APPROXIMATE CONDITIONAL COVERAGE VIA NEURAL MODEL APPROXIMATIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

We propose a new approach for constructing prediction sets for Transformer networks via the strong signals for prediction reliability from KNN-based approximations. This enables a data-driven partitioning of the high-dimensional feature space and a new Inductive Venn Predictor for calibration, the Venn-ADMIT Predictor. Our approach more closely obtains *approximate conditional coverage* than recent work proposing adaptive and localized conformal score functions for deep networks. We analyze coverage on several representative natural language processing classification tasks, including class-imbalanced and distribution-shifted settings.

1 INTRODUCTION

Uncertainty quantification with deep neural networks for classification is challenging. The otherwise strong blackbox *point* predictions are typically not well-calibrated. Constructing distribution-free prediction sets with split-conformal inference (Vovk et al., 2005; Papadopoulos et al., 2002) is an assumption-light frequentist approach suitable when sample sizes are sufficiently large; however, the quantities produced are typically not ideal for instance-level decision making and require some care in properly interpreting, as discussed below. In this work, we propose a score function that provides adaptive coverage in the standard split-conformal framework, and we then use that approach to construct a taxonomy for a Venn Predictor (Vovk et al., 2003). We use the latter to construct regions (prediction sets) as with conformal approaches, but the resulting quantities are more suitable for instance-level decision making. These two distinct but complementary quantities, conformal prediction sets and set-conditioned calibrated probabilities, cover a wide-range of use-cases for quantifying classification uncertainty with deep networks. To motivate our approach, we will first review the quantities produced by split-conformal inference.

In a typical natural language processing (NLP) binary or multi-class classification task, we have access to a computationally expensive blackbox neural model, F; a training dataset, $\mathcal{D}_{\text{tr}} = \{(X_i, Y_i)\}_{i=1}^{I}$ of $|\mathcal{D}_{\text{tr}}| = I$ instances paired with their corresponding ground-truth discrete labels, $Y_i \in \mathcal{Y} = \{1, \ldots, C\}$; and a held-out labeled calibration dataset, $\mathcal{D}_{\text{ca}} = \{(X_j, Y_j)\}_{j=I+1}^{N=I+J}$ of $|\mathcal{D}_{\text{ca}}| = J$ instances. We are then given a new test instance, X_{N+1}, from an unlabeled test set, $\mathcal{D}_{\text{te}}$. We then seek to construct a prediction set, produced by some set-valued function $\hat{\mathcal{C}}(X_{N+1}) \in 2^C$, containing the true unseen label with a specified coverage level $1 - \alpha \in (0, 1)$ on average. How we define that average requires careful consideration of the theoretical assumptions we are willing to make and the practical limitations of existing approaches and our available data and models.

Provided the points of $\mathcal{D}_{\text{tr}}$, $\mathcal{D}_{\text{ca}}$, and $\mathcal{D}_{\text{te}}$ are all drawn exchangeably from the same unknown distribution P_{XY}, standard split-conformal methods provide a finite-sample *marginal* guarantee:

$$\mathbb{P}\left\{Y_{N+1} \in \hat{\mathcal{C}}(X_{N+1})\right\} \geq 1 - \alpha \tag{1}$$

That is to say, the true label is contained within $\hat{\mathcal{C}}$ with a proportion of at least $1 - \alpha$ on average over random draws from P_{XY}. The approach has a parsimonious, intuitive appeal; it amounts to setting thresholds on the quantiles of the distribution of network outputs transformed by a *non-conformity function*, $s(\cdot)$, which encodes similarity of the test point to the calibration set. For example, given the softmax output of a neural network, $\hat{\pi} \in \mathbb{R}^C$, with $\hat{\pi}^y$ as the output of the true class, we set $s(x_j) = 1 - \hat{\pi}_j^y$ and $\hat{l}^\alpha$ as the estimated $\lceil (J+1)(1-\alpha) \rceil / J$ adjusted quantile of scores over the

calibration set, $s(x_{j=I+1}), \ldots, s(x_{I+J})$. We then construct a prediction set for an unseen test point:

$$\hat{C}(x_{N+1}) = \{c \in \mathcal{Y} : \hat{\pi}^c(x_t) \geq \hat{\tau}^\alpha\}, \text{ where } \hat{\tau}^\alpha = 1 - \hat{l}^\alpha \tag{2}$$

In this way, we can avoid the challenge of further characterizing the error distribution of F. Unfortunately, marginal coverage may be too weak for many applications. Systematic under-coverage for certain subpopulations may be unacceptable, and we may not want over-coverage of "easy" to predict classes at the expense of under-coverage of "difficult" to predict classes. Under-coverage for singleton sets, $|\hat{\mathcal{C}}| = 1$, complicates their use for instance-level decision making and is a potential source of mis-interpretation by end-users that we may seek to preemptively avoid in higher-risk settings. As such, we might seek distribution-free *conditional* coverage:

$$\mathbb{P}\left\{Y_{N+1} \in \hat{\mathcal{C}}(X_{N+1}) \,|\, X_{N+1} = x\right\} \geq 1 - \alpha \tag{3}$$

Unfortunately, this is known to result in uninformatively large sets in the distribution-free setting with a finite sample (Vovk, 2012; Lei & Wasserman, 2014). The best we might reasonably hope to achieve, aside from label conditioning, is to condition on the set of points in the partition, $\mathcal{B}$, of some non-trivial probability ξ, around an instance, for which we follow past work in calling *approximate conditional* coverage (Foygel Barber et al., 2020):

$$\mathbb{P}\left\{Y_{N+1} \in \hat{\mathcal{C}}(X_{N+1}) \,|\, X_{N+1} \in \mathcal{B}(x)\right\} \geq 1 - \alpha, \text{ with } P_X(\mathcal{B}(x)) \geq \xi \tag{4}$$

However, given the impossibility of 3, how we define and condition on $\mathcal{B}$ must be selective; the challenge is to partition the data in a manner applicable to a wide-variety of scenarios. Further, the coverage guarantees are a moot point if the distributions are not in fact exchangeable. Domain shifts in our test data could lead to unexpected under-coverage, so we seek reasonable robustness against—and at least, not catastrophic failure in the presence of—covariate and label shifts.

