

©Copyright 2020

Karishma Mandyam

Semi-Supervised Learning for Verbal Autopsy

Karishma Mandyam

A thesis
submitted in partial fulfillment of the
requirements for the degree of

Masters in Computer Science and Engineering

University of Washington

2020

Reading Committee:

Noah A. Smith, Chair

Program Authorized to Offer Degree:
UW CSE M.S. Program

ACKNOWLEDGMENTS

This work would not have been possible without the guidance of Suchin Gururangan. Thank you for teaching me how to approach NLP research by first designing the high-level questions and then the detailed experiments. Most importantly, thank you for showing me how to persevere when experimental results don't support the hypotheses, and how to reframe the research question as an important contribution. I'm very grateful for your mentorship.

I'd also like to express my sincere gratitude to Noah Smith for giving me the opportunity to pursue NLP research as an undergraduate and master's student with the ARK research group. Thank you for teaching me how to critically read papers and think independently about research.

Thank you to Tyler McCormick and Brandon Stewart for the guidance and support on this project.

Thank you to everyone in the ARK research group and at UW NLP who have taken the time to mentor me over the past few years, especially Swabha Swayamdipta, Yangfeng Ji, and Yejin Choi.

Thanks to my fellow ARK undergraduate and master's students, Sam Gehman, Tam Dang, Leo Liu, Michael Zhang, and Nelson Liu. Learning NLP with you all has been the greatest joy.

Finally, thank you Mom, Dad, Pati, and Aishwarya. Your support means the world to me.

TABLE OF CONTENTS

	Page
Chapter 1: Introduction	1
1.1 Contributions	3
Chapter 2: Data	5
2.1 Verbal Autopsy Dataset	5
2.2 Verbal Autopsy Survey Content	5
2.3 Re-Grouping Labels	6
Chapter 3: Existing Methods in Verbal Autopsy	8
3.1 Tariff	9
3.2 InterVA	9
3.3 InSilicoVA	10
3.4 NBC	10
3.5 Simple Neural Classifier (CLF)	11
3.6 Current Approaches to Data Settings	11
Chapter 4: Using Unlabeled Data	12
4.1 VAMPIRE	12
4.1.1 Pre-training with VAMPIRE	13
4.1.2 Incorporating Pre-trained VAE in Downstream Text Classification	14
4.2 RoBERTa and BioMed RoBERTa	14
4.2.1 RoBERTa	14
4.2.2 DAPT + TAPT and BioMed RoBERTa	15
Chapter 5: Domain	16
Chapter 6: Experimental Setup	17
6.1 Motivation	17
6.2 Splitting the Data	17
6.3 Interpreting VA Data for VAMPIRE (and other neural models)	18

6.4	Evaluation Metrics	19
6.5	Constructing Data Resource Settings	19
6.5.1	No Target Domain Data	20
6.5.2	Few Target Domain Data	21
6.5.3	More Target Domain Data	21
Chapter 7:	Evaluating Supervised Methods	22
7.1	Training Details	22
7.2	Results	22
7.2.1	No Target Domain Data	22
7.2.2	Few Target Domain Data	23
7.2.3	More Target Domain Data	24
Chapter 8:	VAMPIRE Performance	25
8.1	Training Details	25
8.2	No Target Domain Data	25
8.3	Few Target Domain Data	26
8.4	More Target Domain Data	27
Chapter 9:	Stacking	29
9.1	Related Work	29
9.2	Stacking Method	29
9.3	Stacked Training Details	30
9.3.1	Building the Augmented Dataset	30
9.3.2	Data Settings and Cross-Validation with Stacking	30
9.3.3	Hyperparameters	31
9.4	Stacked VAMPIRE	31
9.4.1	Stacked VAMPIRE Variants	32
9.5	Stacked VAMPIRE Results	32
9.5.1	No Target Domain Data	32
9.5.2	Few Target Domain Data	33
9.5.3	More Target Domain Data	34
Chapter 10:	Larger Pre-trained Models	35
10.1	Motivation	35
10.2	Experiment Details	35

10.3 Training Details	36
10.4 Results	36
10.5 Analysis	38
Chapter 11: Conclusion	39
11.1 No Target Domain Data	39
11.2 Few Target Domain Data	39
11.3 More Target Domain Data	39
11.4 Discussion	40
Chapter 12: Future Work	41

University of Washington

Abstract

Semi-Supervised Learning for Verbal Autopsy

Karishma Mandyam

Chair of the Supervisory Committee:

Professor Noah A. Smith

Paul G. Allen School of Computer Science and Engineering

Semi-supervised approaches have been widely successful for both high and low resource tasks. In this work, we explore these approaches for a low resource task, Verbal Autopsy (VA). Verbal Autopsy (VA) is a method of determining an individual's cause of death (COD) from information about their symptoms and circumstances, collected through a survey. We frame this task as a text classification task by using a natural language bag-of-words technique to interpret the VA survey data as text. Our work shows that incorporating unlabeled data and pretraining techniques with VAMPIRE (Gururangan et al., 2019) improves performance over current statistical and probabilistic methods. We also find that in some scenarios, ensembling current approaches with pretraining techniques is beneficial. Overall, we demonstrate a lower bound of semi-supervised methods for Verbal Autopsy, which in all data settings we consider, outperforms current methods.

Chapter 1

INTRODUCTION

Verbal Autopsy (VA) is a method of determining an individual's cause of death (COD) from information about their symptoms and circumstances. This information is collected through a Verbal Autopsy survey, administered by trained interviewers in locations around the world where most deaths are otherwise undocumented. The respondents of the survey are family and friends close to the recently deceased individual. They answer specific questions about the individual's behavior, symptoms, and circumstances prior to their death, such as 'did the decedent experience memory loss?', 'was the decedent paralyzed in any way?', 'did the decedent suffer a bite or sting?', etc. Typically, there are three parts of the VA survey, illustrated in Figure 1.1: demographic information of the recently deceased and survey respondent, answers to the VA questions, and open narrative text. The Verbal Autopsy task is to then determine the decedent's cause of death from a list of 34 categories. The likely cause of death is usually determined by health professionals or computer algorithms.

Verbal autopsies are especially important for scientists to track disease patterns and for public health officials to issue appropriate policies. Unfortunately, in most resource-limited environments, performing medical autopsies is extremely expensive and inefficient. As a result, there are few labeled data points in Verbal Autopsy datasets. The result is a larger amount of unlabeled data points.

Verbal Autopsy is a health research and social science challenge not yet studied in natural language processing research. The current methods, explored in Chapter 3, are supervised, statistical and probabilistic models such as InterVA (Byass et al., 2012; Fottrell et al., 2007), which rely on physician provided probabilities, called symptom-cause information (SCI), to associate symptoms and causes of death. These methods do not use the open narrative text portion of the Verbal Autopsy dataset. The open narrative text documents keywords related to medicine and autopsies that appear throughout the conversation during a VA interview. For example, if "blood" appeared anywhere in

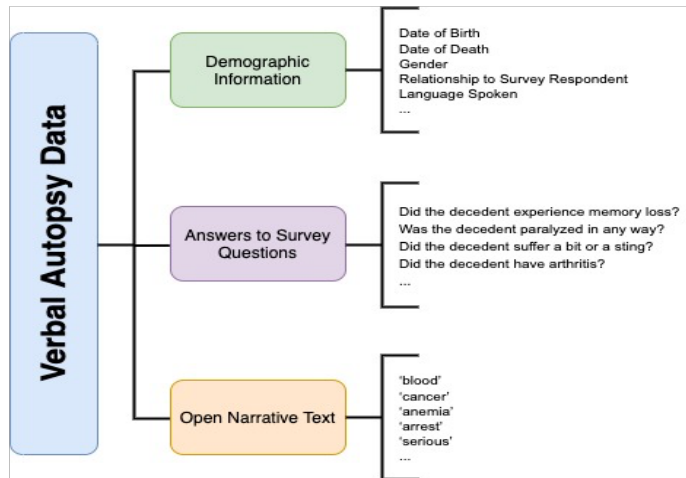


Figure 1.1: The three parts of the Verbal Autopsy dataset

the VA interview, it would be added to the list of open narrative words. This narrative text is essentially a bag of words; each word is not tied to a particular question and the order of the words does not matter. This makes it difficult for statistical models to capture specific symptom-cause information from them. In the earlier example, it’s hard to categorize the meaning of the word “blood”. Was there a lot blood? Not a lot of blood? Hence, this part of the data isn’t used by current methods. Most importantly, supervised methods cannot learn from unlabeled data (which is why current approaches rely on expert probabilities to make deductions).

Recent work in natural language processing has proposed methods of learning effective representations of *unlabeled* data. Models such as BERT, RoBERTa, and VAMPIRE (Devlin et al., 2019; Liu et al., 2019; Gururangan et al., 2019) learn continuous representations of unlabeled text using pretraining methods. Gururangan et al. (2020) have also shown that carefully designed pretraining methods are useful in low-resource classification tasks, for which limited labeled data exists. In this work, we explore semi-supervised methods for Verbal Autopsy, which we frame as a classification task, following prior literature (Clark et al., 2018). Semi-supervised methods can be a useful way of incorporating all aspects of the VA dataset, including the open narrative text, as well as learning from unlabeled data, as we address in Chapter 8.

