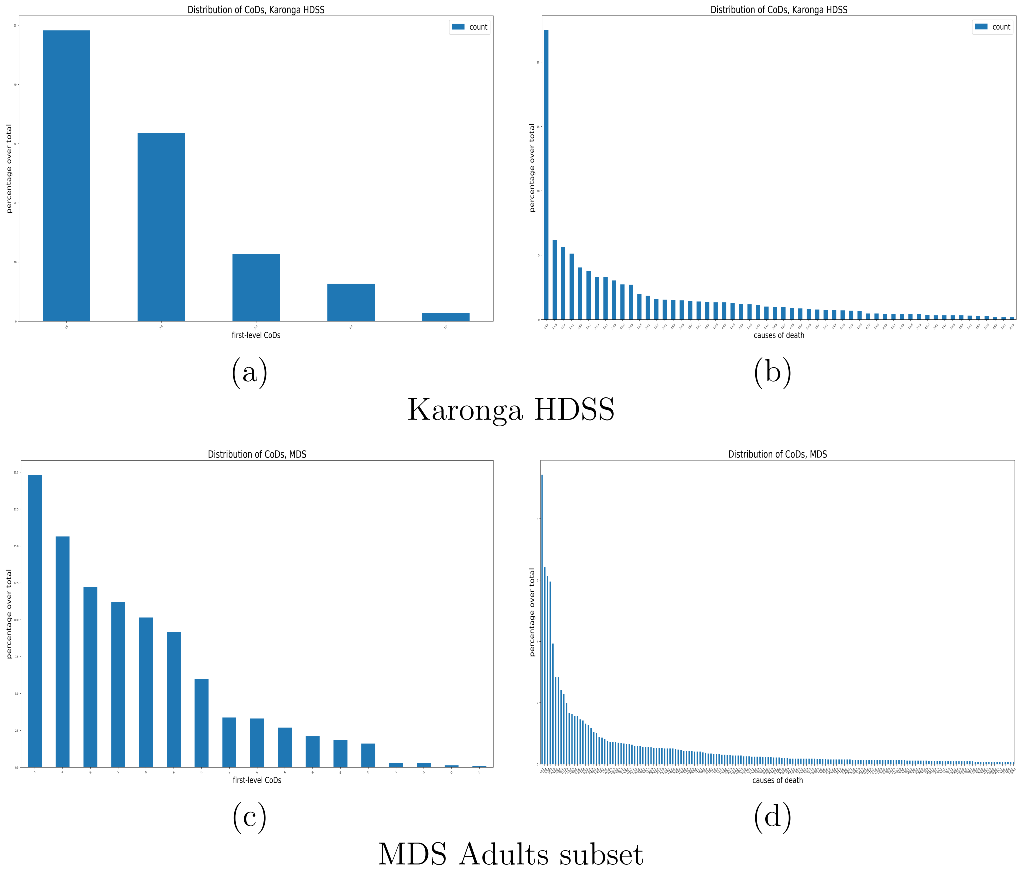
An in-house hierarchical coding system is used, and this variation in protocol is the reason why it was not merged into a single larger dataset with documents from different sources. This codeframe includes 5 labels for the first level (communicable disease, non-communicable disease, infancy-related, maternity-related, external cause) and again the choice is between 10 labels to the utmost for the following levels, given that it is created with ICD-10 as a reference. Total labels after taking out ones with 5 or less items are 57. To explore the contribution of narratives and closed responses in assigning the CoD, we create two sub-sets, the first containing exclusively the narratives and the second with the concatenation of symptoms and the open response.

As can be seen in Figure 5.1 (b), the presence of AIDS causes a notable imbalance in the dataset, as around a quarter of the total deaths is attributed to it alone. The main causes of death after that are acute febrile illness, pneumonia and malaria, followed by birth injury/asphyxia.

Although there are some shorter narratives, most are composed of multiple sentences (average sentence length is around 90 words) separated by a period, and there are no obvious errors, although the texts appear simplified during possible transcription/ translation (for example, by lemmatising verbs or eliminating connectives).

5.2 Evaluation Measures

As claimed by Silla and Freitas (2010), there seems to be a lack of standardised methods for evaluating hierarchical classification systems or establishing ex-



(a) (b)

Karonga HDSS

(c) (d)

MDS Adults subset

Figure 5.1: Barplots of CoDs distribution over the VAs datasets

periments consistently. Evaluation metrics commonly used in flat classification are the predominant choices for assessing Hierarchical Text Classification (HTC) systems (Esuli et al., 2008), although some works (Ceci and Malerba, 2007; Sun and Lim, 2001) argued that evaluation measures specific to the hierarchical case should be used. For the evaluation of our systems we employed conventional measures (i.e., MAE, MRAE), disregarding the presence of the internal nodes and considering our problem as a simple, single-label multiclass quantification problem, since the datapoints all belong to leaf classes. Following the notation in Section 2.1, MRAE and MAE are defined as

$$\mathrm{MRAE}(p_\sigma, \hat{p}_\sigma) = \frac{1}{n} \sum_{y \in \mathcal{Y}} \frac{|\hat{p}_\sigma(y) - p_\sigma(y)|}{p_\sigma(y)}$$

$$\mathrm{MAE}(p_\sigma, \hat{p_\sigma}) = \frac{1}{n} \sum_{y \in \mathcal{Y}} |\hat{p}_\sigma(y) - p_\sigma(y)|$$

MRAE will be undefined if at least one of the classes $y \in \mathcal{Y}$ is absent (has a prevalence of 0) in the sample σ of unlabelled items (test set). To solve this problem, we smooth all $p_\sigma(y)$ and $\hat{p}^\sigma(y)$ values using additive smoothing as described in Esuli et al. (2022b), by taking

$$\underline{p}_\sigma(y) = \frac{\epsilon + p_\sigma(y)}{\epsilon \cdot n + \sum_{y \in \mathcal{Y}} p_\sigma(y)}$$

where $\underline{p}_\sigma(y)$ denotes the smoothed version of $p_\sigma(y)$ and the denominator is a normalizing factor, doing of course the same to the $\underline{\hat{p}}_\sigma(y)$ values. Following Forman (2008) , we use $\epsilon = 1/(2|\sigma|)$ as the smoothing factor. In Equation 1, we then use the smoothed versions of $p_\sigma(y)$ and $\hat{p}_\sigma(y)$ in place of their original non-smoothed versions, ensuring that MRAE is now always defined.

5.3 Evaluation Protocol

Any test set typically employed to assess the accuracy of classification can of course also be utilised to evaluate quantification. However, a challenge arises in quantification as a set of k unlabelled data offers k individual data points for classification but it only provides one test data point for quantification. Consequently, evaluating quantification algorithms becomes challenging due to the limited availability of labelled data for testing purposes.