Given these challenges, recent works have proposed new conformal score functions for multi-class classification tasks for deep networks, seeking adaptivity that more closely resembles conditional coverage (Romano et al., 2020; Angelopoulos et al., 2021, inter alia). Re-weighting has been proposed for addressing covariate shift, provided the likelihood ratio can be estimated (Tibshirani et al., 2019). A recent generalization, localized conformal, demonstrated for a one-dimensional regression response, seeks to move beyond marginal coverage by re-weighting the empirical CDF near a test point (Guan, 2022).

We take a distinct approach from these recent works, based on the old idea of "mondrian" conformal prediction, including conditioning on the labels (Vovk et al., 2005), for partitioning the data into similar regions over which unique conformal thresholds are constructed. We further construct a Venn taxonomy based on these regions, treating our adaptive "ADMIT" prediction sets as outputs of the underlying model. The result is a calibrated probability that can be used independently of—or in conjunction with, as an additional empirical safeguard—the conformal sets, and is more suitable for instance-level decision making. We achieve this with modern Transformer networks (Vaswani et al., 2017) by constructing a weighted KNN approximation of the underlying model (Schmaltz, 2021). The properties of KNNs have been well-studied (Devroye et al., 1996); by leveraging the deep network as a metric learner, we can then partition the high-dimensional feature space with a KNN, yielding useful properties for example-based interpretability, updatability (as by modifying the KNN's support set), and as shown here, uncertainty quantification.

In summary, with minimal assumptions and free parameters by leveraging dense representation matching, we construct prediction sets over Transformer networks, obtaining the quantities of interest needed in a wider-range of settings than existing approaches:

1. ADMIT prediction sets obtain a closer notion of approximate conditional coverage than recent conformal score functions APS (Romano et al., 2020), RAPS (Angelopoulos et al., 2021), and a strong localized approach (Guan, 2022) we adapt for classification.
2. The set-conditioned probabilities from the VENN-ADMIT Predictor are well-calibrated at high confidence levels, even with models with class-conditional accuracy far less than the specified coverage level. This is reflected in sets that obtain coverage stratified by size and class, providing a quantity of interest applicable to a wider range of decision support settings than typical conformal prediction sets.
3. Our data partitioning consistently detects the most reliable subset of the data, even with weak models, providing an additional safeguard in higher-risk settings.

Table 1: Overview of experiments.

Label	Task	$\lvert\mathcal{Y}\rvert$	$\lvert\mathcal{D}_{\text{valid}}\rvert$	$\lvert\mathcal{D}_{\text{te}}\rvert$	$\boldsymbol{r}$	Base network	Acc.	Characteristics
PROTEIN	**SSL**	3	560k	{30k,7k}	$\mathbb{R}^{1000}$	$\sim$ BERT$_{\text{BASE}}$	Mid	In-domain (2 test sets)
GRAMMAR	**SSL**	2	35k	93k	$\mathbb{R}^{1000}$	BERT$_{\text{LARGE}}$	Low	Domain-shifted+imbalanced
SENTIMENT	**DC**	2	16k	488	$\mathbb{R}^{2000}$	BERT$_{\text{LARGE}}$	High	In-domain (acc. $> 1-\alpha$)
SENTIMENTOOD	**DC**	2	16k	5k	$\mathbb{R}^{2000}$	BERT$_{\text{LARGE}}$	Mid-Low	Domain-shifted/OOD

2 GOAL: APPROXIMATE CONDITIONAL COVERAGE

Our base split-conformal approach, ADMIT, obtains the following notion of coverage:

$$\mathbb{P}\left\{Y_{N+1} \in \hat{\mathcal{C}}(X_{N+1}) \,|\, X_{N+1} \in \mathcal{B}(x), Y_{N+1} = y\right\} \geq 1-\alpha, \text{ with } P_X(\mathcal{B}(x)) \geq \xi \tag{5}$$

In words, we construct prediction sets that obtain coverage stratified by data partition, $\mathcal{B}(x)$, and the true label, y. The added conditioning on the label (c.f., Eq. 4) is obtained by applying split-conformal prediction separately for each label, "label-conditional" conformal prediction (Vovk et al., 2005; Vovk, 2012; Sadinle et al., 2018), which provides built-in robustness to label proportion shifts (c.f., Podkopaev & Ramdas, 2021). Robust data partitioning is obtained by dividing the data into groups determined by the depth of true positive matches into the support set of a KNN that approximates the original model.

We will also seek the following stronger notion of approximate conditional coverage, in which we condition on membership of $\hat{\mathcal{C}}(X_{N+1})$:

$$\mathbb{P}\left\{Y_{N+1} \in \hat{\mathcal{C}}(X_{N+1}) \,|\, X_{N+1} \in \mathcal{B}(x), Y_{N+1} = y, \hat{\mathcal{C}}_{N+1} = \mathcal{A}\right\} \geq 1-\alpha,\ P_X(\mathcal{B}(x)) \geq \xi,\ \mathcal{A} \in 2^C \tag{6}$$

We consider all $\mathcal{A} \in 2^C$ given the large $|\mathcal{D}_{\text{ca}}|$ and small $|\mathcal{Y}|$ considered here, but in general, we will primarily want to ensure coverage is maintained for low cardinality sets—and in particular, singleton sets—for each class. We will do so by relying on the multi-probability calibration guarantees of an Inductive Venn Predictor with a taxonomy determined by $\hat{\mathcal{C}}(X)$, the predictions, and the data partitions. A KNN localizer encourages more diffuse estimates when a test point is unexpectedly far from its assigned category. The resulting VENN-ADMIT prediction sets retain at least the standard split-conformal distribution-free, finite-sample guarantees for Eq. 5. In practice, the set-conditioned probabilities from the VENN-ADMIT Predictor are sufficiently well-calibrated that a naive probability mass correction obtains the close approximate conditional coverage of Eq. 6.