Pretraining is not only a useful way of incorporating unlabeled task-related data. It can also be used to learn from data that is completely unrelated to the task. However, not all data is useful to pretrain on. Recent work in NLP has explored constructing different pretraining regimens based on the text domain of classification tasks (Gururangan et al., 2020). This work suggests that domains might be hierarchical, and successive rounds of pretraining on data that is increasingly task related can result in useful representations in a downstream classifier. In general, “domain” is a concept that is important in NLP. While often difficult to categorize, especially with a new task, understanding text domain and how it may be structured allows us to select the most effective training data. In this work, we’re interested in understanding how we can define domain in Verbal Autopsy. One plausible definition might be that VA data from different sites around the world are considered different domains. If we do treat sites as domains, is it possible to achieve domain transfer by training on one site and evaluating on another? What about data from outside the Verbal Autopsy task? Are general medical scientific papers useful to pretrain on? We attempt to answer these questions in our experiments in Chapters 8 and 10. Our goal is to understand which data is useful for the task, both in our downstream classifier and also in the pretraining phase.

1.1 Contributions

This thesis contributes the following:

- We examine current methods and most common data/resource settings for the Verbal Autopsy task (Chapter 3).
- We apply an existing light-weight, semi-supervised model, VAMPIRE, to the Verbal Autopsy task and demonstrate that semi-supervised methods are more successful than statistical baselines for this task (Chapter 8).
- We propose a variant of VAMPIRE, which incorporates input from earlier statistical models, to use expert information in a semi-supervised manner. We demonstrate that in the lowest data resource scenarios, external information is useful (Chapter 9).
- We explore applying larger pretrained models with varying degrees of domain and task adap-

tation to the Verbal Autopsy task and show that sequence based models suffer on a task that is framed for a bag of words method (Chapter 10).

- We propose recommendations on how to utilize data in three different data resource settings for Verbal Autopsy (Chapter 11).

Chapter 2

DATA

2.1 Verbal Autopsy Dataset

The Verbal Autopsy dataset is public and available from the Population Health Metrics Research Consortium (PHMRC) website. It consists of survey data from six geographical sites ranging from 2005 to 2011. Table 2.1 describes the breakdown of data available per site. Each data point is labeled with a cause of death from a list of 34 possible CODs. This task is low-resource. There are a total of about 8000 labeled data points available. Additional unlabeled data does exist, but is unreleased.

2.2 Verbal Autopsy Survey Content

The survey content can be split into three categories: demographic data, responses to survey questions, and open narrative text. The demographic section of the survey includes information such as date of birth, date of death, gender, language spoken, marital status, age, etc. The demographics covers information about both the decedent and the survey respondent. The bulk of the data consists of responses to over 200 survey questions. These questions aim to understand the decedent's symptoms prior to their death. Sample questions include 'how long was the decedent ill before they died?', 'did the decedent have blue lips?', and 'did the decedent have pain upon swallowing?'. At

Site	Labeled Deaths
Andhra Pradesh	1554
Bohol	1259
Dar es Salaam	1726
Mexico City	1586
Pemba	297
Uttar Pradesh	1419

Table 2.1: Number of labeled deaths available per site in the Verbal Autopsy dataset.

times, the questions are progressively more specific versions of previous questions. For example: 'did the decedent have headaches?', 'for how long before death did the decedent have headaches?', 'was the onset of the headache fast or slow?'. Typically, responses are either Yes/No/Don't Know, a numerical unit, or similarly categorical answers (fast/slow, continuous/on and off). Finally, the open narrative text section aims to capture whether certain words appeared anywhere in the VA interview. This portion consists of over 600 words and the 'responses' to these questions are 1 if the word appeared in the open narratives and 0 otherwise. Some examples of these words include 'brain', 'vein', 'malaria', 'failure', etc.

2.3 Re-Grouping Labels

In the public Verbal Autopsy dataset, there are 34 possible causes of death. Each death in Table 2.1 is assigned one of these gold-standard diagnoses, determined by medical records and other medical findings. We frame the Verbal Autopsy task as a text classification task. In other words, given the text of the VA survey, a model must assign a cause of death.

Not all labels are represented in all the sites and some labels are more common than others. For example, in Andhra Pradesh, "Bite of Venomous Snake" is the assigned COD for 31 deaths, whereas in Bohol, only one example has this COD. Similarly, there's only one labeled death for "Prostate Cancer" in Andhra Pradesh, and none in Pemba Island, but over 30 deaths for the label in Dar. The differences in label distribution suggests that 1) judging purely from the data, these sites are inherently different and 2) label sparsity might be a low-resource challenge to overcome for models. In general, data sparsity may arise from not only different causes of death in different geographic locations, but also variance in how the VA survey is administered. Nevertheless, the low-resource nature of the task results in data and label sparsity, which contributes to different label distributions across the sites.

In our preliminary work, we found that labels not represented in our evaluation data skewed our classification metrics; if there is no training data for a particular label, the model has little signal to accurately predict the correct label during evaluation time. Similarly, if there are only a few evaluation points for a given label, then incorrectly predicting one of those points might drastically drop precision/recall. To help fix this, we consolidate these 34 labels into 12 groups

Group Name	Labels Included
Cancer	Stomach Cancer, Leukemia/Lymphomas, Prostate Cancer, Esophageal Cancer, Colorectal Cancer, Breast Cancer, Cervical Cancer, Lung Cancer
Other NCD	Asthma, Other Non-Communicable Diseases, Epilepsy
Diabetes	Diabetes
Renal	Renal Failure
Stroke	Stroke
Liver	Cirrhosis
Cardio	Other Cardiovascular Diseases, Acute Myocardial Infarction, COPD
Other Comm	Malaria, Diarrhea/Dysentery, Other Infectious Diseases
Pneumonia	Pneumonia
TB/AIDS	TB, AIDS
Maternal	Maternal
External	Homicide, Suicide, Bite of Venomous Animal, Road Traffic, Other Injuries, Drowning, Falls, Poisonings, Fires

Table 2.2: We consolidate the original 34 COD labels into 12 groups based on similarities and expert input.

based on label similarity. For example, Lung Cancer and Leukemia/Lymphomas are grouped into one category, Cancer. Table 2.2 describes the groups chosen based on the expert recommendation of our collaborator, Tyler McCormick. We frame our classification task as predicting one of the 12 groupings of COD.

Chapter 3

EXISTING METHODS IN VERBAL AUTOPSY

There are several popular Verbal Autopsy algorithms that are currently used in the field. These models are supervised algorithms which rely on expert information, called SCI (Symptom-Cause-Information), to predict the likely cause of death. SCI can be a variety of information, but it is most commonly conditional probabilities of symptoms and causes, $P(\text{symptom}|\text{cause})$. This probability can then be used, for example, in a naive Bayes classifier, to estimate the cause of death given the symptoms, $P(\text{cause}|\text{symptoms})$.

As Clark et al. (2018) point out, the choice of SCI is at least as important as the choice of VA algorithm. Current methods not only use SCI as a starting place, they also learn the SCI from the training data. The method by which physician prescribed conditional probabilities are combined with learned probabilities results in some variation in SCI.

As pointed out in Chapter 1, current methods use at most expert information and labeled training data (some use only one or the other). They do not use the open text narratives or unlabeled data. In this work, we also consider a supervised neural classifier, discussed in section 3.5, which does use the open text narratives. This additional baseline will allow us to set up a more accurate comparison with semi-supervised methods.

Clark et al. (2018) also perform a comprehensive study of how training data from one site affects evaluation performance when the target site might be different. They find that training and testing on data from the same site is extremely important. This is an intuitive conclusion, because data from the same site is likely to follow similar patterns. However, this is problematic when limited training data exists for a target site. As a result, models must heavily rely on predetermined SCI.

Clark et al. (2018) test four standard VA models, implemented in the OpenVA R package (Li and Clark, 2018 (accessed Sept, 2019,(,)). We utilize these same four models as our baselines.

In addition, we also choose to evaluate a simple neural classifier (CLF), or multi-layer perceptron (MLP), as a fifth baseline. We describe this baseline in more detail in Section 3.5. None of the

current VA baselines are neural, and we hypothesize that by introducing simple non-linearities, we might achieve some gains on this data. We use an existing implementation of a classifier, taken from VAMPIRE, implemented in AllenNLP (Gardner et al., 2018).

3.1 Tariff

Tariff is an additive algorithm which calculates a score, or “tariff”, for each cause of death for each symptom across a VA training dataset. The scores are summed for a given data point and the result informs the predicted cause of death. The tariff is calculated formally for a given cause i and a symptom j as follows and originally described by James et al. (2011):

$$\text{Tariff}_{ij} = \frac{x_{ij} - \text{Median}(x_{ij})}{\text{Interquartile Range}_{ij}}$$

In the above, x_{ij} is the fraction of VA deaths in the dataset for which there is a positive response to deaths from cause i for symptom j . For example, if symptom j was, “abdominal bleeding”, a positive response would be “yes”. The scores are an indication of how likely the symptom is related to a particular cause of death. $\text{Median}(x_{ij})$ is the median fraction of VA deaths with a positive response for symptom j for all causes i . The denominator refers to the interquartile range of positive response rates averaged across causes.

The final tariff score for a given death k and a given cause i is then calculated as

$$\text{Tariff Score}_{ki} = \sum_{j=1}^w \text{Tariff}_{ij} x_{jk}$$

In the above, w refers to the number of symptoms for a given VA death and x_{jk} is the response for a given death k on symptom j (0 if negative, 1 if positive). Note that a separate tariff score is calculated for each COD for each VA death. The final COD assigned is the normalized, highest scoring COD.

3.2 InterVA

InterVA is a Bayesian probabilistic method for interpreting Verbal Autopsy data. If there are a set of possible causes of death ($C_1 \dots C_m$) and a set of possible indicators or symptoms/circumstances

$(I_1 \dots I_n)$, the general Bayes' theorem probability for a C_i and I_j is

$$P(C_i|I_j) = \frac{P(I_j|C_i)P(C_i)}{P(I_j|C_i)P(C_i) + P(I_j|\neg C_i)P(\neg C_i)}$$

where $P(\neg C_i)$ is $1 - P(C_i)$. We can apply this rule to calculate the probability of a cause of death given an indicator as follows.

$$P(C_i|I_j) = \frac{P(I_j|C_i)P(C_i)}{\sum_{k=1}^m P(C_k)}$$

This calculation can be easily extended to multiple indicators. Calculating this quantity relies on an initial set of unconditional probabilities for each cause of death and a matrix of conditional probabilities $P(I_j|C_i)$ for each indicator j and cause of death i . InterVA uses expert input to populate the conditional probabilities matrix.