For test samples extraction, we here use a variant of the widely adopted *artificial prevalence protocol* (APP) (Esuli et al., 2023), first used by Saerens et al. (2002) and experimented in Forman (2005). This version, that we will refer as *uniform prevalence protocol* (UPP), allows keeping the number of distribution vectors generated by the APP protocol practical by simply renouncing to predetermine class prevalence values, allowing them to vary randomly instead. This is done by initially generating a random distribution p, and then generating a sample σ, by randomly selecting items from the population based on p. To ensure that all the distribution vectors p are uniformly sampled at random, so that all legitimate distribution vectors are equally likely, the Kramer algorithm is used as presented in Smith and Tromble (2004). Compared to sampling from a predefined grid, the Kraemer algorithm allows drawing a specific number of samples, rather than requiring the generation of all $K(m, n)$ valid combinations and enables the selection of any possible distribution vector. However, unlike APP, there's no assurance of precise equal coverage across all classes. However, it's favoured in situations where the vast number of potential combinations of APP's grid values renders the task unmanageable. The initial experimental setup in quantification to utilise the Kraemer algorithm as the sample-generating function is detailed in Esuli et al. (2022a).

5.4 System Configuration and Parameter Optimisation

The implementation of our experiments was realised using the QuaPy open-source Python library (Moreo et al., 2021).

5.4.1 Text Preprocessing

As already mentioned in Section 5.1, the textual portions of both datasets were revised by their respective creators, who corrected the spelling, removed the stop words and lemmatised them. Therefore it was not necessary to carry out any other text-preprocessing operations, except to ensure that no other versions of the labels with the CoD were included in the concatenated version of Karonga HDSS.

We utilised both TF-IDF (Term Frequency-Inverse Document Frequency) document vectors and pre-trained embeddings from the DistilBERT model (Sanh et al., 2019).

TF-IDF embeddings represent text by weighting the importance of each word in a document relative to its frequency across all documents in the corpus, thereby highlighting unique terms that are significant for specific documents. In other terms, it transforms the entire document into a fixed-size vector, where each dimension corresponds to a unique term's TF-IDF score within the document. Pre-trained embeddings from transformer models are instead computed by passing text through multiple layers of attention mechanisms and neural networks, which capture contextual relationships and semantic nuances of words based on extensive training on large text corpora. The computational demands of these methods differ significantly: TF-IDF is computationally lightweight and efficient, as it involves simple statistical calculations, while DistilBERT requires substantial computational resources due to its deep neural network architecture and the complexity of its contextual word representations.

5.4.2 Configuration of the Hierarchical Quantification System

We use quantification methods that have an aggregative nature; this means that these algorithms first require the training of a (hard or soft) classifier to issue a prediction for each individual item, and that then output an estimated class prevalence based on these individual predictions (Esuli et al., 2023). The details for the algorithms that estimate class prevalence values can be found in Section 2.4, while as the techniques outlined in that section are applicable across any supervised learning algorithm for training classifiers (Esuli et al., 2023), we chose to use one of logistic regression (LR) and C-support vector classification (SVM) with radial basis function kernel, as the underlying classification algorithm.

5.5 Baselines

The approach we followed when adopting a baseline has been based on the assumption that (i) the classify and count method would lead to poorer results than the more sophisticated ones, and (ii) the hierarchical methods would be better performing than the flat ones. Therefore, the approaches here listed are in ascending order of complexity:

1. **Single-label Multiclass Non-Hierarchical Classify and Count**: it ignores both the hierarchical structure and the nature of quantification task. It is the straight application of the Classify and Count algorithm as described in Section 2.4;

2. **Single-label Multiclass ACC, PCC, PACC, SLD**: they ignore the hierarchical structure but are "real" quantification algorithms; It is the straight application of the quantification algorithms as described in Section 2.4;

3. **Single-label Multiclass Hierarchical Classifier, Classify and Count** and **ACC**, hereafter respectively denoted as h(CC) and h(ACC). These algorithms first employ a hierarchical classifier to categorize the data based on the hierarchical structure of the classes, and then pass the classification results to a quantifier to estimate the prevalence of each category within the dataset. The hierarchical classifier used in this baseline returns only a label assigned to each document rather than posterior probabilities (hard classifier). As a result, quantifiers that require a probability distribution over the labels for each document cannot function properly, and thus this baseline is assessed only with the CC and ACC methods.

4. **Single-label Multiclass Hierarchical Classify and Count:** our fourth baseline is a hierarchical single-label multiclass version of Classify and Count, hereafter denoted H(CC) and correspond to Algorithm 1 with lines 3-6 removed, plus Algorithm 2 with Line 6 removed. Our expectation about this method are low because it ignores he nature of the quantification task, and because classification errors made high up in the hierarchy propagate downwards, i.e., cannot be recovered at the lower leaves of the hierarchy. For instance, assuming that all classifiers in the hierarchy have .9 accuracy, in a hierarchy of 10 levels the accuracy at the 10-th level will be $.9^{(10-1)} \approx .38$.

Chapter 6

Results

In this section we discuss the results obtained with the setting described in Chapter 5. Following the research questions we posed in the introduction, we will start by discussing whether using "real" quantifiers brings benefits compared to a classify and count approach (i). We will then discuss the results obtained using each quantifier with respect to its hierarchical variant (ii), including their performance in terms of time (iii). Furthermore, we will discuss the results obtained by exploiting different types of text representations, and the contribution of closed questions when concatenated to the narratives of each dataset.

The gaps in the upcoming tables are due to the fact that PACC should not be used with a non-probabilistic classifier like SVM, because it relies on calibrated probability estimates for accurate performance, which SVM does not inherently provide without additional calibration steps, for performing which there are often insufficient training data.

6.1 Classify-and-Count approach vs. Direct Estimate of Prevalence

This work starts from the awareness of the suboptimality of Classify and Count, given that a perfect classifier is obviously also a perfect quantifier, but that there is no guarantee that a good classifier is also a good quantifier. In other words, a classifier in which false positives and false negatives compensate each other may appear to be an excellent classifier, but it will give very bad results if it is the basis of a quantification task.

Proceeding to directly estimate the prevalence of classes in the dataset may therefore be better, as it avoids this step, especially for unbalanced datasets (like Karonga HDSS, in which AIDS is responsible for almost a quarter of the total deaths).

In light of these premises, we first expect that more complex algorithms such as ACC, PCC and even more so PACC and SLD will give better results than CC.