3 TASKS: CLASSIFICATION WITH TRANSFORMERS FOR NLP

We first introduce general notation for sequence labeling and document classification tasks. Each instance consists of a document, $\boldsymbol{x} = x_1, \ldots, x_t, \ldots, x_T$, of T tokens: Here, either words or amino acids. In the case of **supervised sequence labeling** (**SSL**), we seek to predict $\hat{\boldsymbol{y}} = \hat{y}_1, \ldots, \hat{y}_t, \ldots, \hat{y}_T$, the token-level labels for each token in the document, and we have the ground-truth token labels, y_t, for training. For **document classification** (**DC**), we seek to predict the document-level label $\hat{y}$, and we have access to y at training.

For each task, our base model is a Transformer network. After training and/or fine-tuning, we fine-tune a kernel-width 1 CNN (MEMORY LAYER) over the output representations of the Transformer, producing predictions and representative dense vectors at a resolution (e.g., word-level or document-level) suitable for each task. Following past work, we will refer to these representations as "exemplar vectors" primarily to contrast with "prototype", which is sometimes taken to refer to class-centroids, whereas the "exemplars" are unique to each instance. We will subsequently use $f(x_t) \in \mathbb{R}^C$ for the prediction logits produced by the MEMORY LAYER corresponding to the token at index t; π^c as the corresponding softmax normalized output for class c; and $\boldsymbol{r}_t$ as the associated exemplar vector. For **SSL** there are T such logits and vectors. For **DC**, t corresponds to a single representation of the document, with $f(x_t)$ formed by a combination of local and global predictions (as described further in the Appendix). In the present work, we will primarily only be concerned with Transformers at the level of abstraction of the exemplar vectors, $\boldsymbol{r}_t$; we refer the reader to the original works describing

Transformers (Vaswani et al., 2017) and the particular choice for the MEMORY LAYER (Schmaltz, 2021) for additional details.

When helpful to avoid ambiguity, we use the subscripts tr, ca, or te to indicate association with the training, calibration, or test set, respectively. We will also assume the existence of an additional split of the data, $\mathcal{D}_{\mathrm{knn}}$, which is disjoint from $\mathcal{D}_{\mathrm{tr}}$, $\mathcal{D}_{\mathrm{ca}}$, and $\mathcal{D}_{\mathrm{te}}$, used for setting the parameters of the KNNs, as described next.

4 METHODS

Non-parametric approximations of Transformers encode strong signals for prediction reliability, including over distribution-shifts, and are at least as effective as the model being approximated (Schmaltz, 2021). Predictions become less reliable at L^2 distances farther from the training set and with increased label and prediction mismatches among the nearest matches. In Section 4.1, we leverage this behavior to partition the feature space in a data-driven manner and to construct adaptive prediction sets (Section 4.2). In Section 4.3, we introduce our Venn Predictor, including an additional KNN approximation that serves as a localizer, relating a test instance to the distribution of the conformal calibration set. Finally, in Section 4.4, we combine these ideas into a straightforward algorithm for constructing prediction sets with size-stratified coverage.

4.1 KNN APPROXIMATION OF A TRANSFORMER NETWORK

In order to partition the feature space, we first approximate the Transformer as a weighted combination of predictions and labels over $\mathcal{D}_{\mathrm{tr}}$ (Section 4.1.1). This approximation, $f(x)_{\mathrm{tr}}^{\mathrm{KNN}}$, then becomes the model we use in practice, rather than the logits from the Transformer itself. We use this approximation to partition the data via a feature that separates more reliable points from less reliable points (Section 4.1.2) and via a distance-to-training band (Section 4.1.3).

4.1.1 RECASTING A TRANSFORMER PREDICTION AS A WEIGHTING OVER THE TRAINING SET

We adapt the distance-weighted KNN approximation of Schmaltz (2021) for the multi-class setting. As in the original work, the KNN is trained to minimize prediction mis-matches against the output of the MEMORY LAYER (*not* the ground-truth labels). Training is performed on a 50/50 split of $\mathcal{D}_{\mathrm{knn}}$:

$$f^c(x_t) \approx f^c(x_t)_{\mathrm{tr}}^{\mathrm{KNN}} = \beta^c + \sum_{k \in \underset{i \in \{1,\ldots,|\mathcal{D}_{\mathrm{tr}}|\}}{\arg\mathrm{K\,min}} ||\boldsymbol{r}_t - \boldsymbol{r}_i||_2} w_k \cdot (\tanh(f^c(x_k)) + \gamma^c \cdot \tilde{y}^c)\,, \tag{7}$$

$$\text{where } w_k = \frac{\exp\left(-||\boldsymbol{r}_t - \boldsymbol{r}_k||_2/\eta\right)}{\sum_{k' \in \underset{i \in \{1,\ldots,|\mathcal{D}_{\mathrm{tr}}|\}}{\arg\mathrm{K\,min}} ||\boldsymbol{r}_t - \boldsymbol{r}_i||_2} \exp\left(-||\boldsymbol{r}_t - \boldsymbol{r}_{k'}||_2/\eta\right)} \tag{8}$$

$\tilde{y}^c$ is the ground-truth label (y for **DC**, y_t for **SSL**) for class c transformed to be in $\{-1, 1\}$. K is small in practice; $K = 25$ in all experiments here, and in general can be chosen usingt $\mathcal{D}_{\mathrm{knn}}$. This approximation has $2 \cdot C + 1$ learnable parameters, corresponding to β^c and γ^c for each class, and the temperature parameter η. The choice of this particular functional form is discussed in Schmaltz (2021) and is designed to be as simple as possible while obtaining a sufficiently close approximation to the deep network. We indicate the softmax normalized output for each class with $\pi^c(x_t)_{\mathrm{tr}}^{\mathrm{KNN}}$. This model is used to produce approximations over all calibration and test instances.

