Detecting Protected Health Information with an Incremental Learning Ensemble: A Case Study on New Zealand Clinical Text

Balkaran Singh*, Quan Sun*, Yun Sing Koh†, Junjae Lee*, Edmond Zhang*
*Orion Health, Auckland, New Zealand
{balkarans,quans,junjael,edmondz}@orionhealth.com
†University of Auckland, Auckland, New Zealand
ykoh@cs.auckland.ac.nz

***Abstract*—Clinical narratives host vast accumulations of patient data pivotal for research and development of health related products. In order for this data to be utilized, the underlying protected health information needs to be de-identified to ensure medical confidentiality. Given the voluminous size of clinical texts, manual de-identification of such large datasets is both expensive and impractical. Therefore, the concept of automated de-identification is a highly appealing prospect. Machine learning or model based sequential labeling algorithms, such as the named entity recognition algorithms and rule-based algorithms are among the most effective approaches to automated de-identification. A natural question to ask is how we can combine them to have the best of both worlds. In this paper, we present an analytical and easy to interpret framework to dynamically combine a sequential labeling model and a soft-rule-based model in an incremental learning setup. This framework is applied to a case study, which is part of a project prototyping automated de-identification system for New Zealand clinical free text data. Evaluations show that our approach can accommodate changes in the incoming data through dynamic updating. The simplicity of the framework also allowed us to gain insights on behaviour e.g. change of importance between the machine learning and rule models.**

***Index Terms*—Incremental Learning, Ensemble Learning, De-identification, Data Privacy**

I. Introduction

In the contemporary age, as health related services have progressed, the vast accumulation of patient data is preserved in the form electronic health records (EHRs). The pervasive amounts of protected health information (PHI) within EHRs undermine their ability to be utilised in down-stream applications such as research and development. To maximise their utility, PHIs within EHRs are required to be de-identified such that patients cannot be re-identified from the data. The manual process de-identification is prohibitively time consuming and uneconomic, hence, entailing the need for automated de-identification. While during the course of PHI identification for removal, it is highly necessary for a de-identification process to retain the medical contents of the records so that this information can help further research and conserve the value of the record.

Developing data science capacity for applications in healthcare requires extra consideration with regard to data privacy. The data, usually in the form of structured databases, text documents, images, needed for building an analytical system or predictive model is not allowed to be used directly until a sophisticated de-identification has been applied or unitl individual consents of patients are obtained. Sometimes, both are required. Taking a natural language processing (NLP) project as an example, the de-identification process of the training (text) data usually involves recruiting annotators with medical background to identify PHI, which is time consuming and expensive. Once the data is de-identified and its re-identification risk is reviewed. The data may be released to be used in downstream research and development. The whole re-identification process can take weeks or months depending upon the size of the data.

In recent years, the demand for an automated de-identification system is increasing, at the same time, machine learning based NLP techniques have improved substantially, e.g. named entity recognition (NER) algorithms developed for sequential labeling tasks, achieved tremendous accuracy in terms of detecting entities in free text data. However, accuracy of a NER model built on one dataset, is expected to decline on another dataset in a different domain. For instance, one could build a NER model with a publicly available dataset, e.g. the i2b2 clinical dataset [1], but the performance is expected to decline if the model is used for making predictions on a New Zealand clinical dataset.

The main reasons attributing to performance deterioration are domain drift and lack of localisation. There is no guarantee that the pre-built model would work equally well, in terms of both accuracy and re-identification risk on a new dataset as they do not share a reasonable amount language level properties [2]. Localisation is another issue, where local vocabulary can never be learned if they were not in the original training data. One approach to overcome the above issues is using transfer learning [3], [4]. In this learning paradigm, models built on one dataset can be updated (re-trained) using some new data from the target dataset/domain. However, transfer learning may still require a significant amount of new data to be able to adapt to new concepts/entities.

In this research, we discuss the outlined challenges by proposing a simple framework for dynamically combining machine learning and rule-based models in an incremental (online) learning setup, which can be seen as a complement to conventional transfer learning given a relatively small amount of new data. The learning model is used in a case study of developing a de-identification system, where the goal is to help clinicians to detect PHI in free text data more easily.

We propose to form an ensemble of a pre-trained machine learning model and a rule-based model. The machine learning model (pre-trained on a publicly available US dataset) is expected to work well on the known patterns it has seen at the semantic level, whereas the rule-based model (to be built incrementally with new data) will help the localization issue that the machine learning will suffer, for instance, Māori names of towns and cities across New Zealand, NZ specified pre-fix IDs, and special words being used in a hospitals. The crux is how to combine them in such a way that the system is able to adjust given the performance of both models on recent predictions. We discuss the proposed ensemble framework further in section III.

II. Recent Work

The most common approach to de-identification is sequentially evaluating each token to determine the presence of any protected information. This is in close proximity to named entity recognition which also focuses on the identification of pertinent information. By the means of substantial developments in neighbouring fields such NLP and deep learning, de-identification has also significantly evolved from relying on handcrafted rules and semantic dictionaries. Machine learning approaches to de-identification are more general in the sense, that they perform better in identifying PHI's not included in dictionaries. Several machine learning approaches have proposed including Conditional Random Fields (CRFs) [5] and its Deep Learning powered variants [6]–[8], Hidden Markov Models (HMMs) [9] and Support Vector Machines (SVMs) [10]. As noted through several studies, CRFs are a prominent approach in identifying PHIs and sequential labelling tasks in general [11]. CRFs are from a family of probabilistic graph models which aim to find the conditional probability of an output vector (labels) given an input vector of observed data. The conditional nature of CRFs provide a theoretical advantage over other probabilistic graph models such as HMMs which assume feature independence. Empirical testing in de-identification tasks have also shown that CRFs generally out-perform other approaches [12]. A critical step in developing machine learning methods for de-identification is the extraction of linguistic/lexical features which represent a word through several binary, nominal or numeric values. Lexical features include part of speech tags, prefixes, suffixes, lowercase token for each word etc. Other, orthographic features such as the shape and length of tokens, whether the token contains digits or letters are also often considered.

Recent deep learning methods have shown remarkable de-identification ability without requiring manually crafted features. A key attribute to this impressive break-through is the distributed representation of words through word embedding algorithms. Word embedding techniques provide individual representations of words as dense real valued vectors associated with specific points in a vector space. Such representations of features, mitigate the need of human intervention for manual feature engineering and extraction. The first de-identification system based on deep learning using word embedding showed state-of-the-art results on the de-identification of i2b2 2014 and the MIMIC datasets, outperforming the previous benchmark achieved by CRFs [13]. The deep learning architecture was based on a type of recurrent neural network, the bi-directional Long Short-Term Memory (LSTMs). More recent work, has evaluated this approach in a cross-institute setting where models trained on the i2b2 corpus were tested on other corpuses after merging data and fine-tuning models [14].