This occurs only partially and in less marked terms than we would have expected: in all datasets the results obtained by the quantification algorithms in terms of MAE and MRAE are substantially similar to those obtained with classify and count.

In the VAs datasets, PCC and PACC perform better in all the tested settings than CC, which however maintains substantially similar results, as reported in Tables 6.1, 6.3 6.5 and 6.7. The results remain better for all quantification algorithms compared to CC if Logistic Regression is the classifier, while when SVM is adopted, CC achieves even better results than ACC and SLD, once again remaining in very close values.

The described pattern stays the same for all three datasets, with Karonga HDSS dataset achieving around 0.020 MAE for all settings, 20newsgroup around 0.011 MAE, and MDS around 0.008 MAE, indicating consistent performance trends across different datasets despite variations in the absolute MAE values.

In terms of execution times, however, the situation is quite different, as shown in Tables 6.2, 6.4, 6.6, 6.8: ACC and PACC always take much longer, while the other quantification algorithms have similar testing times to CC but much longer training times. Additionally, due to the larger size of the dataset, again the execution times for MDS and 20newsgroup are generally higher than those for Karonga HDSS.

6.2 Exploiting the Hierarchical Structure

Utilizing the hierarchical structure through a classifier has proven to be somewhat beneficial: while the baseline employing a hierarchical classifier yielded results that were nearly identical to those obtained with a flat classifier, both for h(CC) and h(ACC) methods (as can be seen in Tables 6.1, 6.3 6.5 and 6.7), the hierarchical approach was frequently less computationally expensive during the training phase while maintaining comparable timing results in the testing phase (observable again in Tables 6.2, 6.4, 6.6, 6.8).

Instead, hierarchical quantification algorithms did not show any improvement with respect to the results obtained employing quantification algorithms that disregard the hierarchical nature of the labels.

There is a consistent pattern of improvement of the hierarchical algorithm H(SLD) compared to SLD which becomes notable when measured with MRAE, while when referring to the MAE, almost identical results are noted.

For the other quantifiers, there is no consistent improvement in the results; they occasionally perform better than their hierarchical versions, but generally, the values are very close, as previously mentioned. As described in Section 5.3, Kraemer's algorithm is employed in this work, an algorithm that simulates the presence of prior probability shift (i.e., the variation in prevalence values between training and testing), which is artificially varied, also testing heavily imbalanced combinations.

In this situation, the results in the previous work would lead to the expectation of H(SLD) being the best performing algorithm, followed by H(PACC), while the last ones should be H(PCC), H(ACC) and H(CC). Therefore, the results obtained in this work do not confirm other results in the literature.

Finally moving on to comment on the computational effort of each algorithm, we can observe that training typically requires less time for hierarchical quantifiers compared to other prevalence estimation algorithms.

However, during the testing phase, H(CC) is generally the fastest, followed by H(PCC) and H(SLD). The fact that ACC takes longer than other quantification algorithms, both in training and testing, is also confirmed in the hierarchical version H(ACC).

6.3 Embedding Typologies

Tables 6.1-6.8 also show the results obtained using TF-IDF vectorization compared to the results obtained using pre-trained embeddings.

Surprisingly, the values for both MAE and MRAE are very similar across all datasets.

It appears that TF-IDF representations were sufficiently complex for the task. This is likely because, as previously mentioned, all datasets were lemmatized, stop words were removed, and syntax was corrected.

However, the training times for the algorithms are significantly higher when pre-trained embeddings are exploited and SVM is the classifier.

On the other hand, with pre-trained embeddings and Logistic Regression as classifier, only H(PCC) has significantly higher times compared the results obtained with the same classifier and TF-IDF vectors.

6.4 Closed Questions, Narratives Only

The previously cited works agree that exploiting all possible information in a verbal autopsy, i.e., including both narratives and closed-ended questions in one's analysis, is beneficial for obtaining better results. This is only partially confirmed in our work.

In fact, in the concatenated version of the Karonga HDSS dataset, created by combining the column names with their respective values and adding them to the narratives, we obtained better MAE results (with a small difference) in the TF-IDF representations, while using pre-trained embeddings the narratives only dataset gave better results.

This can be accounted to the fact that TF-IDF transformation inherently relies on the frequency and uniqueness of terms within the dataset to create feature vectors. In the concatenated version of the dataset, the extra context may have introduced additional structured information that enhances the distinctiveness of the terms.

In contrast, when using pre-trained embeddings, the model benefits from a rich, pre-learned understanding of language semantics that captures meaning and context beyond mere term frequency. Therefore, the more coherent narratives-only dataset performs better with pre-trained embeddings.

Since concatenation did not show any substantial improvement in this comparison, we decided not to repeat the concatenation on the other datasets.

	TF-IDF				Pre-trained			
	SVM		LR		SVM		LR	
	MAE	MRAE	MAE	MRAE	MAE	MRAE	MAE	MRAE
CC	0.020	1.269	0.027	1.667	0.019	1.224	0.022	1.372
ACC	0.021	1.333	0.023	1.390	0.022	1.370	0.023	1.441
PCC	**0.016**	1.168	0.018	1.308	**0.017**	**1.195**	**0.016**	1.169
PACC	-	-	0.021	1.213	-	-	0.020	1.167
SLD	0.027	1.610	0.023	1.023	0.028	1.614	0.020	1.151
h(CC)	0.019	1.243	0.025	1.512	0.021	1.266	0.020	1.340
h(ACC)	0.020	1.257	0.024	1.508	0.023	1.403	0.023	1.341
H(CC)	0.019	1.241	0.025	1.504	0.020	1.267	0.022	1.342
H(ACC)	0.020	**1.109**	0.024	1.274	0.023	1.299	0.024	1.336
H(PCC)	0.021	1.371	**0.017**	1.158	0.018	1.228	0.019	1.256
H(PACC)	-	-	0.021	1.024	-	-	0.021	1.129
H(SLD)	0.023	1.163	0.027	**0.947**	0.026	1.312	0.021	**1.093**

Table 6.1: **Karonga HDSS VAs**, narratives only: table of the results (in MAE and MRAE) obtained with the prevalence estimation algorithms discusses in this work, using TF-IDF vectors or pre-trained embeddings, and comparing SVM and LR as classifiers.