4.1.2 DATA-DRIVEN FEATURE-SPACE PARTITIONING: TRUE POSITIVE MATCHING CONSTRAINT

For each calibration and test point, we define the feature $q_t \in [0, K]$ as the count of *consecutive* sign matches of the prediction of the KNN, $f(x_t)_{\mathrm{tr}}^{\mathrm{KNN}}$, with the true label and MEMORY LAYER prediction of the up to K nearest matches from the *training* set, $\mathcal{D}_{\mathrm{tr}}$:

$$q_t(K) = \sum_{k \in \underset{i \in \{1,\ldots,|\mathcal{D}_{\mathrm{tr}}|\}}{\arg\mathrm{K\,min}} ||\boldsymbol{r}_t - \boldsymbol{r}_i||_2} \left[\hat{y}_t^{\mathrm{KNN}} = \hat{y}_k^{\mathrm{tr}}\right] \wedge \left[\hat{y}_k^{\mathrm{tr}} = y_k^{\mathrm{tr}}\right] \wedge \left[q_t(k-1) = k-1\right], \tag{9}$$

with $q_t(0) := 0$. We further also use the L^2 distance to the nearest training set match as a basis for subsetting the distribution into distance bands, as discussed in the next section:

$$d_t = \min ||\boldsymbol{r}_t - \boldsymbol{r}_i||_2, i \in \{1, \ldots, |\mathcal{D}_{\text{tr}}|\} \tag{10}$$

4.1.3 Data-driven feature-space partitioning: Distance-to-training band

We define the partition, $\mathcal{B}$, around each $x_t \in \mathcal{D}_{\text{te}}$, constrained to q, as the L^2-distance-to-training band with a radius of $\omega = \delta \cdot \hat{\text{s}}$, with $\delta \in \mathbb{R}^+$ as a user-specified parameter and $\hat{\text{s}}$ as the estimated standard deviation of constrained true positive calibration set distances, $\hat{\text{s}} = \text{std}([\text{d}_\text{j} : \text{j} \in \{\text{I} + 1, \ldots, \text{I} + |\mathcal{D}_{\text{ca}}|\}, q_\text{j} > 0, \hat{y}_j^{\text{KNN}} = \text{y}_\text{j}])$:

$$\mathcal{B}(x_t, \omega, q_t, d_t; \mathcal{D}_{\text{ca}}) = \{x_j : x_j \in \mathcal{D}_{\text{ca}}, d_j \in [d_t - \omega, d_t + \omega], q_t = q_j\} \tag{11}$$

4.2 Prediction sets with approximate conditional coverage: ADMIT

Given $f(x)_{\text{tr}}^{\text{KNN}}$, band radius ω, and a user-specified coverage level $1 - \alpha \in (0, 1)$, we construct an ADMIT prediction set for a point by setting class-specific thresholds on the quantile distributions of the calibration points restricted to $\mathcal{B}$ (Eq. 11), with $\hat{\mathcal{C}}(x_t)$ determined as in Eq. 2 but with the class- and partition-specific thresholds. Pseudo-code appears in Appendix C. Remarkably, this simple procedure is sufficient to obtain relatively adaptive coverage compared to alternatives. However, as with split-conformal approaches in general, singleton set coverage by class can be quite poor with coverage obtained over stratifications with low point accuracy by over-covering the larger set cardinalities. We address this via a Venn Predictor, described next.

4.3 Inductive Venn-ADMIT Predictor (IVAP)

A Venn Predictor[1] is a frequentist calibration method, providing validity guarantees for calibration assuming exchangeability. Venn Prediction is a general framework; the analogue to the score function of conformal inference is the modeling choice of the partitioning ("taxonomy") of the data into categories. Taxonomies include variations on nearest neighbors (Johansson et al., 2018) and isotonic regression, the Venn-ABERS Predictor (Vovk & Petej, 2014; Vovk et al., 2015). We take as our base formulation the Inductive variant of Lambrou et al. (2015), in which the training set is subsumed as part of the underlying taxonomy.

The taxonomy of the Venn-ADMIT Predictor is defined based on the prediction of the KNN, $f(x_t)_{\text{tr}}^{\text{KNN}}$, and the ADMIT prediction sets themselves, which are treated as predictions of the underlying model. Specifically, points in the same category, $\mathcal{T}$, as x_t have matching KNN predictions, $\hat{y}^{\text{KNN}}$; belong to the same data partition $\mathcal{B}$; and have corresponding prediction sets $\hat{\mathcal{C}}$ with the same labels selected. Prediction sets for the held-out calibration set are calculated in a leave-one-out manner.[2]

An advantage of Venn Predictors is that they produce multi-calibrated output, with differences across augmented distributions a function of the sample size of the category.[3] We take as the probability for each class the mean over the augmented distributions; categories with smaller sample sizes will have more diffuse distributions, ceteris paribus. We exploit this property further by re-weighting the point weight of the augmented test point as a heuristic for robustness to distribution shifts. To determine the weights, we use a KNN localizer, recasting the test approximation output as a weighted linear combination over the calibration set approximations:

$$f^c(x_t)_{\text{tr}}^{\text{KNN}} \approx f^c(x_t)_{\text{ca}}^{\widehat{\text{KNN}}} = \sum_{k \in \underset{j \in \{I+1, \ldots, I+|\mathcal{D}_{\text{ca}}|\}}{\arg\text{K}\min} ||\boldsymbol{r}_t - \boldsymbol{r}_j||_2} \psi_k \cdot f^c(x_k)_{\text{tr}}^{\text{KNN}}, \text{where } K = |\mathcal{D}_{\text{ca}}| \tag{12}$$

[1] Chapter 6 of Vovk et al. (2005) provides an introduction.