Combining outputs from different de-identification algorithms is an attractive prospect envisioned through ensemble learning. Admittedly, de-identification literature is still lacking in studies involving ensemble learning, although recently, interesting methods have emerged. Most notably, CRF and BI-LSTMs trained using feature extraction and word embeddings were combined using a stacking classifier (SVM) [15]. Moreover, rule-based methods were used to directly extract some identifiers, which were merged into the labelled sequence after stacking. A recent comparative study has explored ensembles of multiple de-identification methods including deep learning, shallow learning and rule-based methods [16]. The ensemble methods included stacking and voting which consistently outperformed individual algorithms.

In general terms, incremental learning refers to learning data as it becomes available in batches. The goal is to confront with such non-stationary situations through adaptive mechanisms to accommodate changes in the incoming data. A typical approach for learning such information involves discarding the previous classifier and retraining as new data becomes available. This approach is known as catastrophic forgetting as previously acquired knowledge is forgotten and the models are retrained. In contrast, adaptive mechanisms aim to continually accommodate new data into the trained classifiers. Often times, this is addressed through adaptability capacity in the classifiers and adaptability in contribution of each classifier in an ensemble.

The available literature for adaptive learning in named entity recognition is quite sparse and virtually non-existent in de-identification terms. Recent work in NER has utilized incremental learning for updating deep learning models in a active learning framework [17]. Since the approach involving catastrophic forgetting is computationally expensive, new incoming data was concatenated to the older samples and the models were retrained for a small number of epochs. Other works, outside the domain of natural language processing explore 'multi-level' adaptive systems, where incremental learning classifiers are combined with adaptive ensembles [18]. This is a hybrid approach where new learners are added over time and the inefficient existing learners are removed. The outputs of active learners are combined using weighted voting. This type of adaptive behaviour allows classification systems to cope with changing environments and handle concept drift. In the deep learning field, incremental learning can be achieved through updating the weights gradually given new data. Common strategies include external memory, constraints-based methods and model plasticity [19].

III. Incremental Learning Ensemble for Data De-identification

In this section, we describe the ensemble framework and the incremental learning setup we used in this research. One goal of our project was being able to show the power of combining machine learning and rule based models in a relatively easy to interpret way. The reason is that in healthcare, we have cross-disciplinary teams, consisting of computer scientists, clinicians, software engineers and project managers. So as a pilot project, we tried to demonstrate the core idea of ensembling and show the team the potential of more advanced algorithms.

The generalisation ability of any algorithm is acutely dependent upon the availability of an adequately representative training set. In de-identification tasks, it is often hard to generalise the ability of models trained public datasets to smaller, local datasets. Moreover, the availability of local datasets is irregular and slow due to manual annotations required for training and pre-processing. Therefore, it is not unusual for these datasets to become available in batches over time. Depending upon the application it may be unfeasible/improvident to wait for the entire data to become available. Under such circumstances it is more compelling to start training a system and perform incremental updates as more data becomes available over time. It is also possible that the performance of different models change with respect to newly available data. As such, it is desirable to develop a system which dynamically adjusts given new training instances. In this paper, we convey this idea through combining a machine learning and a soft rule-based model in an incremental ensemble learning setup. The approach seeks to find an optimal combination of the two models at every increment as new training data becomes available.

A. Application of an Easy to Interpret Linear Ensemble

As a first step in developing the desired approach, we examine and select a suitable combination function for the base classifiers. In classification, most combination methods belong to a family of voting techniques which seek to elect the most suitable label given predictions from base classifiers. The most common approach is to take a majority vote, where every classifier votes for a single class label and final selected label is the one that received the most votes. In our case, it is more practical to utilize something called the soft voting, particularly because our classifiers output k dimensional vectors of class probabilities for each instance x, where k is the number of classes. In its plain form, soft voting generates an output vector by simply averaging all individual outputs from the base classifiers. In this case, we consider a more generic combination where each classifier has some weight, which is to be dynamically adjusted with new increments. Mathematically, we can represent this idea through the following linear combination function.

$$\begin{aligned} f^c(\mathbf{x}) = w f_1(\mathbf{x}) + (1-w) f_2(\mathbf{x}) \\ w \in [0,1] \end{aligned} \tag{1}$$

Where f_1 and f_2 are the two individual learning algorithms and w is the associated weight. We have data of the form $D = \{(\mathbf{x}_1, y_1), ..., (\mathbf{x}_n, y_n)\} \subset X \times Y$, where each $\mathbf{x}_j \in R^d$ is a vector of attributes and $y_j \in Y$ is its corresponding label. f_1 and f_2 are trained through supervised learning where we seek to learn the function $f : X \to Y$, by minimizing a certain criterion. Generally, for the range Y, $y \in Y$ for a particular instance $\mathbf{x_j}$ is a probability vector represented as $f(\mathbf{x}_j) = [p_1, p_2 ... p_k]$, where k is the number of classes. The reason for selecting a linear form for the combination function f^c, is because we needed an ensemble strategy that can easily be explained (to the clinicians and developers) and implemented in a relatively transparent way. Thus, applying the most advanced ensemble strategy, such as boosting and stacking [20], [21] in order to get the best accuracy is not the most important goal for this project.

The intuition behind this formulation is to constraint the weights such that, their sum is always equal to one and each weight is between 1 and 0. This allows us to see the changes of importance over time in an incremental setup. Mathematically, the goal is to find the optimal weight which in turn minimizes the generalisation error of the ensemble pair, this can be achieved through empirical risk minimisation through a loss function of the form $(f(\mathbf{x})^c, y) \to l(f^c(\mathbf{x}), y) \in R$, which maps the effect of having different weights values onto a real number representing some cost. Similar results in the classification/regression context have been discussed extensively in [22], and recently in [23] and [24]. We employ similar mathematical techniques and apply it in the sequential labeling context. Therefore, the goal is to find the optimal weight parameter which minimizes a cost. The log loss or the cross entropy loss is commonly used in many learning algorithms such as logistic regression, gradient boosting, artificial neural networks etc. This is represented as:

$$\text{Log loss} \quad = -\sum_{i=1}^{N}\sum_{k=1}^{K} y_{ij} \log(f^c_{ij}) \tag{2}$$

A drawback with using the log loss function for optimising our combination function is encountered when trying to solve for a closed form solution of the weights. As it turns out, the derivative of this function is a rational function (please see a detailed derivation in Appendix B).

$$0 = -\sum_{i=1}^{N}\sum_{k=1}^{K} \frac{y_{ij}(f_{1ij}) - f_{2ij}))}{w f_{1ij} + (1-w) f_{2ij}} \tag{3}$$

For which, there is no analytical solution as soon as $N \geq 5$. The numerator is a N degree polynomial and as such it cannot be solved using radicals. This is shown through the Abel-Ruffini theorem which naively states that at degrees > 4 a general formula gets to complicated to solve, this can be shown through Galois theory. In such cases we need numerical optimisation to solve for the roots, which in most cases is perfectly feasible. However, want to find a simple closed form solution for the weights, which can be efficiently used for incremental updating.