	TF-IDF				Pre-trained			
	SVM		LR		SVM		LR	
	Train	Test	Train	Test	Train	Test	Train	Test
CC	0.35	**0.15**	4.69	**0.15**	62.42	0.39	4.94	**0.22**
ACC	1.69	11.86	10.26	15.86	217.53	8.60	18.82	11.86
PCC	2.61	0.21	2.98	0.18	200.82	0.42	5.64	**0.22**
PACC	-	-	12.89	39.69	-	-	18.58	12.74
SLD	2.95	0.67	2.99	0.71	188.95	0.52	**4.44**	0.44
h(CC)	0.25	0.52	5.20	0.83	45.17	**0.34**	12.53	0.52
h(ACC)	3.84	27.15	9.77	17.13	170.14	9.01	29.92	11.42
H(CC)	**0.22**	1.15	**2.57**	0.86	53.20	2.30	12.54	1.09
H(ACC)	1.31	11.47	14.84	16.10	170.85	14.70	30.04	15.75
H(PCC)	4.65	2.00	3.03	10.23	**12.57**	1.16	154.10	8.46
H(PACC)	-	-	8.99	10.18	-	-	34.87	16.95
H(SLD)	1.65	11.57	3.02	5.47	147.03	22.38	11.75	4.56

Table 6.2: **Karonga HDSS VAs**, narratives only: table of the training and test running times of the prevalence estimation algorithms discusses in this work.

	TF-IDF				Pre-trained			
	SVM		LR		SVM		LR	
	MAE	MRAE	MAE	MRAE	MAE	MRAE	MAE	MRAE
CC	0.017	1.110	0.026	1.580	0.029	1.779	0.033	1.996
ACC	0.020	1.195	0.025	1.387	0.029	1.790	0.028	1.834
PCC	**0.014**	1.044	0.018	1.282	**0.018**	**1.339**	**0.020**	**1.400**
PACC	-	-	0.020	1.179	-	-	0.028	1.735
SLD	0.024	1.405	0.022	1.023	0.034	2.001	0.032	1.908
h(CC)	0.019	1.096	0.024	1.462	0.029	1.741	0.030	1.806
h(ACC)	0.018	1.078	0.022	1.347	0.029	1.742	0.028	1.731
H(CC)	0.018	1.097	0.024	1.456	0.031	1.740	0.030	1.804
H(ACC)	0.017	**0.942**	0.022	1.288	0.031	1.638	0.032	1.819
H(PCC)	0.020	1.313	**0.014**	0.972	0.026	1.693	0.025	1.635
H(PACC)	-	-	0.021	**0.970**	-	-	0.032	1.838
H(SLD)	0.021	1.021	0.027	0.996	0.033	1.692	0.033	1.834

Table 6.3: **Karonga HDSS VAs**, concatenated: table of the results as previously described in Table 6.1, leveraging the concatenation of both narratives and closed questions.

	TF-IDF				Pre-trained			
	SVM		LR		SVM		LR	
	Train	Test	Train	Test	Train	Test	Train	Test
CC	1.91	**0.16**	7.39	0.16	210.80	**0.23**	**4.22**	0.27
ACC	4.71	10.10	15.68	12.89	411.18	9.56	8.07	8.92
PCC	8.07	0.21	7.07	**0.15**	770.34	0.26	4.24	**0.22**
PACC	-	-	15.73	13.05	-	-	9.22	13.47
SLD	7.85	0.45	7.32	0.60	767.00	0.56	4.39	0.71
h(CC)	0.93	0.50	8.79	0.59	80.30	0.38	11.86	0.36
h(ACC)	2.98	12.75	12.34	13.48	159.32	10.07	16.50	12.59
H(CC)	**0.90**	1.10	8.26	1.01	81.21	1.09	11.77	0.92
H(ACC)	2.72	10.17	13.03	9.55	166.02	9.89	17.34	7.88
H(PCC)	8.29	1.06	**4.36**	5.94	**11.82**	0.92	302.74	6.96
H(PACC)	-	-	13.46	9.76	-	-	18.87	11.34
H(SLD)	4.36	8.84	8.22	5.47	301.70	13.61	10.10	11.73

Table 6.4: **Karonga HDSS VAs**, concatenated: table of the timings obtained as described in Table 6.2, leveraging the concatenation of both narratives and closed questions.

	TF-IDF				Pre-trained			
	SVM		LR		SVM		LR	
	MAE	MRAE	MAE	MRAE	MAE	MRAE	MAE	MRAE
CC	0.008	0.799	0.010	0.958	0.008	0.805	0.009	0.904
ACC	0.008	0.803	0.009	0.894	0.009	0.847	0.009	0.901
PCC	**0.007**	0.807	0.008	0.871	**0.007**	0.839	**0.008**	0.841
PACC	-	-	0.009	0.813	-	-	0.009	0.834
SLD	0.008	0.744	0.008	**0.722**	0.009	0.861	0.009	0.839
h(CC)	0.008	0.796	0.009	0.927	0.008	0.820	**0.008**	0.859
h(ACC)	0.008	0.825	0.009	0.883	0.009	0.850	0.009	0.879
H(CC)	0.008	0.791	0.009	0.926	0.008	0.811	0.008	0.857
H(ACC)	0.009	0.786	0.009	0.818	0.009	0.845	0.009	0.851
H(PCC)	0.008	0.863	**0.007**	0.777	0.008	0.832	**0.008**	0.846
H(PACC)	-	-	0.009	0.781	-	-	0.009	**0.803**
H(SLD)	0.010	**0.716**	0.010	0.747	0.011	**0.795**	0.009	0.809

Table 6.5: **Million-Death Study VAs**, narratives only: table of the results as previously described in Table 6.1, leveraging narratives only.

	TF-IDF				Pre-trained			
	SVM		LR		SVM		LR	
	Train	Test	Train	Test	Train	Test	Train	Test
CC	2.84	0.67	37.83	**0.80**	239.03	**0.65**	24.31	**0.54**
ACC	15.58	246.92	116.49	301.17	1006.64	180.03	106.83	245.32
PCC	14.22	**0.53**	25.83	0.77	923.41	0.69	24.99	0.87
PACC	-	-	124.21	279.98	-	-	105.44	329.26
SLD	13.77	0.82	25.29	**0.80**	786.03	0.92	**21.31**	0.66
h(CC)	**0.84**	1.21	9.79	1.33	120.55	1.13	30.80	0.80
h(ACC)	7.00	300.25	45.15	289.89	514.16	181.49	77.16	209.41
H(CC)	2.45	3.57	9.38	2.80	118.46	2.35	31.90	2.31
H(ACC)	6.48	27.00	45.44	48.28	505.59	27.85	77.86	28.35
H(PCC)	7.65	3.61	**7.34**	16.89	**32.03**	2.42	462.10	19.38
H(PACC)	-	-	39.06	29.98	-	-	81.92	33.39
H(SLD)	5.85	30.67	8.85	16.45	475.43	31.57	31.29	13.26

Table 6.6: **Million-Death Study VAs**, narratives: table of the timings obtained as described in Table 6.2, leveraging leveraging narratives only.