[2] This leave-one-out approach is simply standard inductive split-conformal inference applied to each held-out point in turn, and is distinct from hold-out methods that seek to improve the statistical efficiency of conformal prediction, as with cross-conformal predictors (Vovk, 2015) and the jacknife+ (Barber et al., 2021).

[3] The distribution with validity is not in general identifiable, but with sufficiently large sample sizes, the multiple distributions produced by a Venn Predictor become nearly indistinguishable.

The single parameter, the temperature parameter of ψ_k, which is calculated in an analogous manner as w_k in Equation 8, is trained via gradient descent against $\mathcal{D}_{\text{te}}$ to minimize *prediction* discrepancies between $f(x_t)^{\text{KNN}}_{\text{tr}}$ and $f(x_t)^{\widehat{\text{KNN}}}_{\text{ca}}$. As with $f(x)^{\text{KNN}}_{\text{tr}}$, training is performed using $\mathcal{D}_{\text{knn}}$. For a given test point, we calculate $f(x_t)^{\widehat{\text{KNN}}}_{\text{ca}}$ (Eq. 12), and then determine the augmented distribution for the Venn Predictor by adding the test point with a weight according to the multiplicative inverse of the sum of the subset of the weights for the calibration points appearing in the category, $\mathcal{T}$: $(\sum \psi_j)^{-1},\ x_j \in \mathcal{T}$. When this weight is 1, we have the standard Venn Predictor with validity guarantees; when this weight is greater than 1, it is a sign of a mismatch (due to non-exchangeability, or an otherwise aberrant category assignment) and the distribution is made more diffuse.

4.4 VENN-ADMIT PREDICTION SETS

For a given test point, x_t, we first construct $\hat{\mathcal{C}}(x_t)$ with the ADMIT procedure of Section 4.2, followed by calibration via the VENN-ADMIT Predictor (Section 4.3). Pseudo-code appears in Appendix C. We then perform a greedy probability mass correction: If the sum of the IVAP probabilities of the points in $\hat{\mathcal{C}}(x_t)$ fall below $1-\alpha$, we progressively add the label with the next highest probability until the total mass reaches at least $1-\alpha$.

5 EXPERIMENTS

We evaluate construction of the ADMIT prediction sets, and those with the VENN-ADMIT mass correction, on a wide-range of representative NLP tasks, including challenging domain-shifted and class-imbalanced settings, and in settings in which the point prediction accuracies are quite high (marginally $> 1-\alpha$) and in which they are relatively low. We follow past work in setting $\alpha = 0.1$ in our main experiments. Unless specified otherwise, $\delta = 1$. Additional variations are considered in the Appendix. We summarize and label our benchmark **tasks**, the underlying parametric networks, and data in Table 1[4]. A disjoint set of size 144k, the CB513 set from PROTEIN, was used for initial methods development. In the Table, $\mathcal{D}_{\text{valid}}$ is the original held-out validation set associated with each task. For the ADMIT approaches, a random 10% sample of $\mathcal{D}_{\text{valid}}$ serves as the disjoint $\mathcal{D}_{\text{knn}}$ set for training the KNNs, with the rest of the data used for $\mathcal{D}_{\text{ca}}$. The baseline and comparison methods are given the full $\mathcal{D}_{\text{valid}}$ as $\mathcal{D}_{\text{ca}}$. The Appendix provides implementation details on constructing the exemplar vectors, $\boldsymbol{r}$, for each of the tasks from the MEMORY LAYER.

5.1 COMPARISON MODELS

As a distribution-free **baseline** of comparison we consider the size- and adaptiveness-optimized RAPS algorithm of Angelopoulos et al. (2021), $\text{RAPS}_{\text{SIZE}}$ and $\text{RAPS}_{\text{ADAPT}}$, which combine regularization and post-hoc Platt-scaling calibration (Platt, 1999; Guo et al., 2017), on the output of the MEMORY LAYER. Using stratification of coverage by cardinality as a metric, $\text{RAPS}_{\text{ADAPT}}$, in particular, was reported to more closely approximate conditional coverage than the alternative APS (Romano et al., 2020), with smaller sets. $\text{CONF}_{\text{BASE}}$ is a split-conformal point of reference for simply using the output of $f(x)^{\text{KNN}}_{\text{tr}}$ without further conditioning, nor post-hoc calibration. $\text{LOCAL}_{\text{CONF}}$ is a localized conformal baseline using the KNN localizer $f(x)^{\widehat{\text{KNN}}}_{\text{ca}}$. Across methods, the point prediction is included in the set, which ensures conservative (but not necessarily exact/upper-bounded) coverage by eliminating null sets. We evaluate coverage, $\overline{y \in \mathcal{C}}$, and cardinality, $\overline{|\mathcal{C}|}$, across training distance, class, and cardinality stratifications.

6 RESULTS

Across tasks, the KNNs consistently achieve similar point accuracies as the base networks (Table 2). Table 3 highlights our core motivations for leveraging the signals from the KNNs: There are stark differences across instances as q increases and as the distance to training increases (shown here for

[4]Devlin et al. (2019); Rao et al. (2019); El-Gebali et al. (2019); Berman et al. (2000); Moult et al. (2018); Klausen et al. (2019); Yannakoudakis et al. (2011); Rei & Yannakoudakis (2016); Chelba et al. (2014); Kaushik et al. (2020); Mikolov et al. (2013); Wolf et al. (2020); Rosenthal et al. (2017); Maas et al. (2011)

Table 2: Model approximation vs. MEMORY LAYER accuracy/$F_{0.5}$.