As an alternative, we consider the Brier Scoring function as a possible candidate for calculating the loss. A version of the brier score in multi-class problems is formulated as:

$$\text{Brier Score} = \sum_{i=1}^{N}\sum_{j=1}^{K}(f_{ij}^{c} - y_{ij})^2 \tag{4}$$

As indicated in the expression above, the Brier score measures the mean squared difference between the predicted probabilities for some instance i and the actual value. Using this loss function, we can formulate the objective function for our minimisation problem as follows:

$$\text{arg min}_w = \sum_{i=1}^{N}\sum_{j=1}^{K}((wf_{1_{ij}} + (1-w)f_{2_{ij}}) - y_{ij})^2 \tag{5}$$

Its worth noting, that by using $1 - w$ for f_2 we have also reduced the optimisation problem to a single variable. Furthermore, this also acts as constraint on the function, deterring the need to explicitly define the weight constraints. As such, despite being a constrained problem, we are able to minimise this objective function without using Lagrange multipliers or the KKT strategy. Taking the derivative of the function above w.r.t w and solving for the roots, yields the closed form solution for the weights:

$$w = \frac{\sum_X \sum (y - f_2(\mathbf{x})(f_1(\mathbf{x}) - f_2(\mathbf{x}))}{\sum_X \sum (f_1(\mathbf{x}) - f_2(\mathbf{x}))^2} \tag{6}$$

A detailed derivation of the above solution can be found in Appendix A. Optimising the Brier score for estimating parameters is not very common in machine learning tasks. There are two main reasons for this, particularly, the log loss incurs a higher penalty when low probabilities are predicted for the correct class. Also, in the case of binary logistic regression the sigmoid function with the squared loss results in a non-convex objective function which can be hard to optimise. However in our case, the objective function is still convex and showed similar optimal w to the log-loss. Moreover, there is a close relationship between the log loss and the brier score, which is revealed through the Taylor series expansion of the losses.

$$\begin{aligned}
BS &= (y-1)^2 - 2(y-1)(x-1) + \frac{2(x-1)^2}{2!} \\
&= (x-1) + \frac{2(x-1)^2}{2!} \\
\textit{Log Loss} &\approx \sum_{n=1}^{\infty}(-1)^n \frac{((n-1)!)y(x-1)^n}{n!} \\
&= y(x-1) + \frac{y(x-1)^2}{2!}... \\
&= (x-1) + \frac{(x-1)^2}{2!}... \\
\text{Log Loss} &\approx \frac{1}{2}BS + c
\end{aligned} \tag{7}$$

By expanding the second degree Taylor polynomial for the brier score and Taylor series expansion for the log loss (centered around 1) and having the constant c to account for the remaining terms in the Taylor series. We can observe that the relationship between the two losses is approximately linear, please see Eq. (7).

B. *Incremental Learning*

Our motivation for incremental learning has been previously discussed. In this section, we demonstrate our approach to dynamically adjust the ensemble weights as new training samples arrive. The first phase is to train the base learners f_1 and f_2. Not all algorithms have support for incremental learning, i.e. learning without seeing all examples at once. In such cases, one approach is to concatenate new samples to the previous training data and retrain the model every time new data arrives. However, in the case of deep learning algorithms, this approach can be very time consuming. Given the scope of our project, as our goal is to study the effectiveness of the ensemble, we take a model-agnostic approach with the only restriction being the ensemble consisting of a model based and a rule based algorithm. As such, f_1 can be a relatively heavy static algorithm trained on a public dataset, incrementally updated with new data upon the arrival of new samples. Here, f_2 can be a soft-rule-based online learning algorithm trained incrementally with mini-batches of NZ dataset. Following this, at every increment, we calculate and update the weight w using Eq. (6). There are several implementation options for estimating the weights in an incremental learning setup, we can have an entirely separate validation set, used for predictions at every increment. Alternatively, new incoming batches can also be used as validation sets and in subsequent iterations, they can be used for training the classifiers. For our task at hand, we have considered the latter approach to maximise the utility of all available data.

IV. Experiments and Evaluation

A. *Experimental Setup*

Our evaluations of the prescribed methodology are performed on the publicly available i2b2 dataset and a local dataset collected in New Zealand (NZ).

The i2b2 dataset is a voluminous collection of clinical narratives introduced by the Informatics for Integrating Biology

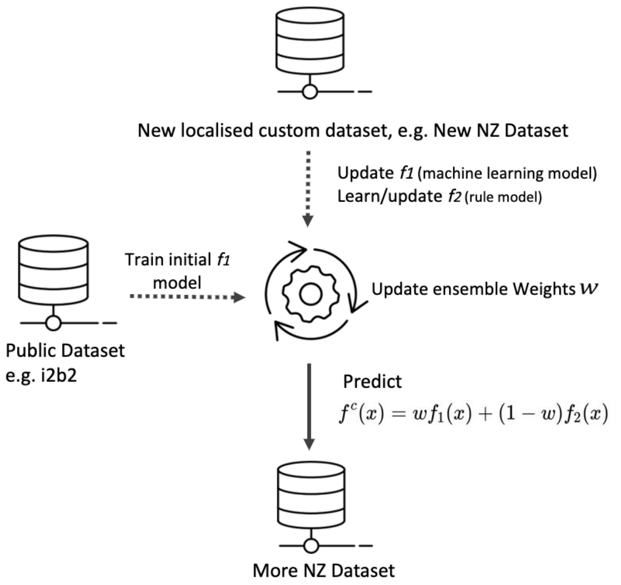
New localised custom dataset, e.g. New NZ Dataset
Update f_1 (machine learning model)
Learn/update f_2 (rule model)
Train initial f_1 model
Update ensemble Weights w
Public Dataset e.g. i2b2
Predict
$f^c(x) = wf_1(x) + (1-w)f_2(x)$
More NZ Dataset

Fig. 1: Illustration of the proposed incremental learning setup

and the Bedside centre [1]. In total, there are 658,987 word tokens, 32,759 sentences and 25 entity types, inclusive of the twelve types as defined by the Health Insurance Portability and Accountability Act (HIPAA) [25].