	TF-IDF				Pre-trained			
	SVM		LR		SVM		LR	
	MAE	MRAE	MAE	MRAE	MAE	MRAE	MAE	MRAE
CC	**0.010**	0.382	**0.012**	0.439	**0.012**	0.465	0.015	0.581
ACC	0.011	0.361	**0.012**	**0.394**	0.014	0.400	0.017	0.485
PCC	0.013	0.558	0.023	1.088	0.017	0.738	0.017	0.730
SLD	0.011	**0.277**	0.023	0.476	0.014	**0.351**	**0.014**	**0.367**
PACC	-	-	0.014	0.436	-	-	**0.014**	0.390
h(CC)	0.012	0.435	0.015	0.554	0.014	0.538	0.016	0.609
h(ACC)	0.011	0.393	0.013	0.441	0.015	0.443	0.017	0.488
H(CC)	0.021	0.622	0.024	0.743	0.022	0.720	0.023	0.793
H(ACC)	0.020	0.575	0.022	0.646	0.022	0.612	0.024	0.674
H(PCC)	0.027	1.048	0.020	0.676	0.023	0.884	0.023	0.844
H(PACC)	-	-	0.022	0.630	-	-	0.023	0.620
H(SLD)	0.020	0.465	0.030	0.575	0.023	0.531	0.023	0.570

Table 6.7: **20newsgroup**: table of the results as previously described in Table 6.1.

	TF-IDF				Pre-trained			
	SVM		LR		SVM		LR	
	Train	Test	Train	Test	Train	Test	Train	Test
CC	1.51	**0.09**	16.15	0.10	117.87	**0.16**	**6.41**	0.17
ACC	3.67	3.92	26.82	4.12	223.50	2.83	13.55	2.54
PCC	7.13	0.19	13.99	**0.09**	475.38	0.19	6.96	**0.16**
PACC	-	-	27.30	4.59	-	-	12.04	3.24
SLD	7.14	0.23	13.98	0.20	473.13	0.24	7.85	0.24
h(CC)	1.15	2.08	16.55	4.10	105.01	0.58	13.13	0.60
h(ACC)	3.08	5.98	26.19	7.57	194.49	3.19	21.86	3.31
H(CC)	**1.10**	0.58	15.21	0.85	105.20	0.44	13.46	0.50
H(ACC)	6.65	6.45	26.85	6.49	198.52	6.24	23.90	6.66
H(PCC)	16.14	0.90	**5.02**	2.75	**16.13**	0.46	409.05	2.77
H(PACC)	-	-	26.95	6.53	-	-	25.95	6.32
H(SLD)	4.87	2.96	16.14	1.70	409.33	2.98	13.27	0.97

Table 6.8: **20newsgroup**: table of the timings obtained as previously described in Table 6.2.

Chapter 7

Conclusion

In this concluding section we summarize the key findings of this research and reflect on the overall contributions and limitations of the study. The guidelines of this work were three research questions, respectively whether a real quantification algorithm would have performed better than a Classify and Count approach (i), and whether exploiting the codeframe hierarchy would have brought a benefit in terms of accuracy of the prevalence estimates (ii) or of computational resources (iii).

The work started from the knowledge of the fact that quantification methods could be better for prior-shift settings. Surprisingly, many among the supposedly more sophisticated quantification methods often fail to improve over CC's performance (i). Even the most sophisticated quantification method (SLD) does not stand as the top performer, while interestingly PCC is the algorithm that obtains the best results in most of the settings. Sporadic benefits have been observed by exploiting the hierarchical version of the proposed algorithms, but we have not obtained the net benefit we expected, neither in terms of better quantification (ii) nor in terms of improved efficiency (iii). However, utilizing a hierarchical classification algorithm instead of a flat one yielded interesting outcomes, as it produced the same results in terms of MAE and MRAE with a shorter training time.

In any case, this work marks the beginning of the discussion regarding hierarchical quantification. The logical evidence of the wisdom of an approach *divide et impera*, that is, thinking about exploiting a (hierarchical) grouping of classes in problems with a large or very large set of classes, nevertheless continues to motivate the need for further investigation.

The comparison between results obtained using TF-IDF representations and pre-trained embeddings also yielded unexpected outcomes, showing no significant difference in terms of MAE and MRAE, and this similarity fails to justify the computational expense associated with utilizing pre-trained embeddings.

7.1 Future Work

Our results are not totally conclusive, as they do not agree with the experimental evidence provided in other previous works. Therefore, more in-depth experimentation would be necessary.

It would be useful to check the quantification accuracy of our systems at level 1 (i.e., the level of the children of the root), at level 2 (grandchildren of the root) and so on, checking all the nodes in the hierarchy of the classes. In other terms, we may want to run level-wise, single-label multiclass evaluations, with the goal of rewarding a system that tends to misclassify a datapoint into "close" classes (i.e., classes hierarchically close to the correct ones) over a system that misclassifies datapoints into "faraway" classes.

Moreover, although to improve the usability or performance of CCVA systems was not among our goals, our models have the notable benefit that they do not depend on the structure of a specific questionnaire and allow narratives to be exploited without specific ad-hoc manipulations. In any case, to affirm that the methods proposed in this work can improve the automated assignment of the CoD, an extensive comparison with existing methods would be necessary, which however have (in almost all cases) the objective of improving the accuracy of the classification of the single document and not of the estimate of the distribution of causes in the dataset.

Besides that, to obtain the results in this work, the UPP evaluation protocol was used, which creates an artificially marked situation of dataset shift. It could be interesting to verify the result of the algorithms on a dataset of VAs in conditions of "natural" imbalance, with the aim of verifying the usability of the proposed methods in a real application in epidemiology.

Bibliography

Aleksandrowicz, L., Malhotra, V., Dikshit, R., Gupta, P. C., Kumar, R., Sheth, J., Rathi, S. K., Suraweera, W., Miasnikof, P., Jotkar, R., Sinha, D., Awasthi, S., Bhatia, P., and Jha, P. (2014). Performance criteria for verbal autopsy-based systems to estimate national causes of death: development and application to the Indian Million Death Study. *BMC Medicine*, 12(1):21.

Anker, M. (1997). The effect of misclassification error on reported cause-specific mortality fractions from verbal autopsy. *International Journal of Epidemiology*, 26(5):1090–1096.

Araya, T., Tensou, B., Davey, G., and Berhane, Y. (2011). Burial surveillance detected significant reduction in HIV-related deaths in Addis Ababa, Ethiopia. *Tropical Medicine & International Health*, 16(12):1483–1489.