3.3 *InSilicoVA*

InSilicoVA is another probabilistic method which builds on InterVA. It uses the same physician provided conditional probabilities as InterVA. It aims to improve the statistical model used by InterVA in numerous ways, including quantifying uncertainty. The key difference between InterVA and InSilicoVA is that the latter shares information between inferred individual causes of death and population COD distributions while the former does not (Clark et al., 2015).

3.4 *NBC*

The NBC method is a naive Bayes classifier. It uses the same physician provided conditional probabilities to calculate the probability of each cause of death given a set of observed symptoms using Bayes rule. Note that in this method, and all the methods that use physician provided conditional probabilities, the probability matrix between causes and symptoms is subjective. It's not clear, even amongst physicians, how to estimate the joint probability of causes and symptoms, especially because it's often the case that a person dies from multiple causes and symptoms.

3.5 Simple Neural Classifier (CLF)

This classifier is a multi-layer perceptron (MLP) which uses a bag-of-words encoder and a linear layer for classification, originally used and implemented by Gururangan et al. (2019). The output of the linear layer is a set of classification logits for each of the causes of death. In general, a multi-layer perceptron consists of an input layer, output layer, and one or more hidden layers of nonlinearly-activating nodes.

In Chapter 6.3 we describe how we transform the original Verbal Autopsy data to a text only format to be fed into our semi-supervised methods. This transformation process also applies to the CLF. The neural CLF does not use any expert probabilities, but instead learns from the training data.

We mostly use the same default parameters from Gururangan et al. (2019). These include a learning rate of 0.001, dropout with $p = 0.3$, batch size 32, 1 encoder layer, and 2 output layers. We fix random seeds throughout our initial experiments to be 0 for the CLF.

3.6 Current Approaches to Data Settings

Clark et al. (2018) and the general VA literature point out that obtaining labeled training data is very difficult. This is because medical autopsies are expensive. Annotating causes of death requires significant physician expertise and is infeasible for large amounts of VA surveys. The result is large amounts of unlabeled data. Clark et al. (2018) also use the practice of training and testing with data from the same site. This is intuitive, because as Clark et al. (2018) note, data from different PHMRC sites are heterogeneous and usually, scientists are interested in evaluating sites one at a time. In this work, we also choose to evaluate our methods on one site, but experiment with training on multiple sites. Clark et al. (2018) use an experimental setup where they train models on one site and evaluate on another site. The PHMRC dataset consists of 6 sites, so for a given model, they evaluate 6 x 6 different experimental setups. In our experimental setups, as discussed in Chapter 6, we explore whether including data from other sites during training time is important and if so, how best to include it.

Chapter 4

USING UNLABELED DATA

Large models pre-trained on vast amounts of text, such as BERT, RoBERTa (Devlin et al., 2019; Liu et al., 2019), and other variants have normalized the pretraining paradigm as an essential part of today’s NLP methods. Pretraining is the process of training a language model with large corpora of unlabeled data. The goal is to learn a general language representation which can be used in downstream tasks. In our case, this task is text classification. The representations learned during pretraining have been shown to improve performance on a wide variety of tasks, both high and low resource (Gururangan et al., 2020). However, despite their widespread success, transformer based models are computationally expensive. Recent work by Gururangan et al. (2019) has proposed computationally efficient alternatives to transformers for pretraining. The VAMPIRE model uses a variational auto-encoder to learn interpretable text representations. This method has also been shown to be especially useful for low-resource tasks.

In this work, we consider both high and low computational resource settings. Since the current approaches explored in Chapter 3 are computationally cheap, we examine VAMPIRE as the most viable semi-supervised method for Verbal Autopsy. In the case where we have access to larger computational resources, we ask, are larger models better suited for this task? We explore RoBERTa as an alternative to VAMPIRE. We choose RoBERTa over other large pretrained transformer models because it has been shown to outperform BERT in many settings due to its larger size and pretraining corpus (Liu et al., 2019).

4.1 VAMPIRE

We explore using the VAMPIRE model on the Verbal Autopsy data. VAMPIRE is a text classification framework which takes advantage of the pretraining paradigm. It uses a variational auto encoder to pretrain embeddings on unlabeled text data. These embeddings are later used as features in a downstream classifier. VAMPIRE is unique in the way that it performs the pretraining step.

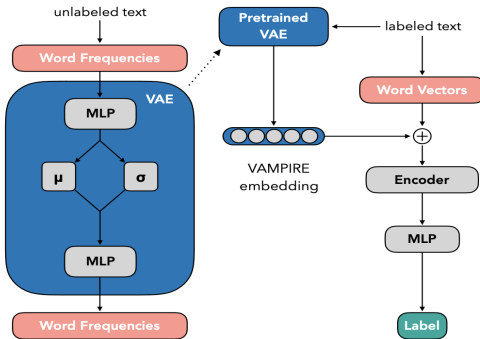


Figure 4.1: We explore applying the VAMPIRE model to the Verbal Autopsy data. Figure taken from Gururangan et al. (2019)

While models such as BERT and RoBERTa interpret text as a sequence of words, VAMPIRE interprets inputs as a bag of words. As a result, VAMPIRE is a lightweight model and can be trained efficiently on a CPU.

We choose to explore using VAMPIRE for Verbal Autopsy for several reasons. VAMPIRE was designed specifically with the low computational resource environment in mind. VAMPIRE also performs effectively on tasks with limited labeled data, but access to some unlabeled data (Gururangan et al., 2019). Additionally, with survey data, sequence may not matter; the order of questions does not necessarily affect their responses. Using a sequence-agnostic model like VAMPIRE could be better suited for this task.

4.1.1 Pre-training with VAMPIRE

The VAMPIRE model assumes that all documents in the pretraining corpus are generated from a latent variable z . The goal is to estimate z using an encoder and a decoder. The encoder maps the input text x to an approximate posterior $q(z|x)$, which is assumed to be normally distributed, and the decoder reconstructs the text. The normal posterior is parameterized by σ and μ . Rather than using sequence to sequence encoding and decoding, VAMPIRE uses word frequencies.

To illustrate the encoding procedure, VAMPIRE starts with a word frequencies vector c_i for each data point i . This vector is passed through an MLP to produce the hidden state h_i . The hidden state, in combination with separate feedforward layers, is used to estimate the parameters of the normal

posterior, μ and σ . The approximated posterior, z_i , is a linear combination of μ and σ .

The decoder takes z_i and places a softmax over it to produce θ_i , a distribution over latent topics, and the reconstruction of the original counts vector c_i using another feedforward layer.

4.1.2 Incorporating Pre-trained VAE in Downstream Text Classification

As mentioned by Gururangan et al. (2019), Kingma and Welling (2014) proposed using the internal states of a VAE as features in a downstream classifier. Gururangan et al. (2019) builds on this idea by using a combination of θ_i from above and a weighted sum over each hidden layer k , denoted $h_i^{(k)}$, in the encoder MLP. The resulting vector is concatenated with the text representation within the downstream classifier, which then produces classification logits.

4.2 RoBERTa and BioMed RoBERTa

We consider two variants of the large, transformer-based pretrained models: RoBERTa and BioMed RoBERTa (Liu et al., 2019; Gururangan et al., 2020). These two models are at opposite ends of the pretraining spectrum. The former is generally pretrained on a large corpus of English text while the latter is additionally pretrained on biomedical papers. In other words, we explore using a model with no domain adaptation and a model with both domain and task adaptation for Verbal Autopsy to understand whether larger models perform well on this task.

4.2.1 RoBERTa

RoBERTa uses the transformer architecture from Vaswani et al. (2017). During pretraining, BERT originally uses the Masked Language Model (MLM) and Next Sentence Prediction (NSP) objectives. However, with RoBERTa, Liu et al. (2019) found that without the NSP objective, the embeddings performed better in downstream tasks. RoBERTa is pretrained for significantly more steps than BERT and on a larger pretraining corpus consisting of BOOKCORPUS (Zhu et al., 2015) plus ENGLISH WIKIPEDIA as well as CC-NEWS, OPENWEBTEXT, and STORIES (Trinh and Le, 2018).

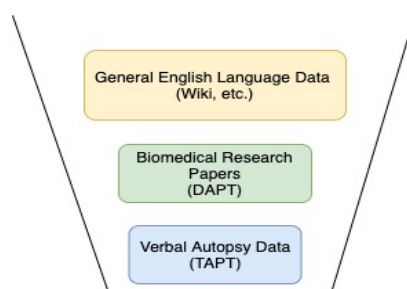


Figure 4.2: We explore using DAPT+TAPT with RoBERTa on Verbal Autopsy.

4.2.2 DAPT + TAPT and BioMed RoBERTa

Gururangan et al. (2020) suggest that generically pretrained models, such as RoBERTa, can be adapted to a particular task through multiple rounds of more task specific pretraining. Gururangan et al. (2020) propose two methods of adapting models through pretraining: domain adaptive pretraining (DAPT) and task adaptive pretraining (TAPT), illustrated in Figure 4.2. With DAPT, we additionally pretrain RoBERTa on a corpus of text that is assumed to be in the same text domain as our task data. Theoretically, this should give the model a better idea of what the task data looks like. DAPT is often performed with a larger corpus, especially in the case where large amounts of unlabeled domain data exists but similar task specific data may not exist. On the other hand, if there is some unlabeled task specific data, we can continue pretraining RoBERTa with TAPT. Gururangan et al. (2020) show that the combination of DAPT + TAPT, performed in that order, allow the large pretrained model to learn a progressively better representation of the unlabeled task data.