The NZ data was collected through a partnership with a local district health board (DHB), with the goal of creating a dataset for building a de-identification tool that highlights PHI in chucks of text in the portable document (PDF) and Word document formats. We were able to obtain a set of documents prior to the COVID-19 lockdown in New Zealand. Although the plan was to create 20 documents, our clinical partners had to stop the task as the team needed to re-schedule work priorities due to COVID-19. About the documents we use for experiments, they are semi-synthetic (rewritten by clinicians based on original documents) communication letters between clinicians and nurses about patients with diabetes. The annotators (clinicians) applied a substitution approach to de-identification so the documents are realistic enough to be used for our research.

Next, we describe how we converted the documents. The NZ dataset is composed of 15 clinical documents with PHIs annotated according to the HIPAA convention. The entity types were grouped into seven main classes, Name, Profession, Location, Age, Date, Contact and ID. For consistency, the 25 entity types of the i2b2 dataset were also aggregated into the same seven categories. Tokens were also chunked according to BILOU schema, to identify when entities spans over several tokens. Using this format, the beginning token was labeled as B, tokens inside the entity span as I and the last token in line as L. Singular entity tokens were labeled as S and finally non-entities (non PHIs) as O. Henceforth, empty tokens, surrounding whitespace and other types of low level formatting was also performed to ensure dataset compatibility. Tokens were also assigned into Part of Speech (POS) categories according to the universal POS tagset. Following the analogous prepossessing, the PHI distribution for both datasets is shown in Table I.

TABLE I: Distribution of PHI types of the i2b2 and NZ datasets.

	i2b2 Dataset		NZ Dataset	
	Count	% of PHI	Count	% of PHI
Name	7546	29.72	198	42.2
Location	4326	17.03	117	24.9
Date	11609	45.72	56	11.9
Contact	557	2.19	41	8.7
Profession	356	1.40	33	7
ID	176	0.69	19	4
Age	819	3.22	5	1

Additionally, we also created second version of the datasets to solely represent direct and quasi identifiers, which is to support a slightly different de-identification use case. This distinction is attributed to the nature of the PHI, where direct identifiers are "unique" which can be used to re-identify a patient. While, quasi identifiers are not unique in themselves, they can be sufficiently correlated with other identifiers to form a unique identifier. Which again poses the risk of re-identification. In this case, name, contact and ID were labelled as direct identifiers. While, the quasi identifiers included date, location, profession and age. We refer to this set as the 'Direct/Quasi' datasets shown in Table II.

TABLE II: Distribution of PHI types in the Direct/Quasi version of the datasets.

	i2b2 Dataset		NZ Dataset	
	Count	% of PHI	Count	% of PHI
Direct	8516	33.07%	407	58.14%
Quasi	17233	66.92%	293	41.85%

For the base ensemble learners, we first examine the CRF algorithm [5] which is widely used in named entity recognition tasks. As specified earlier, CRF aims infer labels Y given a sequence of observed tokens X. The model training phase, aims to learn the distributions between $y_1, y_2 ... y_n$ and $\mathbf{x}_1, \mathbf{x}_2 ... \mathbf{x}_n$, $p(Y|X)$. The goal of inference is to determine the most likely sequence Y given X. CRFs also utilize L1 and L2 regularisation to penalize the objective function whilst training. Larger values diminish and nullify features such that the algorithm learns to generalise and does not depend on memorisation. These parameters are tuned through hyper parameter optimisation.

Rule based methods utilize dictionary lookups, pattern matching and regular expressions to identify tokens of interest. However, this approach requires linguistic domain experts to manually create pattern rules and dictionaries. Alternatively, we can also use a supervised learning algorithm to simulate the behaviour of a rule based approach. The incremental Naive Bayes algorithm [26], is a probabilistic model formulated through the bayes theorem. The goal is to determine the probability that a particular token has a label y_i given some features $\mathbf{x}_i$. We can do this by calculating the posterior probabilities given by the bayes rule:

$$P(y_i|\mathbf{x}_i) = \frac{P(\mathbf{x}_i|\mathbf{x}_i) \cdot P(y_i)}{P(\mathbf{x}_i)}$$

Under the naive assumption of conditional independence we can efficiently calculate the posteriors as the product of likelihoods and the priors. Thereafter, we can obtain the predicted labels using decision rules as illustrated in the following simplified example:

```
  If $P(phi|\mathbf{x}_i) \geq P(not - phi|\mathbf{x}_i)$
      classify as 'PHI'
  else
      classify as 'Not-PHI'
```

Where phi and $not - phi$ are the labels and $\mathbf{x}_i$ is the observed data (features). To simulate the behaviour of a rule based algorithm we extract the following features from the data.

1) **Orthographic features**: Defines word characteristics such as whether the word is all upper, mixed or lower case, contains digits, punctuation.
2) **POS tags**: Part of Speech tags for the previous, current and the next tokens.
3) **Affixes**: Suffixes and prefixes of all words from length 2 to 3.
4) **Section information**: The distance of current word from the beginning of the sentence.
5) **Bag-of-words**: lower case of the current token and stem words for the surrounding tokens.

In de-identification, there are two types of evaluations, the token level or the entity level. The difference between two methods pertains to the position constraint by the BILOU schema of particular entities. At token level evaluation, each label is evaluated as a single entity, while entity level evaluation considers the entire span of the named entity. For example, in this sample sentence "35 sample st" '35', 'sample' and 'st' are separate tokens. Evaluating this at token level, we consider predictions for each token one by one, meaning if one of these tokens is mislabelled only that particular token is penalised. However, at entity level, the entire named entity will be considered as the wrong prediction. Interchangeably, if all tokens in a named entity are correctly labelled only then it is considered the right prediction. We have preferred the latter approach as we want cover the entire span of the Protected Health Information. To evaluate the performance of our de-identification system we utilize the the F1-score metrics.

B. Ensemble Performance in a Stationary Setup

We first evaluate the proposed ensemble method in a stationary setup using the closed form solution (Eq. (6)) of the weights. Our motivation is to compare ensemble performance with individual base learners trained on different versions of the datasets. The datasets under consideration are the HIPAA and the Direct/Quasi versions of the i2b2 and NZ datasets. The datasets were split into 60% training, 20% validation, 20% test sets. The validation set was used to estimate the ensemble weights and the test set was used for final evaluation of the model performance. An example of estimating the weight on a validation set and using it for ensemble predictions on the test set is shown in the appendix section Fig 7. This experiment was repeated 20 times, the mean and the standard deviations for this are shown in Fig 2. For i2b2 datasets (HIPAA and the Direct/Quasi versions) there is disparity between the performance of CRF and incremental Naive Bayes. As a result, the ensemble F1 scores are only slightly higher in comparison with CRF (the best performing base learner). However, in the NZ dataset (HIPAA and Direct/Quasi) CRF and Naive Bayes performances are matched more closely hence, the ensemble performs better by exploiting the disagreements between the predictions. The results from Fig 2 are also tested using the Wilcoxon signed-rank test [27]. (see Appendix).