Astikainen, K., Holm, L., Pitkänen, E., Szedmak, S., and Rousu, J. (2008). Towards structured output prediction of enzyme function. *BMC Proceedings*, 2(4):S2.

Bailo, P., Gibelli, F., Ricci, G., and Sirignano, A. (2022). Verbal Autopsy as a Tool for Defining Causes of Death in Specific Healthcare Contexts: Study of Applicability through a Traditional Literature Review. *International Journal of Environmental Research and Public Health*, 19(18):11749.

Bella, A., Ferri, C., Hernández-Orallo, J., and Ramírez-Quintana, M. J. (2010). Quantification via probability estimators. In *Proceedings of the 11th IEEE International Conference on Data Mining (ICDM 2010)*, pages 737–742, Sydney, AU.

Biraud, Y. (1956). Méthodes pour l'enregistrement par des non médecins des causes élémentaires de décès dans les zones sous-développées [methods for registration of elementary causes of death by non-medically trained workers in developing countries]. *Geneva: World Health Organization; 1956. WHO document HS/60.*

Blanco, A., Pérez, A., and Casillas, A. (2022). Exploiting icd hierarchy for classification of ehrs in spanish through multi-task transformers. *IEEE Journal of Biomedical and Health Informatics*, 26(3):1374–1383.

Blanco, A., Pérez, A., Casillas, A., and Cobos, D. (2021). Extracting cause of death from verbal autopsy with deep learning interpretable methods. *IEEE Journal of Biomedical and Health Informatics*, 25(4):1315–1325.

Boulle, A., Chandramohan, D., and Weller, P. (2001). A case study of using artificial neural networks for classifying cause of death from verbal autopsy. *International Journal of Epidemiology*, 30(3):515–520.

Brown, P. E., Izawa, Y., Balakrishnan, K., Fu, S. H., Chakma, J., Menon, G., Dikshit, R., Dhaliwal, R., Rodriguez, P. S., Huang, G., Begum, R., Hu, H., D'Souza, G., Guleria, R., and Jha, P. (2022). Mortality Associated with Ambient PM2.5 Exposure in India: Results from the Million Death Study. *Environmental Health Perspectives*, 130(9):097004.

Bunse, M. (2022). On multi-class extensions of adjusted classify and count. In *Proceedings of the 2nd International Workshop on Learning to Quantify (LQ 2022)*, pages 43–50, Grenoble, IT.

Byass, P., Chandramohan, D., Clark, S. J., D'Ambruoso, L., Fottrell, E., Graham, W. J., Herbst, A. J., Hodgson, A., Hounton, S., Kahn, K., Krishnan, A., Leitao, J., Odhiambo, F., Sankoh, O. A., and Tollman, S. M. (2012). Strengthening standardised interpretation of verbal autopsy data: the new InterVA-4 tool. *Global Health Action*, 5(1):19281.

Byass, P., Dao Lan Huong, and Hoang Van Minh (2003). A probabilistic approach to interpreting verbal autopsies: methodology and preliminary validation in Vietnam. *Scandinavian Journal of Public Health*, 31(62_suppl):32–37.

Byass, P., Hussain-Alkhateeb, L., D'Ambruoso, L., Clark, S., Davies, J., Fottrell, E., Bird, J., Kabudula, C., Tollman, S., Kahn, K., Schiöler, L., and Petzold, M. (2019). An integrated approach to processing WHO-2016 verbal autopsy data: the InterVA-5 model. *BMC Medicine*, 17(1):102.

Ceci, M. and Malerba, D. (2007). Classifying web documents in a hierarchy of categories: a comprehensive study. *Journal of Intelligent Information Systems*, 28(1):37–78.

Chandramohan, D., Maude, G. H., Rodrigues, L. C., and Hayes, R. J. (1998). Verbal autopsies for adult deaths: their development and validation in a multicentre study. *Tropical Medicine & International Health*, 3(6):436–446.

Cheng, C.-H., Tang, J., Wai-Chee, A., and King, I. (2001). Hierarchical classification of documents with error control. In *Proceedings of the 5th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD 2001)*, pages 433–443, Hong Kong, CN.

Crampin, A. C. and Dube, A. (2003-2017). Malawi - karonga hdss indepth core dataset 2003-2017 (release 2019). `https://doi.org/10.7796/INDEPTH.MW011.CMD2017.v1`.

Danso, S., Atwell, E., and Johnson, O. (2013a). Linguistic and statistically derived features for cause of death prediction from verbal autopsy text. In Gurevych, I., Biemann, C., and Zesch, T., editors, *Language Processing and Knowledge in the Web*, pages 47–60, Berlin, Heidelberg. Springer Berlin Heidelberg.

Danso, S. O., Atwell, E., and Johnson, O. (2013b). A comparative study of machine learning methods for verbal autopsy text classification. *IJCSI International Journal of Computer Science Issues*, 10.

De Souza, P. M. M., Gerson, G., Dias, J. S., De Melo, D. N., De Souza, S. G., Ruiz, E. M., Fernandes Tavora, F. R., and Cavalcanti, L. P. D. G. (2020). Validation of verbal autopsy and nasopharyngeal swab collection for the investigation of deaths at home during the COVID-19 pandemics in Brazil. *PLOS Neglected Tropical Diseases*, 14(11):e0008830.

Deressa, W., Fantahun, M., and Ali, A. (2007). Malaria-related mortality based on verbal autopsy in an area of low endemicity in a predominantly rural population in Ethiopia. *Malaria Journal*, 6(1):128.

Dikshit, R., Gupta, P. C., Ramasundarahettige, C., Gajalakshmi, V., Aleksandrowicz, L., Badwe, R., Kumar, R., Roy, S., Suraweera, W., Bray, F., Mallath, M., Singh, P. K., Sinha, D. N., Shet, A. S., Gelband, H., and Jha, P. (2012). Cancer mortality in India: a nationally representative survey. *The Lancet*, 379(9828):1807–1816.

Esuli, A., Fabris, A., Moreo, A., and Sebastiani, F. (2023). *Learning to quantify*. Springer Nature, Cham, CH.

Esuli, A., Fagni, T., and Sebastiani, F. (2008). Boosting multi-label hierarchical text categorization. *Information Retrieval*, 11(4):287–313.

Esuli, A., Moreo, A., and Sebastiani, F. (2022a). LeQua@CLEF2022: Learning to Quantify. In *Proceedings of the 44th European Conference on Information Retrieval (ECIR 2022)*, pages 374–381, Stavanger, NO.