Gururangan et al. (2020) propose a variant of RoBERTa, BioMed RoBERTa, which is additionally pretrained on 2.68 million biomedical papers from the Semantic Scholar Open Research Corpus (Lo et al., 2019). If we consider biomedical research papers as a parent domain of Verbal Autopsy data, we can use BioMed RoBERTa as a step in the domain-adaptive pretraining process. To understand whether this will be useful, we compare the performance of RoBERTa (with no additional pretraining) with BioMed RoBERTa (domain adaptive pretraining) with task-adaptive pretraining on the Verbal Autopsy data.

Chapter 5

DOMAIN

“Domain” is an important concept in NLP. Intuitively, text data from different sources and contexts should be interpreted in different ways. For example, technical research papers use different jargon than a Wikipedia article, both of which are different from casual messages between friends. These different types of texts can be categorized as different text domains. Understanding the domain of our evaluation data will give us a good idea of how to train the model and which data is most useful to train and pretrain on.

In Verbal Autopsy, we know that data comes from six geographic sites. Perhaps these locations are different domains? Prior work indicates that training data from sites outside the evaluation site is less helpful for a supervised classifier (Clark et al., 2018). No work has explored using this data in an unlabeled manner for pretraining. Presumably, there are similarities between all Verbal Autopsy data points. Responses might be comparable between similar geographic locations (Andhra Pradesh and Uttar Pradesh are both in India, Pemba Island and Dar es Salaam are separated by a small body of water). However, as we showed in Chapter 2, label distributions across sites can be vastly different and therefore, data from one site may not be applicable to another. In this work, we are interested in discovering the circumstances required for domain transfer between sites.

Gururangan et al. (2020) suggests that domain might be hierarchical. In other words, data from Dar es Salaam could be a smaller domain in the overarching Verbal Autopsy domain. We can also think about which domains encompass VA data. One such domain might be general medical data, or even medical publications. Gururangan et al. (2020) demonstrated that a combination of domain adaptive pre-training (DAPT) and task adaptive pretraining (TAPT) achieve better results on several classification tasks. In our experiments, we carefully vary pre-training and training data for our models to assess this method. While it may not necessarily be the case that biomedical data is a parent domain of Verbal Autopsy data, we explore using domain adaptive pretraining with RoBERTa for the Verbal Autopsy task in Chapter 10.

Chapter 6

EXPERIMENTAL SETUP

6.1 Motivation

As mentioned in Chapter §3, we are interested in closely examining a particular relevant scenario within Verbal Autopsy. In this scenario, researchers conduct VA surveys in a new site where there exists some labeled data but potentially larger amounts of unlabeled data. This scenario is interesting for several reasons. Unreleased unlabeled data exists for many of these sites, and finding ways to effectively use this data serves as motivation to release more of the unlabeled data. The unreleased data exists in World Health Organization archives, for example, but is not released publicly or for researchers to use. In a practical scenario, most sites have some labeled and unlabeled data, so developing a model that can handle both types of data is more scalable; we can always add more data as it becomes available. In addition, simulating the scenario of both labeled and unlabeled data allows us to understand how we can achieve domain transfer within the Verbal Autopsy data as well as parent domains of VA data. We frame our VA experiments in terms of the target site, which we term the “target domain” (TD), and the remaining five sites, which we refer to as the “non-target domain” (NTD). While it is entirely possible that data from the target site closely matches data from external sites, we anticipate that the label distribution and certain attributes about the target site will be unique. As discussed in Chapter §3, previous work suggests that target domain data is the most useful, because it most closely resembles the evaluation data. We additionally explore how useful labeled data (TD or NTD) is, compared to unlabeled data. We hypothesize that labeled data is more useful, because it gives a direct signal between the VA survey text and the cause of death, whereas with unlabeled data, we must learn an effective representation first.

6.2 Splitting the Data

In order to simulate the data scenario described in Section 6.1, we consider all data from Dar es Salaam to be our target domain data. Data from the remaining five sites is considered to be non-

target domain. The choice of Dar as the target site is mostly arbitrary, but mainly driven by the fact that it has more data points available for us to test hypotheses while minimizing noise associated with a smaller target site.

6.3 Interpreting VA Data for VAMPIRE (and other neural models)

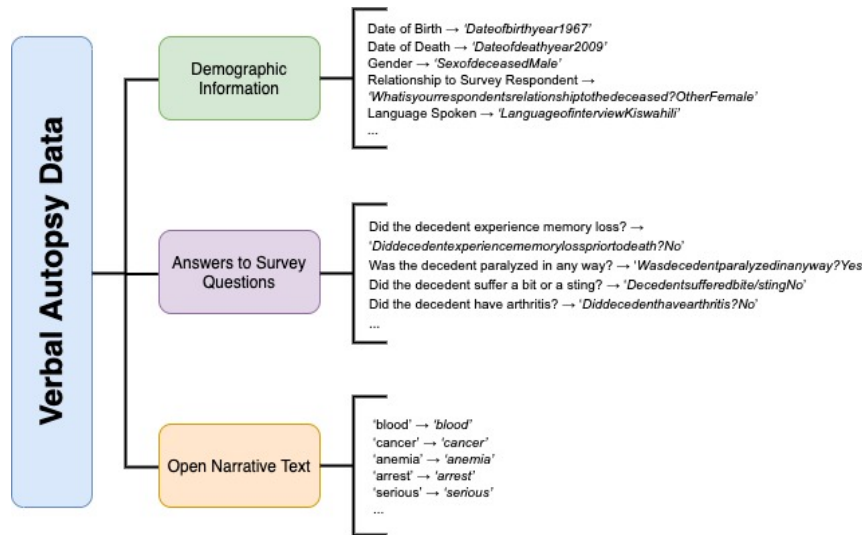


Figure 6.1: We concatenate questions and answers for each part of the Verbal Autopsy dataset to produce the data for VAMPIRE and all neural models

As mentioned in Chapter 2, the Verbal Autopsy data consists of three major parts: demographic data, survey responses, and open narratives text. VAMPIRE, and all the neural models we consider, takes as input a string of text. The sequence of text we provide to VAMPIRE is constructed from the VA data as illustrated in Figure 6.1. For demographic data, we concatenate the data category with the data response. For example, if the decedent’s gender is female, the token we produce is 'Sex-ofdeceasedFemale'. Or, if the last known age of the deceased was 37, the token produced is 'Last-knownageofthedeceasedyears37'. For survey data, we employ a similar procedure, concatenating the question and the answer. For example, if the question was 'did the decedent have hypertension?' and the answer was 'no', the token produced would be 'DidDecedentHaveHypertension?No'. Finally, for the open narratives data, we include the word only if the response is 1. In other words, if the word appeared in the open narrative, it also appears in our text representation. In particular,

this representation of the Verbal Autopsy data does not take into account prior probabilities or any form of symptom-cause information. We experimented with several methods of interpreting the VA survey data with VAMPIRE but found that the above mentioned simple concatenation technique and treating the result as a single token was the best performing.

6.4 Evaluation Metrics

Clark et al. (2018) report several metrics for their experimental results, among which are Top 1 Accuracy and Top 3 Accuracy. However, these metrics are less informative when the label distribution in the evaluation data is skewed. We are interested in false positives and false negatives and precision, recall and F_1 allow us to quantify these metrics. In all our experiments we report precision, recall, and F_1 scores per label as well as a macro F_1 scores averaging over all labels.

6.5 Constructing Data Resource Settings

Though we are most interested in the data scenario with large amounts of unlabeled data, it's also important to consider cases when we have access to some labeled in-domain data. Clark et al. (2018) show that labeled target domain data is the most useful and we use this finding to build our three data resource settings. We frame our experiments in the context of three settings: No Labeled Target Domain Data, Few Labeled Target Domain Data, and More Labeled Target Domain Data. As the names suggest, these settings simulate different combinations of labeled and unlabeled data and will inform which data we use to pretrain and train on. Varying the amount of labeled target domain data also allows us to understand the relative important of labeled and unlabeled non-target domain data. In all three settings, we assume access to a small amount of labeled target domain data, which we reserve for evaluation.

Figure 6.2 highlights our data settings. These three resource settings drive our experiments and our eventual recommendations. We believe that these three settings encompass the scope of Verbal Autopsy scenarios a researcher might encounter when entering a new site. A practical VA model should perform well in all three scenarios. We hypothesize that different models may be useful in different environments, depending on the resources available.

Cross-Validation The Verbal Autopsy task is in itself a low-resource task. Prior work has demonstrated that with smaller datasets, there’s more variance in model performance (Dodge et al., 2020). To ensure that our methods generalize, we evaluate all our models using three-fold cross-validation using the target domain data, since it is our evaluation data.

It is important to note that we do not need to divide up the non-target domain data in a similar way. In each of our experiments, we only consider using the non-target domain data in its entirety or not at all. In other words, the non-target domain data remains constant throughout all our cross-validation splits.

For each cross-validation split, the non-target domain data is constant and consists of 6115 labeled data points from five sites. There are 1726 data points from Dar. This would make each partition in our cross-validation setup be 33% of the data, or approximately 575 points. Thus, for each cross-validation split, the training data is 67% of the Dar data and the evaluation data is the remaining 33%.