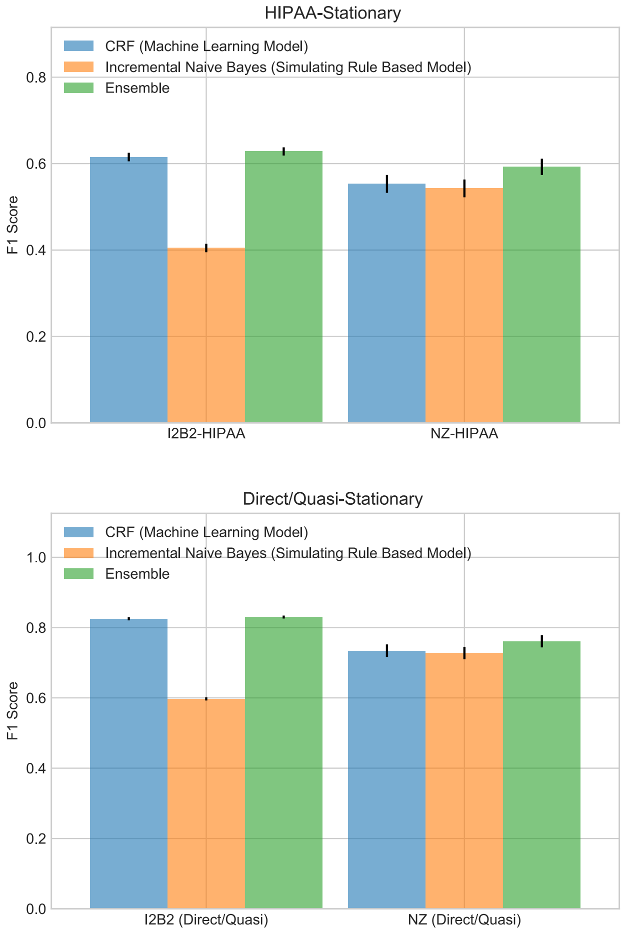

Fig. 2: Comparison of predictions from individual base learners and ensemble in a stationary setup

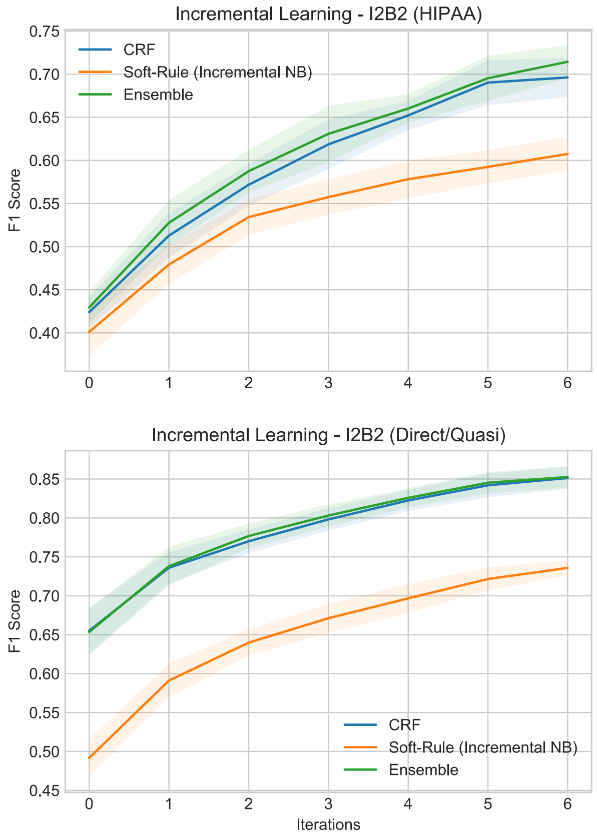

Fig. 3: F1 scores from incrementally learning the I2B2 dataset

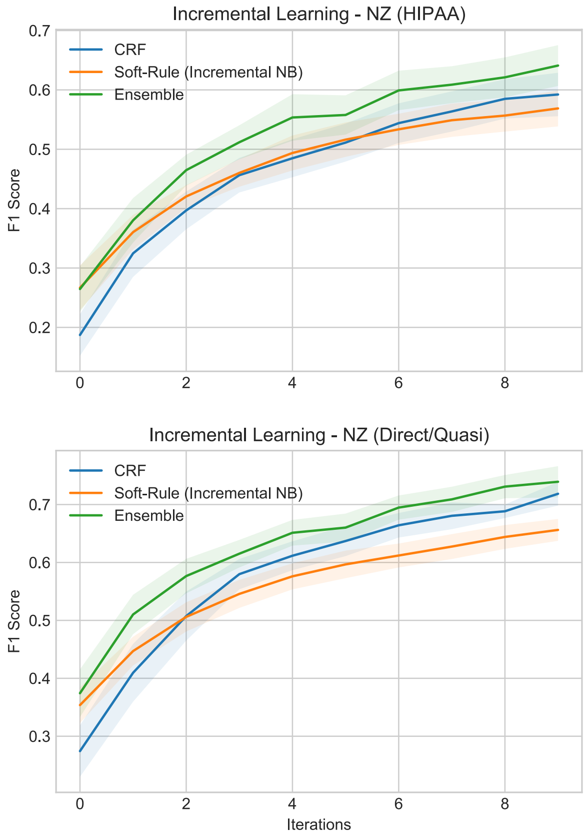

Fig. 4: F1 scores from incrementally learning the NZ dataset

C. *Incremental Learning on the i2b2 and NZ Dataset*

The incentive of this subsection is to verify, whether the ensemble performs better than the base learners in an incremental learning setup. Here, we consider both versions of the NZ and the i2b2 datasets separately. For the i2b2 datasets, we have two versions of the dataset as described earlier, namely, I2B2-HIPAA and I2B2-Direct/Quasi. In this experiment, 20% of the total observations were reserved for testing, while CRF and incremental Naive Bayes were trained on batches of 1000 sentences in every iteration. The ensemble weights were updated in each iteration using the next chunk of incoming data, not yet used for training. Finally, the estimated weight was used to combine the base learner predictions on the reserved 20% test set in every increment. The experiment is repeated 20 times and we calculate the mean and confidence intervals for the F1 scores. Figure 3 shows the mean F1 scores obtained from the ensemble predictions and the two base learners. We can observe that the ensemble F1 scores are always higher than the base learners. Although, in this case, there is not a substantial improvement as incremental Naive Bayes performs poorly compared to CRF. The performance ability of the ensemble is directly contingent upon the individual performances and the dissimilarities between the two base learners, as such, to augment the ability of the ensemble both learners need to have small error while giving different predictions from each other. For the NZ datasets (NZ-HIPAA and NZ-Direct/Quasi) we use a similar setup to the previous experiment, in every iteration, 25 sentences are added to training corpus and the models are retrained. As shown in Fig 4, In both version the HIPAA and Direct/Quasi dataset, the ensemble of base learners has a higher mean F1-score over the course of all iterations. Furthermore, we can observe that initially Incremental Naive Bayes performed better but as new data comes in CRF starts performing better, however, this does not effect the performance of the ensemble. This describes the dynamic behaviour of our proposed method, which is able to adapt and prefer an algorithm which contributes in minimizing the generalisation error.