Esuli, A., Moreo, A., Sebastiani, F., and Sperduti, G. (2022b). A detailed overview of LeQua 2022: Learning to quantify. In *Working Notes of the 13th Conference and Labs of the Evaluation Forum (CLEF 2022)*, Bologna, IT.

Fernandes Vaz, A., Izbicki, R., and Bassi Stern, R. (2019). Quantification under prior probability shift: The ratio estimator and its extensions. *Journal of Machine Learning Research*, 20:79:1–79:33.

Flaxman, A. D., Serina, P. T., Hernandez, B., Murray, C. J. L., Riley, I., and Lopez, A. D. (2015). Measuring causes of death in populations: a new metric that corrects cause-specific mortality fractions for chance. *Population Health Metrics*, 13(1):28.

for Health Metrics, I. and application., E. S.-A. (2024). https://www.healthdata.org/data-tools-practices/verbal-autopsy.

Forman, G. (2005). Counting positives accurately despite inaccurate classification. In *Proceedings of the 16th European Conference on Machine Learning (ECML 2005)*, pages 564–575, Porto, PT.

Forman, G. (2006). Quantifying trends accurately despite classifier error and class imbalance. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2006)*, pages 157–166, Philadelphia, US.

Forman, G. (2008). Quantifying counts and costs via classification. *Data Mining and Knowledge Discovery*, 17(2):164–206.

Gao, W. and Sebastiani, F. (2016). From classification to quantification in tweet sentiment analysis. *Social Network Analysis and Mining*, 6(19):1–22.

Gemechu, T., Tinsae, M., Ashenafi, S., Rodriguez, V. M., Lori, A., Collins, M., Hurford, R., Haimanot, R., Sandoval, M., Mehari, E., and Langford, T. D. (2009). Most common causes of natural and injury-related deaths in Addis Ababa, Ethiopia. *Pathology - Research and Practice*, 205(9):608–614.

Glynn, J. R., Calvert, C., Price, A., Chihana, M., Kachiwanda, L., Mboma, S., Zaba, B., and Crampin, A. C. (2014). Measuring causes of adult mortality in rural northern Malawi over a decade of change. *Global Health Action*, 7(1):23621.

Hart, J. D., De André, P. A., De André, C. D. S., Adair, T., Barroso, L. P., Valongueiro, S., Bierrenbach, A. L., De Carvalho, P. I., Antunes, M. B. D. C., De Oliveira, C. M., Pereira, L. A. A., Minto, C. M., Bezerra, T. M. D. S., Costa, S. P., De Azevedo, B. A., De Lima, J. R. A., Mota, D. S. D. M., Ramos, A. M. D. O., De Souza, M. D. F. M., Da Silva, L. F. F., França, E. B., McLaughlin, D., Riley, I. D., and Saldiva, P. H. N. (2022). Validation of SmartVA using conventional autopsy: A study of adult deaths in Brazil. *The Lancet Regional Health - Americas*, 5:100081.

James, S. L., Flaxman, A. D., and Murray, C. J. (2011). Performance of the Tariff Method: validation of a simple additive algorithm for analysis of verbal autopsies. *Population Health Metrics*, 9(1):31.

Jana, S., Fu, S. H., Gelband, H., Brown, P., and Jha, P. (2022). Spatio-temporal modelling of malaria mortality in India from 2004 to 2013 from the Million Death Study. *Malaria Journal*, 21(1):90.

Jeblee, S., Gomes, M., Jha, P., Rudzicz, F., and Hirst, G. (2019). Automatically determining cause of death from verbal autopsy narratives. *BMC Medical Informatics and Decision Making*, 19(1):127.

Jerzak, C. T., King, G., and Strezhnev, A. (2023). An improved method of automated nonparametric content analysis for social science. *Political Analysis*, 31(1):42–58.

Kalter, H. D., Perin, J., and Black, R. E. (2016). Validating hierarchical verbal autopsy expert algorithms in a large data set with known causes of death. *Journal of Global Health*, 6(1):010601.

Ke, C., Gupta, R., Shah, B. R., Stukel, T. A., Xavier, D., and Jha, P. (2021). Association of Hypertension and Diabetes with Ischemic Heart Disease and Stroke Mortality in India: The Million Death Study. *Global Heart*, 16(1):69.

King, G. and Lu, Y. (2008). Verbal autopsy methods with multiple causes of death. *Statistical Science*, 23(1):78–91.

King, G., Lu, Y., and Shibuya, K. (2010). Designing verbal autopsy studies. *Population Health Metrics*, 19(8).

Koller, D. and Sahami, M. (1997). Hierarchically classifying documents using very few words. In *Proceedings of the 14th International Conference on Machine Learning (ICML 1997)*, pages 170–178, Nashville, US.

Leitao, J., Desai, N., Aleksandrowicz, L., Byass, P., Miasnikof, P., Tollman, S., Alam, D., Lu, Y., Rathi, S. K., Singh, A., Suraweera, W., Ram, F., and Jha, P. (2014). Comparison of physician-certified verbal autopsy with computer-coded verbal autopsy for cause of death assignment in hospitalized patients in low- and middle-income countries: systematic review. *BMC Medicine*, 12(1):22.

Li, Z. R., Thomas, J., Choi, E., McCormick, T. H., and Clark, S. J. (2022). The openva toolkit for verbal autopsies.

Lozano, R., Lopez, A. D., Atkinson, C., Naghavi, M., Flaxman, A. D., and Murray, C. J. (2011). Performance of physician-certified verbal autopsies: multisite validation study using clinical diagnostic gold standards. *Population Health Metrics*, 9(1):32.

Mahesh, B. P. K., Hart, J. D., Acharya, A., Chowdhury, H. R., Joshi, R., Adair, T., and Hazard, R. H. (2022). Validation studies of verbal autopsy methods: a systematic review. *BMC Public Health*, 22(1):2215.

Manaka, T., van Zyl, T., and Kar, D. (2022). Improving Cause-of-Death Classification from Verbal Autopsy Reports. unpublished.

McCormick, T. H., Li, Z. R., Calvert, C., Crampin, A. C., Kahn, K., and Clark, S. J. (2016). Probabilistic Cause-of-Death Assignment Using Verbal Autopsies. *Journal of the American Statistical Association*, 111(515):1036–1049.

Menéndez, C., Quintó, L., Castillo, P., Carrilho, C., Ismail, M. R., Lorenzoni, C., Fernandes, F., Hurtado, J. C., Rakislova, N., Munguambe, K., Maixenchs, M., Macete, E., Mandomando, I., Martínez, M. J., Bassat, Q., Alonso, P. L., and Ordi, J. (2021). Limitations to current methods to estimate cause of death: a validation study of a verbal autopsy model. *Gates Open Research*, 4:55.