Model/Approx.	PROTEIN (ACC.)			GRAMMAR ($F_{0.5}$)		SENTIMENT (ACC.)		SENTIMENTOOD (ACC.)	
	$\mathcal{D}_{\text{ca}}$	TS115	CASP12	$\mathcal{D}_{\text{ca}}$	$\mathcal{D}_{\text{te}}$	$\mathcal{D}_{\text{ca}}$	$\mathcal{D}_{\text{te}}$	$\mathcal{D}_{\text{ca}}$	$\mathcal{D}_{\text{te}}$
MEMORY LAYER	0.75	0.77	0.70	0.59	0.40	0.92	0.93	0.92	0.78
$f(x)^{\text{KNN}}_{\text{tr}}$	0.76	0.77	0.71	0.58	0.43	0.92	0.93	0.92	0.79
$f(x)^{\widehat{\text{KNN}}}_{\text{ca}}$	-	0.77	0.70	-	0.42	-	0.93	-	0.78

Table 3: The empirical behavior of the calibration points differs significantly with $q = 0$ vs. $q = K$, and as the distance to training (d_t) varies, in terms of $f(x)^{\text{KNN}}_{\text{tr}}$ point accuracy (ACC.), and the distribution of over-confidence and under-confidence (reflected in $\hat{\tau}^{0.1}$). (Validation set of PROTEIN.)

	PROTEIN: Class Label (Amino-Acid/Token-Level Sequence Labeling)											
	$y = \mathbf{H}$ELIX			$y = \mathbf{S}$TRAND			$y = \mathbf{O}$THER			$y \in \{\mathbf{H}, \mathbf{S}, \mathbf{O}\}$		
Subset	$\hat{\tau}^{0.1}_c$	ACC.	$\frac{n}{N}$	$\hat{\tau}^{0.1}_c$	ACC.	$\frac{n}{N}$	$\hat{\tau}^{0.1}_c$	ACC.	$\frac{n}{N}$	$\hat{\tau}^{0.1}$	ACC.	$\frac{n}{N}$
$q = 0$	0.07	0.59	0.07	0.07	0.56	0.05	0.18	0.56	0.10	0.11	0.57	0.22
$q = K$	0.96	0.98	0.12	0.95	0.94	0.04	0.92	0.92	0.08	0.94	0.96	0.24
$q \in [0, K]$	0.12	0.81	0.37	0.06	0.70	0.21	0.13	0.74	0.42	0.11	0.76	1.
$d_t <$ median												
$q = 0$	0.09	0.64	0.02	0.07	0.65	0.01	0.16	0.55	0.03	0.12	0.60	0.07
$q = K$	0.96	0.98	0.07	0.95	0.96	0.03	0.93	0.94	0.06	0.94	0.96	0.15
$q \in [0, K]$	0.27	0.87	0.16	0.09	0.81	0.08	0.14	0.79	0.18	0.16	0.82	0.43
$d_t \geq$ median												
$q = 0$	0.07	0.57	0.05	0.07	0.53	0.04	0.19	0.57	0.07	0.11	0.56	0.16
$q = K$	0.96	0.98	0.05	0.04	0.90	0.01	0.03	0.87	0.02	0.93	0.94	0.09
$q \in [0, K]$	0.09	0.76	0.21	0.05	0.62	0.12	0.13	0.70	0.24	0.09	0.70	0.57

PROTEIN, but observed across tasks). In order to obtain coverage on datasets with proportionally more points with $q < K$, and/or far from training, the quantile thresholds must correspondingly be different across these subsets.

Only the VENN-ADMIT sets consistently obtain acceptable singleton set coverage across datasets. Table 4 and the right side of Table 5 demonstrate that generally strong approaches, such as LOCAL$_{\text{CONF}}$, and even the base ADMIT sets, can have quite poor singleton set coverage. In this way, the base split-conformal approaches are not ideal for instance-level decision making.

Both ADMIT and VENN-ADMIT sets obtain coverage by training distance and also by class, across in-distribution tasks and challenging distribution-shifted, low-accuracy settings. The price to pay for this robustness is generally larger sets. However, given that only the VENN-ADMIT sets obtain reliable singleton set coverage, that is likely a necessary cost for many applications.

Essentially all singleton sets occur in the $q = K$ partition. Prospectively, higher-risk settings can refrain from admitting points outside the $q = K$ partition. On the high-accuracy SENTIMENT task, 92% of points are in the $q = K$ partition; in contrast, it is 16% for CASP12, 25% for TS115, 35% for SENTIMENTOOD, and 31% for GRAMMAR.

RAPS and APS behave as advertised. As expected, marginal coverage is consistently obtained across all split-conformal approaches for in-distribution data. LOCAL$_{\text{CONF}}$ exhibits similar coverage and cardinality characteristics as RAPS$_{\text{ADAPT}}$. The benefit of the additional adaptivity of ADMIT is clear even on the in-distribution PROTEIN task, with unexpected under-coverage on the STRAND class with the alternatives (Fig. 1 and 2). On the GRAMMAR task, in which the minority class occurs with a proportion less than α, the stronger per-class mondrian validity guarantees of ADMIT are required to obtain acceptable coverage. The under-coverage of the baseline approaches could come as a surprise to end-users, further exacerbated by poor singleton set coverage. Across datasets, the baseline CONF$_{\text{BASE}}$ exhibits inferior adaptivity; the choice of score function is significant with conformal prediction.