D. *Incremental Learning on the NZ+I2B2 Dataset*

Previously we have considered the i2b2 and NZ datasets separately, for the following experiment we use i2b2 as the initial training data for f_1 (CRF) as described in Figure 1. When learning the NZ data by itself CRF initially performs worse and as more data comes in it starts to outperform incremental Naive Bayes. This drives our motivation to provide CRF with an initial boost using the i2b2 dataset (similar to the concept of transfer learning) and study how it effects the ensemble predictions. In this regard, CRF is initially trained on i2b2 and incrementally updated with batches of 30 sentences from the

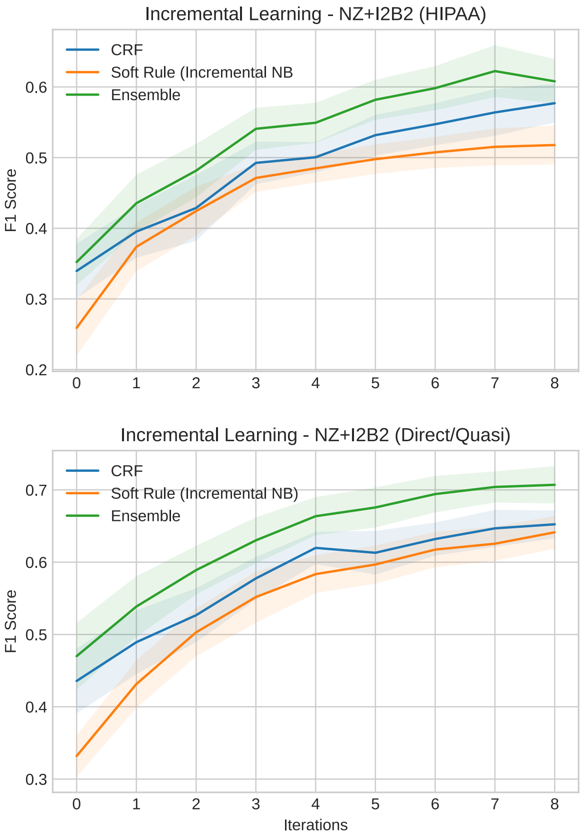

Fig. 5: F1 scores from initial training on the I2B2 and incrementally learning the NZ dataset

NZ dataset. While, incremental Naive Bayes is only trained on the NZ dataset with the same batches of 30 sentences. As with previous experiments, we implement this on both the HIPAA and the Direct/Quasi versions of the datasets.

Fig 5, shows that for the first few iterations, CRF and consequently the ensemble performance, improves with the addition of i2b2. In Fig 4 we saw that initially incremental Naive Bayes had a higher F1 score but with the inclusion of i2b2 plus more NZ data, CRF starts to perform better. In both cases, our method is able to adapt to these changes which results in higher ensemble F1-Scores. To take a deeper look at how this works, we examine the estimated weights for the ensemble outputs shown in Fig 5. The incremental weight updates with respect to f_1 (CRF) are shown in Fig 6. For the Direct/Quasi dataset, during the first few increments we see that CRF performs much better than incremental Naive Bayes and as such it incurs higher weight, as the performance of the two learners gets close, the weights decrease. When the performances of the algorithms match we see that the ensemble still performs better by taking advantage of the dissimilarities between the learners. Similarly, for the HIPAA version, when there is disparity between the performances of the base learners, more weight is given to the best performing algorithm. When the both algorithms start to perform similarly, the weights decrease. As demonstrated through this experiment, initial training on a public dataset leads better performance in the first few increments. This can be a useful feature for a de-identification tool to improve initial model performance when local data is scarce and accumulates incrementally. However, in the long run and among the final iterations, the inclusion of i2b2 does not seem improve the model performance. Overall, we have seen that the ensemble generally performs better than the individual base learners over all iterations in both versions of the datasets. The Direct/Quasi datasets always performed better in the described experimental setups, in the specific use case where it is not required to distinguish between different PHIs it is more beneficial to use this set.

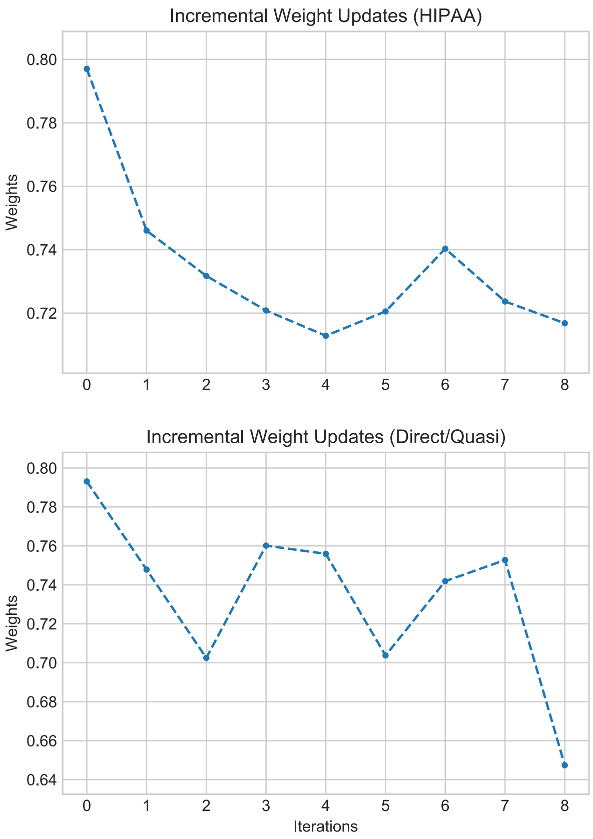

Fig. 6: Evolution of weights for f_1 over increments

V. Limitations

Although, we have presented a simple framework for incremental de-identification on NZ clinical texts, our study may be subject to potential limitations. Constrained by the time available for the task, we have considered simple lexical and orthographic features and did not exploit the full characteristics of the corpus for the soft rule leaner (simulated by Naive Bayes). With better and more general features we would expect the soft rule leaner to perform better and consequently expect the ensemble performance to ascend. To estimate the weights we have used a validation set with an assumption that it is similar to the test set. A possible downside to this is that, if the test set changes, then the validation set should also reflect these changes. In practical applications, it would be wise to reserve some documents which are very similar to the test set for the sole purpose of estimating the weights. We have only evaluated the model performances on the F1 scores. In the data privacy context, re-identification risk is probably more important in certain applications. Therefore, the performance should be evaluated on a risk score as well, a candidate is the method proposed in [28], which accurately reflect the risk of a patient being re-identified. However, the risk-based evaluation methods work on the document level, which requires more data than we currently have for the case study.