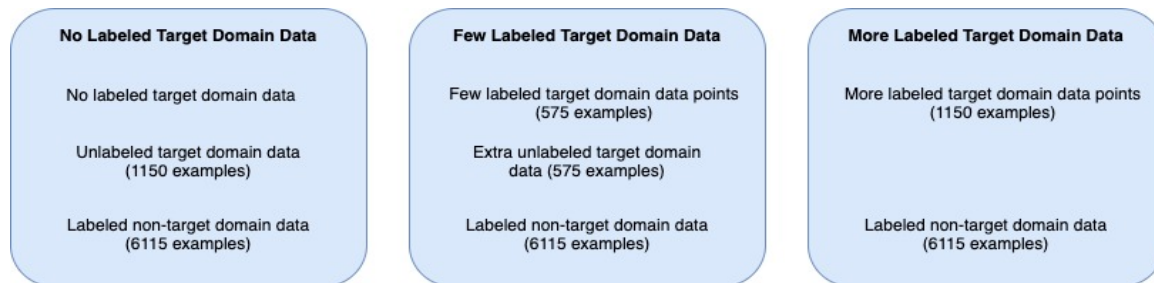


Figure 6.2: We explore three data settings in this work. These settings simulate the range of available labeled and unlabeled target domain and non-target domain data a researcher might encounter at a new site

6.5.1 No Target Domain Data

In this setting, we consider the scenario where we do not have access to any labeled target domain data for training. We do, however, have access to all the labeled non-target domain data as well as some unlabeled target domain data. We treat the training portion of the cross-validation split, or the approximately 1150 deaths of the Dar data, as our unlabeled target domain data.

6.5.2 *Few Target Domain Data*

In this setting, we consider adding a relatively small amount of labeled target domain data. We take the training portion of the cross-validation split (1150 of the Dar data points) and randomly split it in half. We treat one half of the training data as labeled data and the other half as unlabeled. In other words, for this setting, we have access to 33% labeled Dar data (approx. 575 deaths), and 33% unlabeled Dar data (approx. 575 deaths). We still have access to all the labeled non-target domain data, but seeing as we have labeled target domain data, we can choose whether we use this non-target data with or without the labeled target data. This scenario simulates the case where researchers have access to a small handful of labeled target domain data (about 500 examples).

6.5.3 *More Target Domain Data*

In this highest data resource setting, we aim to simulate the case where researchers have access to over 1000 labeled target domain data points. To do this, we treat the entire training portion of the cross-validation split as labeled. In other words, we have access to 67% (approx. 1150 deaths) of labeled Dar data, as well as all the labeled non-target domain data, and in our experiments we explore different combinations of this labeled data as we described in the previous section.

Chapter 7

EVALUATING SUPERVISED METHODS

We begin by exploring how our supervised models, explained in Chapter 3, perform on the Verbal Autopsy task. None of these experiments utilize the unlabeled data in each experimental setup. Performing these baseline experiments is necessary because our work evaluates models differently than previous work since we consolidate labels into 12 groups and use precision/recall and F_1 as our evaluation metrics. Understanding how models perform in the supervised settings gives us baselines to compare against when we consider semi-supervised methods.

7.1 Training Details

All baselines are trained using the implementation provided in the OpenVA R package. For the baselines, we fix the random seed to be 5. For the classifier, we fix the random seed to be 0. The simple neural classifier is trained for 60 epochs. We use the existing hyperparameters from the CLF used by Gururangan et al. (2019) without additional tuning.

7.2 Results

Table 7.1 describes the experimental results for each baseline model in each data setting. As we described in Chapter 6, each data setting is driven by the amount labeled target domain (TD) data available. For the latter two data settings, we have labeled target domain and non-target domain (NTD) data, so we consider either training on just the TD data or both TD and NTD data.

7.2.1 No Target Domain Data

In this setting, the only option we have is to train the baseline models on the labeled non-target domain data. Tariff is the highest performing model in this scenario, by about 2 F1 points over the next best model, the CLF. Performance in this category reflects how well a model can extrapolate

Model	No TD Data	Few TD Data		More TD Data	
	NTD	TD	TD + NTD	TD	TD + NTD
Tariff	0.353 _(0.03)	0.368 _(0.00)	0.373 _(0.02)	0.387 _(0.00)	0.381 _(0.02)
InterVA	0.309 _(0.01)	0.327 _(0.00)	0.327 _(0.01)	0.350 _(0.00)	0.341 _(0.02)
NBC	0.329 _(0.01)	0.384 _(0.00)	0.354 _(0.01)	0.401 _(0.00)	0.359 _(0.01)
InSilicoVA	0.287 _(0.03)	0.398 _(0.00)	0.311 _(0.01)	0.405 _(0.00)	0.312 _(0.01)
CLF	0.333 _(0.01)	0.344 _(0.03)	0.454 _(0.01)	0.397 _(0.00)	0.468 _(0.03)

Table 7.1: F_1 results for all baseline models in each data resource setting. The number in the subscript indicates the standard deviation across the three cross-validation splits. Bolded numbers indicate the best performing model in each data setting.

information from non-target domain data. In this case, it appears that the SCIs used by the baselines are helpful when there is limited information.

7.2.2 Few Target Domain Data

For both remaining data settings, we have access to two pools of labeled data: target domain and non-target domain. For these two experiments, we consider two options: either we train on only the target domain data or both of the target domain and non-target domain datasets.

In the first case, when we only train on the small amount of target domain data, we observe that every model performs better than if they were only trained on the non-target domain data, as in Section 7.2.1. This is, perhaps, an intuitive finding. Labeled target domain data more closely resembles the evaluation data and the statistical baselines can learn a distribution that is more helpful during evaluation time. More interestingly, our findings show that a small amount of labeled target domain data (500 examples) is at least as useful as a larger amount of non-target domain data (6000 examples).

In our second case, we consider training on all the available labeled target domain data as well as the non-target domain data. Surprisingly, some of the baseline models perform worse with this additional non-target domain data. Perhaps they struggle to handle the differences between the two types of data. On the other hand, the CLF greatly improves with the additional data, demonstrating an 11 point F_1 gain over the case where we only use labeled TD data. This suggests that when we add a small amount of labeled target domain data to our larger amount of labeled non-target domain data, the CLF learns best and receives more signal about the evaluation data.

7.2.3 *More Target Domain Data*

Once again, we consider training on only the labeled target domain data or both the target domain and non-target domain data. Here, we have access to a far larger amount of labeled TD data (over 1000 examples).

As expected, training only on a larger amount of target domain labeled data does improve model performance over the parallel Few Target Domain Data setting. On average, the baseline models trained on about 1000 TD data points improve by about 2 F_1 points over the same models trained on approximately 500 TD data points (the difference between the two TD columns in Table 7.1). The neural CLF improves by about 6 F_1 points by the same comparison.

However, when we add in the out-of-domain data, we see mixed results for the statistical baselines. In two cases, the difference between these models and the models trained on only the target domain data is minimal. And for the remaining two, performance drops drastically. However, for the CLF, we still notice an increase in performance over the previous setting’s TD + NTD experiment, albeit a smaller one. This suggests that the neural classifier responds well to any additional data, TD or NTD, and is better at reconciling any inherent differences between the TD and NTD data. The statistical baselines may not have the same ability to differentiate between TD and NTD data. That, or the statistical baselines may have plateaued and adding additional data is less helpful.

Amongst our latter two data settings, it appears that unsurprisingly, labeled target domain data is consistently more valuable than labeled non-target domain data. Adding more of the former steadily increases model performance while for the latter, we have mixed results. In general, we notice that over the three data settings, as we add more labeled target domain data, the statistical models improve much less than the neural CLF (which drastically improves between the first two data settings). This suggests that neural methods have potential to do better.

Chapter 8

VAMPIRE PERFORMANCE

The VAMPIRE model allows us to use portions of the training data which we were previously unable to take advantage of in the No TD data and Few TD Data settings. As we explored in Chapter 7, we also must make a choice of what to pretrain on, and which labeled data to use in our downstream classifier. In this section, we explore different choices of pre-training and training data combinations and how they affect performance on the same datasets.

8.1 Training Details

All versions of VAMPIRE are trained using the implementation from Gururangan et al. (2019). We fix the random seed to be 0 and re-use most of the hyperparameters from Gururangan et al. (2019). We set the patience to be 15 and train our downstream classifier using this value. The patience was set using manual search.

8.2 No Target Domain Data

In the No Target Domain Data setting, the only choice we must make with VAMPIRE is which data to pretrain on. It’s a given that we will train our downstream classifier on the labeled non-target domain data. Either we pretrain on just the non-target domain data (which we already have labels

Model	No TD Data
Tariff	0.353 _(0.03)
CLF (NTD)	0.333 _(0.01)
+ VAMPIRE (TD)	0.335 _(0.02)
+ VAMPIRE (NTD)	0.348 _(0.01)
+ VAMPIRE (TD + NTD)	0.348 _(0.02)

Table 8.1: F_1 results for VAMPIRE in the No Target Domain Data setting. We also include our best performing baseline, Tariff, as well as the CLF performance without pretraining, both of which are taken directly from Table 7.1.

Model	Training on TD Data	Training on TD + NTD Data
Tariff	0.368 _(0.00)	0.373 _(0.02)
InSilicoVA	0.398 _(0.00)	0.311 _(0.01)
CLF	0.344 _(0.03)	0.454 _(0.01)
+ VAMPIRE (TD)	0.390 _(0.01)	-
+ VAMPIRE (TD + NTD)	0.385 _(0.04)	0.473 _(0.03)

Table 8.2: F_1 results for VAMPIRE in the Few Target Domain Data setting. We also include our two best performing baselines, Tariff and InSilicoVA, as well as the CLF performance without pretraining, both of which are taken directly from Table 7.1.

for), the unlabeled target domain data, or both.

Table 8.1 describes the experimental results for VAMPIRE in this data setting. The three variants of VAMPIRE perform very similarly, although the last scenario, where we pretrain on as much unlabeled data as possible, is marginally better than the rest, if we just examine macro F1 scores. All three cases of VAMPIRE outperform most of the statistical baselines and the neural CLF in this data setting. This is an indication that pretraining, whether on target domain or non-target domain data, is a beneficial strategy. However, the Tariff method still is the best performing in this setting. Further experiments remain to show exactly which pretraining data is most important in each resource setting.