Test-point weighting for the VENN-ADMIT Predictor only makes a substantive difference on the challenging GRAMMAR task; however, importantly it does not harm coverage or calibration on the remaining sets. When not computationally feasible to use the additional KNN localizer, similar singleton set coverage can be obtained with an unweighted VENN-ADMIT Predictor (i.e., always setting the augmented test point weight to 1) and only admitting points with $q = K$. The disadvantage is that all points in other data partitions would then receive full cardinality; however, that trade-

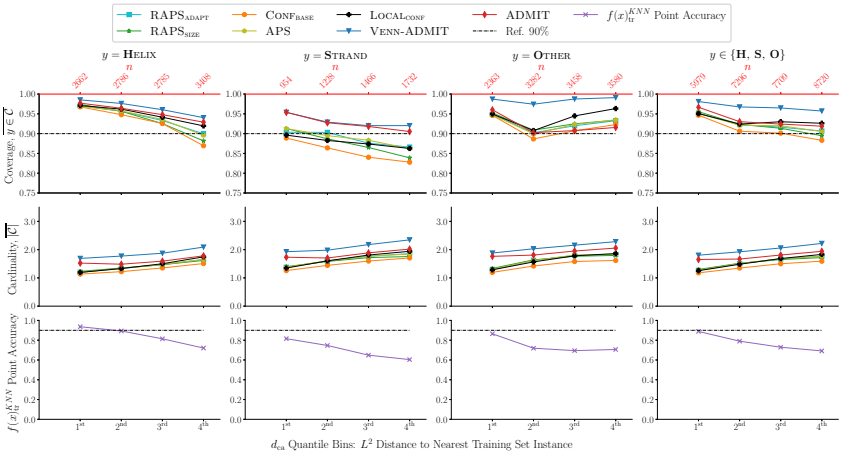

Figure 1: Coverage, cardinality, and point accuracy for the TS115 test set from the PROTEIN task.

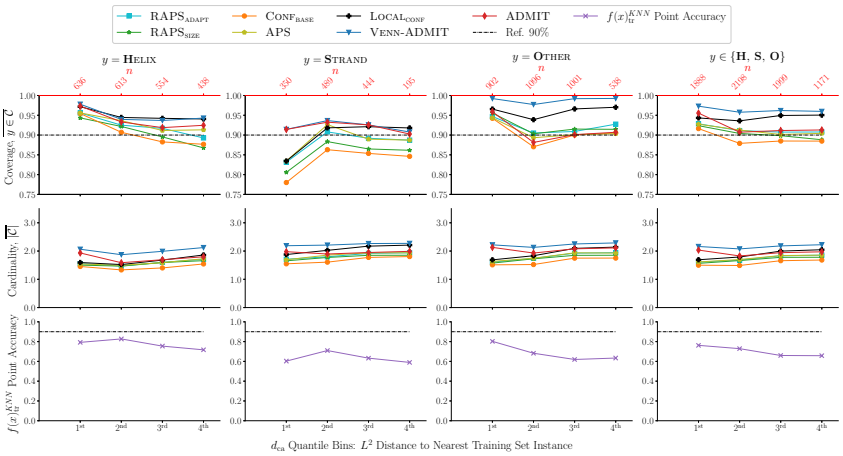

Figure 2: Coverage, cardinality, and point accuracy for the CASP12 test set from the PROTEIN task.

off is likely suitable for applications with high-accuracy Transformers in which the non-singleton sets are treated as rejected predictions sent to humans for further adjudication. The high-accuracy SENTIMENT task is representative of such a scenario, with most in-distribution points in the $q = K$ partition, and with coverage then maintained over distribution-shifts. This is demonstrated with SENTIMENTOOD, which shares the same training and calibration set as the SENTIMENT task, while $\mathcal{D}_{\text{te}}$ contains Twitter data rather than the in-distribution movie reviews of the SENTIMENT task.

7 CONCLUSION

Transformer networks can be approximated as a linear combination over instances with known labels via dense representation matching. We use such KNN approximations to partition the feature space for data-driven mondrian conformal prediction and to construct a taxonomy for an Inductive Venn Predictor. Our resulting split-conformal prediction sets, and the associated set-conditioned calibrated probabilities, are straightforward to construct and obtain adaptive coverage in diverse settings, from in-distribution, high-accuracy tasks to distribution-shifted, low-accuracy tasks.

Table 4: Coverage by cardinality on PROTEIN test sets.

		Class Label (Amino-Acid/Token-Level Sequence Labeling)															
		$y = \textbf{H}$ELIX				$y = \textbf{S}$TRAND				$y = \textbf{O}$THER				$y \in \{\textbf{H}, \textbf{S}, \textbf{O}\}$			
		$\lvert\mathcal{C}\rvert = 1$		$\lvert\mathcal{C}\rvert = 2$		$\lvert\mathcal{C}\rvert = 1$		$\lvert\mathcal{C}\rvert = 2$		$\lvert\mathcal{C}\rvert = 1$		$\lvert\mathcal{C}\rvert = 2$		$\lvert\mathcal{C}\rvert = 1$		$\lvert\mathcal{C}\rvert = 2$	
Set	Method	$\overline{y \in \mathcal{C}}$	$\frac{n}{N}$	$\overline{y \in \mathcal{C}}$	$\frac{n}{N}$	$\overline{y \in \mathcal{C}}$	$\frac{n}{N}$	$\overline{y \in \mathcal{C}}$	$\frac{n}{N}$	$\overline{y \in \mathcal{C}}$	$\frac{n}{N}$	$\overline{y \in \mathcal{C}}$	$\frac{n}{N}$	$\overline{y \in \mathcal{C}}$	$\frac{n}{N}$	$\overline{y \in \mathcal{C}}$	$\frac{n}{N}$
TS115 ($N = 29,704$)																	
	RAPS$_{\text{SIZE}}$	0.96	0.22	0.89	0.17	0.88	0.06	0.87	0.12	0.82	0.14	0.98	0.28	0.90	0.43	0.93	0.57
	RAPS$_{\text{ADAPT}}$	0.96	0.24	0.90	0.14	0.86	0.07	0.88	0.09	0.82	0.17	0.99	0.23	0.90	0.48	0.94	0.46
	APS	0.96	0.24	0.89	0.14	0.86	0.07	0.88	0.09	0.83	0.17	0.99	0.23	0.90	0.48	0.94	0.46
	LOCAL$_{\text{CONF}}$	0.96	0.23	0.91	0.13	0.85	0.07	0.87	0.09	0.86	0.18	1.00	0.21	0.91	0.49	0.95	0.43
	ADMIT	0.96	0.23	0.89	0.09	0.88	0.07	0.92	0.07	0.76	0.14	0.99	0.18	0.88	0.44	0.95	0.34
	VENN-ADMIT	0.98	0.14	0.93	0.17	0.92	0.03	0.89	0.10	0.92	0.08	1.00	0.23	0.96	0.24	0.95	0.50
CASP12 ($N = 7,256$)																	
	RAPS$_{\text{SIZE}}$	0.96	0.14	0.87	0.17	0.85	0.05	0.86	0.16	0.77	0.13	0.97	0.36	0.87	0.31	0.92	0.69
	RAPS$_{\text{ADAPT}}$	0.95	0.16	0.88	0.13	0.85	0.06	0.88	0.12	0.78	0.15	0.98	0.29	0.86	0.36	0.93	0.54
	APS	0.95	0.15	0.90	0.13	0.86	0.05	0.88	0.13	0.74	0.15	0.98	0.28	0.85	0.36	0.94	0.54
	LOCAL$_{\text{CONF}}$	0.97	0.14	0.92	0.13	0.82	0.03	0.89	0.13	0.84	0.12	1.00	0.28	0.90	0.30	0.95	0.55
	ADMIT	0.94	0.16	0.88	0.07	0.87	0.06	0.91	0.09	0.67	0.13	0.99	0.21	0.83	0.35	0.95	0.37
	VENN-ADMIT	0.96	0.09	0.91	0.13	0.87	0.01	0.89	0.13	0.89	0.06	1.00	0.27	0.93	0.16	0.95	0.54