Miasnikof, P., Giannakeas, V., Gomes, M., Aleksandrowicz, L., Shestopaloff, A. Y., Alam, D., Tollman, S., Samarikhalaj, A., and Jha, P. (2015). Naive Bayes classifiers for verbal autopsies: comparison to physician-based classification for 21,000 child and adult deaths. *BMC Medicine*, 13(1):286.

Misganaw, A., Mariam, D. H., Araya, T., and Aneneh, A. (2012). Validity of verbal autopsy method to determine causes of death among adults in the urban setting of Ethiopia. *BMC Medical Research Methodology*, 12(1):130.

Moreo, A., Esuli, A., and Sebastiani, F. (2021). Quapy.

Moreo, A., Francisco, M., and Sebastiani, F. (2023). Multi-label quantification. *ACM Transactions on Knowledge Discovery and Data*, 18(1):Article 4.

Moreo, A. and Sebastiani, F. (2021). Re-assessing the "classify and count" quantification method. In *Proceedings of the 43rd European Conference on Information Retrieval (ECIR 2021)*, volume II, pages 75–91, Lucca, IT.

Murray, C. J., Lopez, A. D., Black, R., Ahuja, R., Ali, S. M., Baqui, A., Dandona, L., Dantzer, E., Das, V., Dhingra, U., Dutta, A., Fawzi, W., Flaxman, A. D., Gómez, S., Hernández, B., Joshi, R., Kalter, H., Kumar, A., Kumar, V., Lozano, R., Lucero, M., Mehta, S., Neal, B., Ohno, S. L., Prasad, R., Praveen, D., Premji, Z., Ramírez-Villalobos, D., Remolador, H., Riley, I., Romero, M., Said, M., Sanvictores, D., Sazawal, S., and Tallo, V. (2011a). Population

Health Metrics Research Consortium gold standard verbal autopsy validation study: design, implementation, and development of analysis datasets. *Population Health Metrics*, 9(1):27.

Murray, C. J., Lozano, R., Flaxman, A. D., Vahdatpour, A., and Lopez, A. D. (2011b). Robust metrics for assessing the performance of different verbal autopsy cause assignment methods in validation studies. *Population Health Metrics*, 9(1):28.

Nichols, E. K., Byass, P., Chandramohan, D., Clark, S. J., Flaxman, A. D., Jakob, R., Leitao, J., Maire, N., Rao, C., Riley, I., Setel, P. W., and on behalf of the WHO Verbal Autopsy Working Group (2018). The WHO 2016 verbal autopsy instrument: An international standard suitable for automated analysis by InterVA, InSilicoVA, and Tariff 2.0. *PLOS Medicine*, 15(1):e1002486.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Pham, B. N., Abori, N., Maraga, S., Jorry, R., Jaukae, G. S., Silas, V. D., Aga, T., Okely, T., and Pomat, W. (2023). Validating the InterVA-5 cause of death analytical tool: using mortality data from the Comprehensive Health and Epidemiological Surveillance System in Papua New Guinea. *BMJ Open*, 13(5):e066560.

Ruiz, M. and Srinivasan, P. (2002). Hierarchical text categorization using neural networks. *Information Retrieval*, 5(1):87–118.

Saerens, M., Latinne, P., and Decaestecker, C. (2002). Adjusting the outputs of a classifier to new a priori probabilities: A simple procedure. *Neural Computation*, 14(1):21–41.

Sanh, V., Debut, L., Chaumond, J., and Wolf, T. (2019). Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *ArXiv*, abs/1910.01108.

Seeger, M. (2008). Cross-validation optimization for large scale structured classification kernel methods. *Journal of Machine Learning Research*, 9:1147–1178.

Serina, P., Riley, I., Stewart, A., James, S. L., Flaxman, A. D., Lozano, R., Hernandez, B., Mooney, M. D., Luning, R., Black, R., Ahuja, R., Alam, N., Alam, S. S., Ali, S. M., Atkinson, C., Baqui, A. H., Chowdhury, H. R., Dandona, L., Dandona, R., Dantzer, E., Darmstadt, G. L., Das, V., Dhingra, U., Dutta, A., Fawzi, W., Freeman, M., Gomez, S., Gouda, H. N., Joshi, R., Kalter, H. D., Kumar, A., Kumar, V., Lucero, M., Maraga, S., Mehta, S., Neal, B., Ohno,

S. L., Phillips, D., Pierce, K., Prasad, R., Praveen, D., Premji, Z., Ramirez-Villalobos, D., Rarau, P., Remolador, H., Romero, M., Said, M., Sanvictores, D., Sazawal, S., Streatfield, P. K., Tallo, V., Vadhatpour, A., Vano, M., Murray, C. J. L., and Lopez, A. D. (2015). Improving performance of the Tariff Method for assigning causes of death to verbal autopsies. *BMC Medicine*, 13(1):291.

Silla, C. N. and Freitas, A. A. (2010). A survey of hierarchical classification across different application domains. *Data Mining and Knowledge Discovery*, 22(1/2):31–72.

Smith, N. A. and Tromble, R. W. (2004). Sampling uniformly from the unit simplex. Technical report, Johns Hopkins University. `https://www.cs.cmu.edu/~nasmith/papers/smith+tromble.tr04.pdf`.

Steindel, S. J. (2010). International classification of diseases, 10th edition, clinical modification and procedure coding system: descriptive overview of the next generation HIPAA code sets. *Journal of the American Medical Informatics Association*, 17(3):274–282.

Sun, A. and Lim, E.-P. (2001). Hierarchical text classification and evaluation. In *Proceedings of the 1st IEEE International Conference on Data Mining (ICDM 2001)*, pages 521–528, San Jose, US.

Tunga, M., Lungo, J., Chambua, J., and Kateule, R. (2021). Verbal autopsy models in determining causes of death. *Tropical Medicine & International Health*, 26(12):1560–1567.

Van Eijk, A. M., Adazu, K., Ofware, P., Vulule, J., Hamel, M., and Slutsker, L. (2008). Causes of deaths using verbal autopsy among adolescents and adults in rural western Kenya. *Tropical Medicine & International Health*, 13(10):1314–1324.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS 2017)*, pages 5998–6008, Long Beach, US.

Wu, F., Chen, Y., Parvez, F., Segers, S., Argos, M., Islam, T., Ahmed, A., Rakibuz-Zaman, M., Hasan, R., Sarwar, G., and Ahsan, H. (2013). A Prospective Study of Tobacco Smoking and Mortality in Bangladesh. *PLoS ONE*, 8(3):e58516.