8.3 Few Target Domain Data

In the Few TD Data setting, we consider what happens when we steadily increase the amount of data we have available to us. We start with just the labeled TD data. Our first version of VAMPIRE pretrains on the TD data and has a downstream classifier trained on the fewer TD labels. This model outperforms just the CLF trained on the TD data, as can be seen from the first column of Table 8.2, so it appears that the VAMPIRE embeddings are useful. Now, if we also pretrain on the unlabeled NTD data, but still train the downstream classifier on only the labeled TD data, we notice a slight decline in performance (last number in the first column of Table 8.2). VAMPIRE still does better than the CLF on its own, but adding in unlabeled non-target data during pretraining seems to confuse the model enough to drop performance. This could be because the non-target domain data sufficiently skews the learned embeddings away from a representation useful for target domain

Model	Training on TD Data	Training on TD + NTD Data
Tariff	0.387 _(0.00)	0.381 _(0.02)
InSilicoVA	0.405 _(0.00)	0.312 _(0.01)
CLF	0.397 _(0.00)	0.468 _(0.03)
+ VAMPIRE (TD)	0.414 _(0.01)	-
+ VAMPIRE (TD + NTD)	0.413 _(0.02)	0.482 _(0.03)

Table 8.3: F_1 results for VAMPIRE in the More Target Domain Data setting. We also include our two best performing baselines, Tariff and InSilicoVA, as well as the CLF performance without pretraining, both of which are taken directly from Table 7.1.

data in the downstream classifier. We do not consider the scenario where we train the downstream classifier on all available data (TD + NTD) but only pretrain on TD data because we intuit that if data is available to us in the downstream classifier, we should pretrain on it. Finally, we use all available labeled and unlabeled data by pretraining and training the downstream classifier on the TD and NTD data. As expected, this performs the best. VAMPIRE, and in general, our neural models seem to respond favorably to more data, once again. That said, if we trained a neural CLF on both TD and NTD data, as we show in the second column of Table 8.2, the corresponding VAMPIRE model is only a couple of F_1 points better.

8.4 More Target Domain Data

Just like in the previous data setting, we consider three versions of VAMPIRE, all pretrained and trained on different data. As we see in Table 8.3, VAMPIRE, pretrained and trained only on target domain data is marginally better than the CLF trained only on TD data. Once again, when we also pretrain on NTD data, we notice a drop in performance, but this time it’s much smaller. Finally, when we pretrain and train on both TD and NTD data, we notice the largest improvement, just as in the previous setting. The performance of VAMPIRE in all these setting suggests that as we have access to more labeled and unlabeled data, VAMPIRE is the best model to use.

In all these three data settings, we notice that the best performing VAMPIRE model outperforms the best performing CLF model by only 1 or 2 F_1 points. Typically, when we add pretraining, we see much larger gains. However, our results represent a lower bound on the performance of VAMPIRE. We show that adding more data, labeled or unlabeled, increases the model’s performance. This

indicates that VAMPIRE has potential to do better with additional data.

Chapter 9

STACKING

In our earlier experiments, we find that in the setting with no labeled target domain data, the baselines perform better than the CLF or even VAMPIRE. VAMPIRE, in its original form, knows nothing about symptoms and causes and how they are related. However, the baselines have access to external symptom-cause information. We hypothesize that this expert information helps the models make better decisions when labeled target domain data is limited. In this section, we explore what would happen if VAMPIRE had access to the symptom-cause information, or could query the baseline predictions. In other words, we explore a variant of ensembling, called stacked learning, with the goal of improving VAMPIRE’s performance in the lowest data resource setting.

9.1 Related Work

Stacked learning is a technique originally proposed by [Wolpert \(1992\)](#) and [Breiman \(1996\)](#). The idea is to combine predictions from layers of classifiers in a hierarchical manner. More recently, this method has been applied to dependency parsing ([Martins et al., 2008](#)). In this case, stacking allows for better feature identification, which is important for the higher level dependency parser. Should I add more?

9.2 Stacking Method

Stacking involves training two levels of classifiers, level 0 and level 1. The level 1 classifier takes the classification outputs or probabilities from the level 0 classifiers, combines them with its own prediction, and outputs a new prediction. This method allows the level 1 classifier to gather extra information about a data point, by querying the baselines. In our scenario, we treat the four baselines as level 0 classifiers and our VAMPIRE + CLF model is the level 1 classifier. During the downstream classifier training, we incorporate the outputs of the level 0 classifiers into the logits of the CLF (which already uses its own prediction calculated from the VAMPIRE embeddings).

9.3 Stacked Training Details

9.3.1 Building the Augmented Dataset

We follow the stacked training procedure specified by [Martins et al. \(2008\)](#). For each stacked learning setting, we have a training dataset D which is split into M partitions. We train M instances of each of the four level 0 classifiers, c_i , using a hold-one-out method. In other words, the m -th instance of the i -th classifier c_i^m is trained on $D \setminus D^m$. We then use c_i^m to output predictions for the unseen partition D^m . This procedure allows us to build an augmented dataset, which is the union of the predictions of each instance of a given classifier on its respective unseen partition. We train the level 1 classifier on the augmented dataset. The intuition behind this augmented dataset is that it allows us to avoid using output predictions from the level 0 classifiers on data points they have already seen during training. For computing the level 0 classifier outputs to be used during the evaluation phase of the level 1 classifier, we train each level 0 classifier on the entire training set D and use the outputs predicted on the evaluation set. In our case, we choose M to be 10. Our dataset is small enough that choosing a smaller value for M would hinder the quality of the augmented dataset by reducing the number of training examples available to the level 0 classifiers.

9.3.2 Data Settings and Cross-Validation with Stacking

In each data setting, we consider the training dataset D from the previous section to be the set of all available labeled data, or what we would train our downstream classifier on. In both the stacking models we explore later in this chapter, we consider the training data to be all the labeled target domain and non-target domain data in each setting, even when we could choose not to use some of it, as we did for previous models. To be clear, in the No Target Domain Data setting, D is all the non-target domain data. In the Few Target Domain Data setting, D is all the non-target domain data plus 575 target domain data points and in the More Target Domain Data setting, D is all the non-target domain data and 1150 target domain data points.

If we construct our training dataset D as mentioned, we can still continue to perform three-fold cross-validation, as before. Previously, the cross-validation splits consisted of three training and evaluation partitions (as well as the non-target domain data). We use the the respective training partition for each cross-validation split to decide what constitutes labeled and unlabeled target-

domain data, just as before. We combine the labeled target-domain data with our non-target domain data to construct our stacked learning datasets. In other words, we train a separate stacked model for each cross-validation split.

9.3.3 Hyperparameters

We choose a dropout with $p = 0.2$. Our models are trained using the same hyperparameters as originally in Gururangan et al. (2019).

9.4 Stacked VAMPIRE

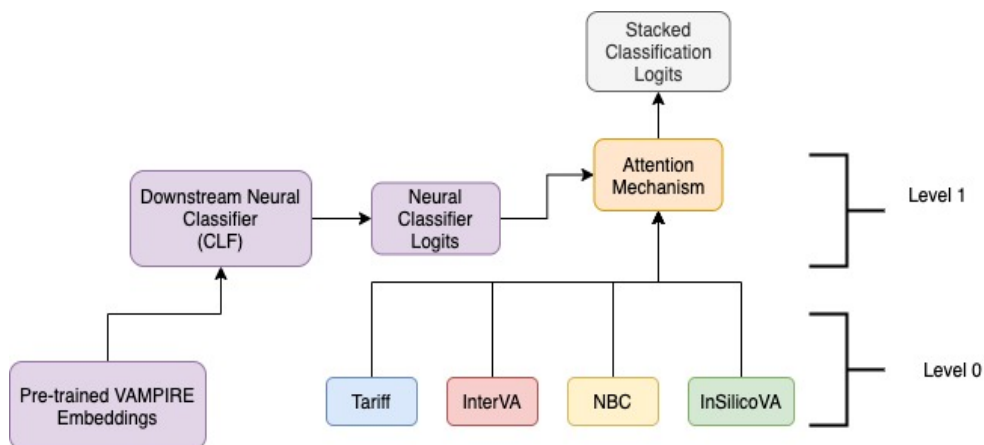


Figure 9.1: Our stacked learning setup with VAMPIRE, the downstream CLF, and inputs from the statistical baselines.

In the stacked VAMPIRE semi-supervised method, we incorporate baseline predictions only during the downstream classifier training. In other words, we don't look at any predicted labels during pretraining, so this phase is identical to the pretraining phase in earlier VAMPIRE experiments. Figure 9.1 describes the architecture of stacked VAMPIRE.

The downstream classifier (CLF) trains using the augmented dataset, as explained in Section 9.3.1. It reads the text input, minus the baseline predictions, and generates its own classification logits just as with earlier versions of VAMPIRE + CLF. Once we have the CLF logits, we combine them with the baseline predictions, interpreted as logits. The baseline predictions are single labels (i.e. 'Can-

cer’). We interpret this label as a 1×12 vector, where the index corresponding to the label is 1 and every other index has value 0. Thus, for four baseline predictions, we have a 4×12 matrix, where 12 is the number of COD labels. Our CLF logits are also a 1×12 vector, so we can concatenate it with our baseline results to produce a 5×12 matrix. To produce our final set of classification logits, we learn a 5×12 attention matrix. The goal of the attention matrix is to learn how to weight predictions made by each classifier for each label. Our final predicted logits are an attention weighted sum over the 5×12 logit matrix.