VI. Conclusion

Given the voluminous size of clinical datasets, the concept of automated de-identification is a highly appealing prospect. Machine learning algorithms and rule-based algorithms are among the most effective approaches to automated de-identification. In this paper, we discussed an ensemble framework for dynamically combining machine learning and rule-based models in an incremental learning setup, which may speed up transfer learning given a relatively small amount of new data. We provided a case study based on a New Zealand clinical dataset. Evaluations show that our approach can accommodate for changes in the incoming data through dynamic updating. The simplicity of the framework also allowed us to see and gain insights on behaviour e.g. change of importance between the machine learning and rule models. In the future, we will continue with our DHB partners on obtaining more data. Algorithm-wise, we have several planned works, including employing more advanced ensemble strategies based on the trust of our clinical advisors and developers on combining machine learning and rule-based models. Furthermore, we will try replacing the simple and easy to interpret CRF with a current state-of-art deep learning algorithm, for instance, the BERT [29] and transformer [30] powered deep learning architecture [6] with a CRF output layer.

Acknowledgement

We would like to acknowledge Precision Driven Health (PDH) for funding this work.

References

[1] A. Stubbs and Ö. Uzuner, "Annotating longitudinal clinical narratives for de-identification: The 2014 i2b2/uthealth corpus," *Journal of biomedical informatics*, vol. 58, pp. S20–S29, 2015.

[2] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *Journal of Big data*, vol. 3, no. 1, p. 9, 2016.

[3] C. B. Do and A. Y. Ng, "Transfer learning for text classification," in *Proceedings of the 18th International Conference on Neural Information Processing Systems*, ser. NIPS'05. Cambridge, MA, USA: MIT Press, 2005, p. 299–306.

[4] R. Raina, A. Y. Ng, and D. Koller, "Constructing informative priors using transfer learning," in *Proceedings of the 23rd International Conference on Machine Learning*, ser. ICML '06. New York, NY, USA: Association for Computing Machinery, 2006, p. 713–720.

[5] J. Lafferty, A. McCallum, and F. C. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," 2001.

[6] Z. Huang, W. Xu, and K. Yu, "Bidirectional lstm-crf models for sequence tagging," *ArXiv*, vol. abs/1508.01991, 2015.

[7] H. Cho and H. Lee, "Biomedical named entity recognition using deep neural networks with contextual information," *BMC Bioinformatics*, vol. 20, 2019.

[8] J. li, A. Sun, R. Han, and C. Li, "A survey on deep learning for named entity recognition," *IEEE Transactions on Knowledge and Data Engineering*, vol. PP, pp. 1–1, 03 2020.

[9] S. R. Eddy, "Profile hidden markov models." *Bioinformatics (Oxford, England)*, vol. 14, no. 9, pp. 755–763, 1998.

[10] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intelligent Systems and their applications*, vol. 13, no. 4, pp. 18–28, 1998.

[11] A. Stubbs, C. Kotfila, and Ö. Uzuner, "Automated systems for the de-identification of longitudinal clinical narratives: Overview of 2014 i2b2/uthealth shared task track 1," *Journal of biomedical informatics*, vol. 58, pp. S11–S19, 2015.

[12] B. He, Y. Guan, J. Cheng, K. Cen, and W. Hua, "Crfs based de-identification of medical records," *Journal of biomedical informatics*, vol. 58, pp. S39–S46, 2015.

[13] F. Dernoncourt, J. Y. Lee, O. Uzuner, and P. Szolovits, "De-identification of patient notes with recurrent neural networks," *Journal of the American Medical Informatics Association*, vol. 24, no. 3, pp. 596–606, 2017.

[14] X. Yang, T. Lyu, Q. Li, C.-Y. Lee, J. Bian, W. R. Hogan, and Y. Wu, "A study of deep learning methods for de-identification of clinical notes in cross-institute settings," *BMC Medical Informatics and Decision Making*, vol. 19, no. 5, p. 232, 2019.

[15] Z. Liu, B. Tang, X. Wang, and Q. Chen, "De-identification of clinical notes via recurrent neural network and conditional random field," *Journal of biomedical informatics*, vol. 75, pp. S34–S42, 2017.

[16] Y. Kim, P. Heider, and S. Meystre, "Ensemble-based methods to improve de-identification of electronic health record narratives," in *AMIA Annual Symposium Proceedings*, vol. 2018. American Medical Informatics Association, 2018, p. 663.

[17] Y. Shen, H. Yun, Z. C. Lipton, Y. Kronrod, and A. Anandkumar, "Deep active learning for named entity recognition," *arXiv preprint arXiv:1707.05928*, 2017.

[18] A. Bouchachia, "Incremental learning with multi-level adaptation," *Neurocomputing*, vol. 74, no. 11, pp. 1785–1799, 2011.

[19] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, and S. Wermter, "Continual lifelong learning with neural networks: A review," *Neural Networks*, vol. 113, pp. 54 – 71, 2019.

[20] R. E. Schapire and Y. Freund, *Boosting: Foundations and Algorithms*. The MIT Press, 2012.

[21] D. H. Wolpert, "Stacked generalization," *Neural Networks*, vol. 5, no. 2, 1992.

[22] A. Krogh and J. Vedelsby, "Neural network ensembles, cross validation and active learning," in *Proceedings of the 7th International Conference on Neural Information Processing Systems*, ser. NIPS'94. Cambridge, MA, USA: MIT Press, 1994, p. 231–238.

[23] Y. Yu, Z. Zhou, and K. M. Ting, "Cocktail ensemble for regression," in *Seventh IEEE International Conference on Data Mining (ICDM 2007)*, 2007, pp. 721–726.

[24] Z.-H. Zhou, *Ensemble methods: foundations and algorithms*. CRC press, 2012.

[25] U. States, "The health insurance portability and accountability act (hipaa). washington, d.c.: U.s. dept. of labor, employee benefits security administration." 2004.

[26] H. Zhang, "The optimality of naïve bayes," in *In FLAIRS2004 conference*, 2004.

[27] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *Journal of Machine learning research*, vol. 7, no. Jan, pp. 1–30, 2006.