9.4.1 Stacked VAMPIRE Variants

We experiment with two versions of stacked VAMPIRE. The first uses inputs from all four baselines and combines them with the attention mechanism as described above. The second uses inputs from two baselines, chosen based on prior work and recommendations, and also uses the attention matrix but with dropout.

9.5 Stacked VAMPIRE Results

We compare our stacked VAMPIRE model to all previous models in the context of the same three data resource settings. As previously shown with VAMPIRE, pretraining and training on all the available labeled and unlabeled data produces the best results, so we use these settings for our stacked model. In other words, given the pool of data to use for each resource setting, we do not choose to leave out any of that data.

9.5.1 No Target Domain Data

In the first setting, our stacked VAMPIRE model pretrains on both the non-target domain data as well as the unlabeled target domain data. The downstream classifier is only trained on the labeled non-target domain data and uses inputs from the baselines. Stacked VAMPIRE improves the average F1 by about 2.5 points over the previous best performing model, Tariff. In this setting, where we have very little information about the target domain data, the symptom-cause information from the baselines, when combined with our neural methods, appears to play an important role in boosting performance. This information helps the model make better decisions on data points for which there

Model	No TD Data
Tariff	0.353 _(0.03)
CLF (NTD)	0.333 _(0.01)
+ VAMPIRE (TD + NTD)	0.348 _(0.02)
Stacked VAMPIRE (Attn)	0.368 _(0.02)
Stacked VAMPIRE (Attn, 2B, Dropout)	0.377 _(0.01)

Table 9.1: F_1 results for Stacked VAMPIRE in the No Target Domain Data setting. We also include our best performing baseline, Tariff, as well as the CLF and best configuration of VAMPIRE from Chapter 8.

Model	Few TD Data
InSilicoVA (TD)	0.398 _(0.00)
CLF (TD + NTD)	0.454 _(0.01)
+ VAMPIRE (TD + NTD)	0.473 _(0.03)
Stacked VAMPIRE (Attn)	0.4561 _(0.03)
Stacked VAMPIRE (Attn, 2B, Dropout)	0.4651 _(0.02)

Table 9.2: F_1 results for Stacked VAMPIRE in the Few Target Domain Data setting. We also include our best performing baseline, InSilicoVA, as well as the CLF and best configuration of VAMPIRE from earlier.

is little relevant training data to extrapolate from.

In general, we notice that when we remove two of the baselines, the resulting model is stronger. We observe that baseline predictions can be noisy, since the baselines themselves have their strengths and weaknesses with predicting different labels. This noise can sometimes confuse the model enough so that it doesn’t learn an effective attention matrix when we have all four baselines. Removing two of them seems to decrease the noise and improve performance.

Both stacked VAMPIRE models are the best performing models in this data setting.

9.5.2 Few Target Domain Data

In the few TD data setting, the best stacked VAMPIRE model performs slightly worse than the best VAMPIRE + CLF model. The version of Stacked VAMPIRE taking in all four baselines as input is weaker than the version which takes in only two baselines, as in the No TD data setting. In our analysis of the baselines, we showed that the statistical models improve much less than our neural

Model	More TD Data
InSilicoVA (TD)	0.405 _(0.00)
CLF (TD + NTD)	0.468 _(0.03)
+ VAMPIRE (TD + NTD)	0.482 _(0.03)
Stacked VAMPIRE (Attn)	0.438 _(0.02)
Stacked VAMPIRE (Attn, 2B, Dropout)	0.454 _(0.01)

Table 9.3: F_1 results for Stacked VAMPIRE in the More Target Domain Data setting. We include our strongest baseline, InSilicoVA, as well as our strongest configurations of the CLF and VAMPIRE.

models when trained on additional data. Since stacked VAMPIRE depends on the performance of the statistical baselines, this ensembling method is weaker when our baselines are weaker and the neural models are themselves stronger. It makes sense that when we remove two of the baselines, we reduce the amount of noise from mis-classified labels in the baselines, and as a result, model performance increases slightly, but not more than VAMPIRE + CLF.

9.5.3 More Target Domain Data

In the More TD data setting, the results follow from our observations in the previous setting. Since the baseline models do not improve as much, the best stacked VAMPIRE is weaker than VAMPIRE + CLF (by about 3 F1 points). Our stacking procedure treats the inputs from the four baselines and the CLF predictions with equal weight to begin with, so learning that the CLF predictions are more valuable in the higher resource settings is more difficult when the baselines more heavily skew the predictions away from the correct label. It’s clear that as we increase the amount of data available to us, VAMPIRE + CLF is the best model to use.

Chapter 10

LARGER PRE-TRAINED MODELS

10.1 Motivation

Our stacked VAMPIRE model appears to perform quite well in several data settings. In fact, all our neural models (CLF, VAMPIRE + CLF, stacked VAMPIRE) consistently outperform the baseline models in the data settings we've explored. Perhaps a natural follow-up question would be, do larger neural models outperform our lightweight ones? Research groups have proposed variants of BERT, a popular transformer-based model, in which they modify the transformer architecture, change the pretraining data, or prescribe novel training setups. One of these methods is RoBERTa, which utilizes the same Transformer architecture as BERT, but is pretrained for a longer amount of time on more data and has more parameters (Liu et al., 2019). Our goal is to understand whether these models have capabilities beyond VAMPIRE for Verbal Autopsy. We hypothesize that these models might struggle with our formulation of the VA data. We interpret the data as non-sequential, but RoBERTa and its variants, in their original forms, use positional embeddings.

At the same time, our analysis of BioMed RoBERTa can help us explore how different domains outside of Verbal Autopsy are related to VA. We assume that biomedical research papers are a parent domain of Verbal Autopsy data, but this may not be true.

10.2 Experiment Details

In this experiment we consider two models: RoBERTa and BioMed RoBERTa. For the former, we train a downstream classifier on top of a saved pretrained model, as provided by the HuggingFace interface. For BioMed RoBERTa, however, we continue pretraining the existing RoBERTa model and then train a downstream classifier on top of that language model, using the DAPT + TAPT method explored by Gururangan et al. (2020). In a way, these two models represent the range of usages for these larger pre-trained models: directly using weights all the way to additional fine-tuning of the underlying language model.

Model	No TD Data
Tariff	0.353 _(0.03)
CLF (NTD)	0.333 _(0.01)
+ VAMPIRE (TD + NTD)	0.348 _(0.02)
Stacked VAMPIRE (Attn)	0.368 _(0.02)
Stacked VAMPIRE (Attn, 2B, Dropout)	0.377 _(0.01)
RoBERTa (seed = 25)	0.160 _(0.01)
BioMed RoBERTa (DAPT + TAPT) (seed = 42)	0.151 _(0.00)

Table 10.1: F_1 results for RoBERTa and BioMed RoBERTa (DAPT + TAPT) in the No Target Domain Data setting. We also include our previous best performing models.

We consider the exact same three data settings used in our advanced semi-supervised chapter. In these data settings, RoBERTa operates like one of our supervised methods, we perform no additional pretraining for the model, and BioMed RoBERTa operates like an alternative to our VAMPIRE and Stacked VAMPIRE setups.

10.3 Training Details

Prior work has shown that 1) the initialization of large pretrained BERT-like models heavily influences the performance of the model (Dodge et al., 2020) and 2) these larger models should be trained for several more epochs on smaller datasets (Gururangan et al., 2020). For these reasons, we run our experiments with multiple seeds (0, 8, 25, 42). In preliminary experiments, we found that there wasn't any significant difference between training the downstream classifiers in these models for 100 epochs and 25 epochs, so in the interest of time, we fine-tune all classifiers for 25 epochs. The BioMed RoBERTa models are additionally pre-trained for 5 epochs on the specified pretraining data. All versions of RoBERTa and BioMed RoBERTa are from the HuggingFace repository and the fine-tuning code utilizes the SimpleTransformers API (Wolf et al., 2019).

10.4 Results

In all result tables, we report performance for the best seed (indicated in the table) for each of the RoBERTa models we evaluate.

Model	Few TD Data
InSilicoVA (TD)	0.398 _(0.00)
CLF (TD + NTD)	0.454 _(0.01)
+ VAMPIRE (TD + NTD)	0.473 _(0.03)
Stacked VAMPIRE (Attn)	0.4561 _(0.03)
Stacked VAMPIRE (Attn, 2B, Dropout)	0.4651 _(0.02)
RoBERTa (seed = 42)	0.230 _(0.01)
BioMed RoBERTa (DAPT + TAPT) (seed = 25)	0.250 _(0.01)

Table 10.2: F_1 results for RoBERTa and BioMed RoBERTa in the Few Target Domain Data setting. We also include our previous best performing models.

Model	More TD Data
InSilicoVA (TD)	0.405 _(0.00)
CLF (TD + NTD)	0.468 _(0.03)
+ VAMPIRE (TD + NTD)	0.482 _(0.03)
Stacked VAMPIRE (Attn)	0.438 _(0.02)
Stacked VAMPIRE (Attn, 2B, Dropout)	0.454 _(0.01)
RoBERTa (seed = 42)	0.238 _(0.01)
BioMed RoBERTa (DAPT + TAPT) (seed = 0)	0.268 _(0.02)

Table 10.3: F_1 results for RoBERTa and BioMed RoBERTa in the More Target Domain Data setting. We include our previous best performing models as well.

10.5 Analysis

The macro F1 scores in all settings indicate that RoBERTa and BioMed RoBERTa struggle with our formulation of the Verbal Autopsy data. The differences between RoBERTa and BioMed RoBERTa are minor in the No Target Domain data setting. RoBERTa is very slightly better than BioMed RoBERTa. In the later two settings, however, it appears that fine-tuning the BioMed RoBERTa language model leads to larger gains over RoBERTa. This suggests that if these larger models were to be used, a more elaborate pretraining scheme could be useful. This could also suggest that biomedical research papers are not the best parent domain of Verbal Autopsy data when it comes to DAPT.

One key reason for these models failing is that our formulation of VA data is inherently non-sequential. There is no meaning to the order of the terms in our data. RoBERTa and BioMed RoBERTa are sequential models, and as a result, aim to find meaning in the order of the tokens. This clearly hinders performance on this particular task, but is another reason why a Bag-Of-Words approach such as VAMPIRE might be superior.

Chapter 11

CONCLUSION

Given the three resource settings and all the models and their variants we have considered in this report, we provide model and training recommendations for each scenario.

11.1 No Target Domain Data

In the lowest resource setting, stacking with VAMPIRE is the best method. The baselines, while sometimes noisy and inconsistent, carry valuable information about data points for which we have very little training support. Despite not ever seeing any data from Dar, the baselines are able to use symptom-cause information to make strong predictions. We already demonstrated that VAMPIRE and, in general, neural methods are strong in this setting, it makes sense that combining the strengths of the baselines and the neural methods with stacking produces the best performing model.

11.2 Few Target Domain Data

In this setting, VAMPIRE + CLF is the best performing model, closely followed by one of our stacked VAMPIRE configurations. Even when we analyze the break down of the performance per label, the two models perform very similarly. It appears that when we provide a small amount of labeled in-domain data, VAMPIRE and the downstream CLF are able to extrapolate most successfully from the data.

11.3 More Target Domain Data

Once again, in the highest resource setting, VAMPIRE + CLF is the best performing method, with the stacking ensemble performing slightly worse again. When we have access to more in-domain labeled data, we recommend using VAMPIRE.

11.4 Discussion

As explored and concluded in previous work, we find that labeled target domain data is the most useful type of data. All models benefit from adding labeled TD data, as we do in our second and third data settings, but the neural models benefit the most. We find that labeled non-target domain data is also useful in large quantities. Our analysis in Chapter 7 suggests that training on a small amount (approx. 500 examples) of labeled target domain data is about the same as training on a larger amount (approx. 6000 examples) or labeled non-target domain data. However, when we combine the two, our neural models perform the best. In general, we find that pretraining on unlabeled data improves performance over a supervised classifier. In particular, VAMPIRE responds the best when we pretrain and train on all available unlabeled and labeled data. Due to the limited amount of data in our study, we emphasize that VAMPIRE’s performance is a lower bound. There is room for improvement. We also find that stacking VAMPIRE is useful in the case where we do not have labeled target domain data. In this case, it’s helpful to query baselines, which have access to symptom-cause information. Finally, we show that larger pretrained models struggle with this non-sequential formulation of the Verbal Autopsy task. While DAPT + TAPT with BioMed RoBERTa outperforms the out-of-the-box RoBERTa model in the later two data settings, both models are nowhere near the performance of VAMPIRE.

We find that it is possible to achieve some amount of domain transfer when training on sites that are different from the target domain. With some additional expert information, we can develop a strong model for this data resource setting. This suggests that different sites in the VA dataset might be more closely related. Based on the general success of BioMed RoBERTa over RoBERTa, we can deduce that biomedical research papers are closer to the domain of Verbal Autopsy data than general English data, such as Wikipedia. The performance of DAPT + TAPT in Chapter 10 suggests that the domain of biomedical research papers has some overlap with the Verbal Autopsy domain.

Chapter 12

FUTURE WORK

As established in Chapter 11, the performance of VAMPIRE in this paper is a lower bound. We show that VAMPIRE improves with more labeled and unlabeled data, and we hypothesize that adding more data will be helpful. This data could include additional unreleased Verbal Autopsy data from this study or a similar study, such as the Million Deaths Study ([Centre for Global Health Research and of Toronto, 2003 \(accessed June, 2020\)](#)). In Chapter 10 we explored how applying the DAPT + TAPT pretraining technique to RoBERTa performs on the Verbal Autopsy task. While this method wasn't successful, we hypothesized that it was because of the non-sequential formulation of our data. In the future, we can explore DAPT + TAPT for VAMPIRE, which we have already shown to be successful on this dataset. We can also explore a more sequence-based interpretation of the VA data. In a few cases, the order of the questions could be significant ("Did the decedent experience bleeding?", "For how long?"), so a sequential framing might be helpful. Similarly, we can repeat our RoBERTa and BioMed RoBERTa experiments and remove their positional embeddings to better understand how larger models perform on bag-of-words data.

We begin to explore the idea of domain transfer in Verbal Autopsy by training on non-target and target domain data separately. While we find that training on target domain data is the most useful, there are still open questions about the non-target domain data. In particular, are some sites better for domain transfer than others? How do we quantify domain similarity between sites? Can we leverage this idea to get better domain transfer?

In this work, we explore one method of interpreting survey data for neural models. However, our method is fairly naive. Concatenating the questions and answers might have worked well for Verbal Autopsy because the range of answers is relatively small (Yes/No/Don't Know). Does this method work well for more open text responses? Future work might consider alternative methods of encoding survey data.

BIBLIOGRAPHY

- Leo Breiman. 1996. Stacked regressions. *Machine Learning*, 24:49–64. [9.1](#)
- Peter Byass, Daniel Chandramohan, Samuel J. Clark, Lucia D’Ambruoso, Edward Fottrell, Wendy Jane Graham, Abraham J. Herbst, Abraham Hodgson, Sennen Houesse Hounton, Kathleen Kahn, Anand Krishnan, Jordana Leitao, Frank O. Odhiambo, Osman Sankoh, and Stephen M. Tollman. 2012. Strengthening standardised interpretation of verbal autopsy data: the new interva-4 tool. *Global Health Action*, 5. [1](#)
- Samuel J. Clark, Zehang Richard Li, and Tyler H. McCormick. 2018. Quantifying the contributions of training data and algorithm logic to the performance of automated cause-assignment algorithms for verbal autopsy. *arXiv: Applications*. [1](#), [3](#), [3.6](#), [5](#), [6.4](#), [6.5](#)
- Samuel J. Clark, Tyler H. McCormick, Zehang Richard Li, and J. C. Wakefield. 2015. Insilicova: A method to automate cause of death assignment for verbal autopsy. *arXiv: Other Statistics*. [3.3](#)
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. *ArXiv*, abs/1810.04805. [1](#), [4](#)
- Jesse Dodge, Gabriel Ilharco, Roy Schwartz, Ali Farhadi, Hannaneh Hajishirzi, and Noah A. Smith. 2020. Fine-tuning pretrained language models: Weight initializations, data orders, and early stopping. *ArXiv*, abs/2002.06305. [6.5](#), [10.3](#)
- Edward Fottrell, Peter Byass, Thomas W Ouédraogo, Cécile Tamini, Adjima Gbangou, Issi-aka Sombié, Ulf Högberg, Karen H Witten, Sohinee Bhattacharya, T Desta, Sylvia Deganus, Janet Ansong Tornui, Ann E. Fitzmaurice, Nicolas Meda, and Wendy Jane Graham. 2007. Revealing the burden of maternal mortality: a probabilistic model for determining pregnancy-related causes of death from verbal autopsies. *Population Health Metrics*, 5:1 – 1. [1](#)
- Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu,

- Matthew E. Peters, Michael Schmitz, and Luke Zettlemoyer. 2018. Allennlp: A deep semantic natural language processing platform. *ArXiv*, abs/1803.07640. 3
- St. Michael's Hospital Centre for Global Health Research and University of Toronto. 2003 (accessed June, 2020). *Million Death Study (MDS)*. 12
- Suchin Gururangan, Tam Dang, Dallas Card, and Noah A. Smith. 2019. Pretrained vae vampire embedding unlabeled text word frequencies vae word frequencies mlp mlp label. (*document*), 1, 3.5, 4, 4.1, 4.1.2, 7.1, 8.1, 9.3.3
- Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. *ArXiv*, abs/2004.10964. 1, 4, 4.2, 4.2.2, 5, 10.2, 10.3
- Spencer L. James, Abraham D. Flaxman, and Christopher JL Murray. 2011. Performance of the tariff method: validation of a simple additive algorithm for analysis of verbal autopsies. *Population Health Metrics*, 9(1):31. 3.1
- Diederik P. Kingma and Max Welling. 2014. Auto-encoding variational bayes. *CoRR*, abs/1312.6114. 4.1.2
- T. McCormick Li, Z. and S. J. Clark. 2018 (accessed Sept, 2019)a. *InSilicoVA: Probabilistic Verbal Autopsy Coding with 'InSilicoVA' Algorithm*. 3
- T. McCormick Li, Z. and S. J. Clark. 2018 (accessed Sept, 2019)b. *InterVA4: Replicate and Analyse 'InterVA4'*. 3
- T. McCormick Li, Z. and S. J. Clark. 2018 (accessed Sept, 2019)c. *Tariff: Replicate Tariff Method for Verbal Autopsy*. 3
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pre-training approach. *ArXiv*, abs/1907.11692. 1, 4, 4.2, 4.2.1, 10.1
- Kyle Lo, Lucy Lu Wang, Mark E Neumann, Rodney Michael Kinney, and Daniel S. Weld. 2019. S2orc: The semantic scholar open research corpus. *arXiv: Computation and Language*. 4.2.2

- André F. T. Martins, Dipanjan Das, Noah A. Smith, and Eric P. Xing. 2008. Stacking dependency parsers. In *EMNLP*. 9.1, 9.3.1
- Trieu H. Trinh and Quoc V. Le. 2018. A simple method for commonsense reasoning. *ArXiv*, abs/1806.02847. 4.2.1
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *ArXiv*, abs/1706.03762. 4.2.1
- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R’emi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface’s transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771. 10.3
- David H. Wolpert. 1992. Stacked generalization. *Neural Networks*, 5:241–259. 9.1
- Yukun Zhu, Ryan Kiros, Richard S. Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 19–27. 4.2.1