[28] M. Scaiano, G. Middleton, L. Arbuckle, V. Kolhatkar, L. Peyton, M. Dowling, D. S. Gipson, and K. El Emam, "A unified framework for evaluating the risk of re-identification of text de-identification tools," *J. of Biomedical Informatics*, vol. 63, no. C, p. 174–183, Oct. 2016.

[29] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of NAACL-HLT 2019*, 2019.

[30] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems 30*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds. Curran Associates, Inc., 2017, pp. 5998–6008.

Appendix

In this section, we cover proofs for mathematical claims in the paper, details on significance tests and additional figures.

A. Derivation of the weight equation for Brier score

$$
\begin{aligned}
&\sum_{i=1}^{K}\sum_{j=1}^{K}(f^c_{ij} - y_{ij})^2 \\
\frac{\partial}{\partial p} &= \sum_X \sum (w f_1(\mathbf{x}) + (1-w) f_2(\mathbf{x}) - y)^2 \\
&= \sum_X \sum 2(w f_1(\mathbf{x}) + (1-w) f_2(\mathbf{x}) - y)(f_1(\mathbf{x}) - f_2(\mathbf{x})) \\
0 &= 2\sum_X \sum (w(f_1(\mathbf{x}) - f_2(\mathbf{x}))^2 + (f_2(\mathbf{x}) - y))(f_1(\mathbf{x}) - f_2(\mathbf{x})) \\
w &= \frac{\sum_X \sum -f_1(\mathbf{x}) f_2(\mathbf{x}) + f_1(\mathbf{x}) y + f_2(\mathbf{x})^2 - f_2(\mathbf{x}) y}{\sum_X \sum f_1(\mathbf{x})^2 - 2 f_1(\mathbf{x}) f_2(\mathbf{x}) + f_2(\mathbf{x})^2} \\
w &= \frac{\sum_X \sum (y - f_2(\mathbf{x}))(f_1(\mathbf{x}) - f_2(\mathbf{x}))}{\sum_X \sum (f_1(\mathbf{x}) - f_2(\mathbf{x}))^2} \\
\frac{\partial^2}{\partial p^2} &= \frac{\partial}{\partial p} \sum_X \sum 2(w f_1(\mathbf{x}) + (1-w) f_2(\mathbf{x}) - y)(f_1(\mathbf{x}) - f_2(\mathbf{x})) \\
&Using\ the\ product\ rule\ we\ get: \\
&= \sum_X \sum (2 f_1(\mathbf{x}) - 2 f_2(\mathbf{x}))(f_1(\mathbf{x}) - f_2(\mathbf{x})) \\
&= \sum_X \sum 2(f_1(\mathbf{x}) - 2 f_2(\mathbf{x}))^2 \\
&The\ range\ of\ f_1(\mathbf{x})\ and\ f_2(\mathbf{x})\ is\ \mathbf{X}^d\ where\ x \in [0,1]\ so\ as \\
&\frac{\partial^2}{\partial p^2} \geq 0,\ the\ sufficient\ condition\ for\ convexity\ holds.
\end{aligned}
$$

B. Derivation of the weight equation for Log-Loss

$$
\begin{aligned}
&-\sum_{i=1}^{N}\sum_{k=1}^{K} y_{ij} log(f^c_{ij}) \\
\frac{\partial L}{\partial p} &= -\sum_{i=1}^{N}\sum_{k=1}^{K} y_{ij} log(w f_1(\mathbf{x}) + (1-w) f_2(\mathbf{x})) \\
&= -\sum_X \sum \frac{y_{ij}(f_1(\mathbf{x}) - f_2(\mathbf{x}))}{w f_1(\mathbf{x}) + (1-w) f_2(\mathbf{x})} \\
0 &= -\sum_X \sum \frac{y_{ij}(f_1(\mathbf{x}) - f_2(\mathbf{x}))}{w f_1(\mathbf{x}) + (1-w) f_2(\mathbf{x})}
\end{aligned}
$$

Solving at 0 yields, $N-1$ solutions. We can solve for the root through newtons method by providing a good starting point within a suitable search interval.

C. Significance Tests

Version	Dataset	Task	P-Value
HIPAA	NZ	CRF vs Ensemble	2.6297e-05
		INB vs Ensemble	4.1490e-06
	i2b2	CRF vs Ensemble	0.001885
		INB vs Ensemble	4.4287e-05
Direct/Quasi	NZ	CRF vs Ensemble	0.0079
		INB vs Ensemble	8.2980e-06
	i2b2	CRF vs Ensemble	0.000156
		INB vs Ensemble	1.2290e-05

TABLE III: P-values from the Wilcoxon signed-rank test

The null hypothesis is that the base learning algorithm performs better than the ensemble, while the alternative hypothesis is that ensemble performs better. At the 5% confidence level, in all cases, we reject the null and accept the alternative hypothesis.

D. Weight Estimation

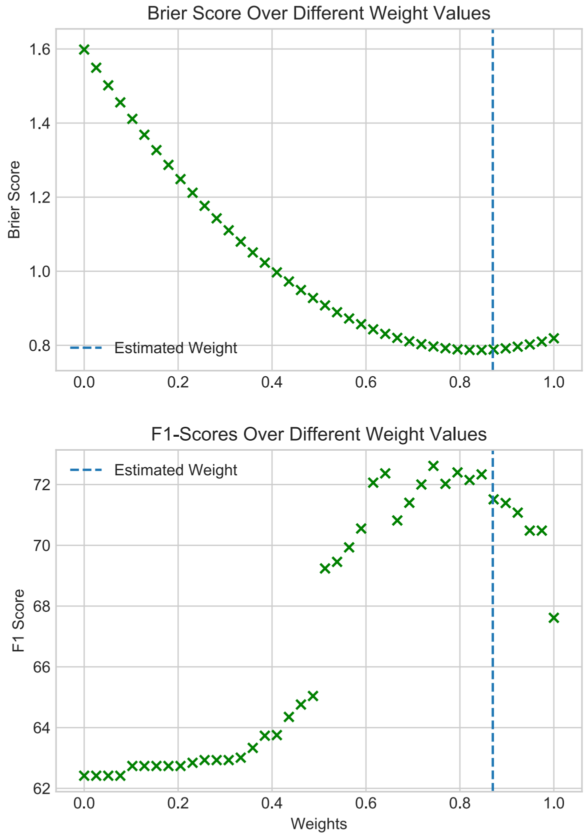

Fig. 7: Brier Score loss (top) and F1 scores (bottom) of the ensemble on the test set given different weight values. The dashed vertical line is the estimated weight on the validation set. The charts are generated from the HIPAA version of the NZ data set as an example. Similar charts can be obtained from other datasets.