Table 5: Coverage on domain-shifted data. SENTIMENT (high acc., in-domain) for contrast.

			d_{ca} Quantile Bins (2 total) by Class Label (Binary **SSL** and **DC**)									
			$y = 0$		$y = 1$		$y \in \{0, 1\}$		$\lvert\mathcal{C}\rvert = 1, y = 0$		$\lvert\mathcal{C}\rvert = 1, y = 1$	
Set	Method	Metric	1st	2nd	1st	2nd	1st	2nd	$1^{\text{st}} + 2^{\text{nd}}$	($\frac{n}{N}$)	$1^{\text{st}} + 2^{\text{nd}}$	($\frac{n}{N}$)
SENTIMENTOOD ($N = 4750$)		n/N in bin	0.15	0.35	0.20	0.30	0.35	0.65	-	-	-	-
	$f(x)_{\text{tr}}^{\text{KNN}}$	ACC.	0.95	0.82	0.54	0.84	0.72	0.83	-	-	-	-
	CONF$_{\text{BASE}}$	$\overline{y \in \mathcal{C}}$	0.95	0.82	0.54	0.84	0.72	0.83	0.86	0.50	0.72	0.50
		$\overline{\lvert\mathcal{C}\rvert}$	1.00	1.00	1.00	1.00	1.00	1.00	1.	-	1.	-
	RAPS$_{\text{ADAPT}}$	$\overline{y \in \mathcal{C}}$	0.94	0.86	0.90	0.96	0.92	0.91	0.79	0.27	0.91	0.33
		$\overline{\lvert\mathcal{C}\rvert}$	1.45	1.46	1.41	1.28	1.43	1.38	1.	-	1.	-
	RAPS$_{\text{SIZE}}$	$\overline{y \in \mathcal{C}}$	0.84	0.71	0.68	0.88	0.75	0.79	0.75	0.50	0.80	0.50
		$\overline{\lvert\mathcal{C}\rvert}$	1.00	1.00	1.00	1.00	1.00	1.00	1.	-	1.	-
	APS	$\overline{y \in \mathcal{C}}$	0.94	0.87	0.89	0.97	0.91	0.91	0.80	0.28	0.91	0.33
		$\overline{\lvert\mathcal{C}\rvert}$	1.43	1.44	1.42	1.28	1.43	1.37	1.	-	1.	-
	LOCAL$_{\text{CONF}}$	$\overline{y \in \mathcal{C}}$	1.00	0.97	0.99	0.98	0.99	0.98	0.90	0.10	0.96	0.18
		$\overline{\lvert\mathcal{C}\rvert}$	1.84	1.78	1.77	1.55	1.80	1.67	1.	-	1.	-
	ADMIT	$\overline{y \in \mathcal{C}}$	1.00	0.98	0.97	0.98	0.98	0.98	0.96	0.15	0.92	0.17
		$\overline{\lvert\mathcal{C}\rvert}$	1.78	1.65	1.79	1.58	1.78	1.62	1.	-	1.	-
	VENN-ADMIT	$\overline{y \in \mathcal{C}}$	1.00	0.98	0.97	0.98	0.98	0.98	0.96	0.13	0.94	0.17
		$\overline{\lvert\mathcal{C}\rvert}$	1.78	1.72	1.79	1.59	1.78	1.66	1.	-	1.	-
SENTIMENT ($N = 488$)		n/N in bin	0.26	0.24	0.24	0.27	0.49	0.51	-	-	-	-
	$f(x)_{\text{tr}}^{\text{KNN}}$	ACC.	0.97	0.91	0.92	0.90	0.95	0.91	-	-	-	-
	CONF$_{\text{BASE}}$	$\overline{y \in \mathcal{C}}$	0.97	0.91	0.92	0.90	0.95	0.91	0.94	0.50	0.91	0.50
		$\overline{\lvert\mathcal{C}\rvert}$	1.00	1.00	1.00	1.00	1.00	1.00	1.	-	1.	-
	RAPS$_{\text{ADAPT}}$	$\overline{y \in \mathcal{C}}$	0.98	0.96	0.97	0.95	0.98	0.96	0.97	0.47	0.96	0.46
		$\overline{\lvert\mathcal{C}\rvert}$	1.02	1.10	1.08	1.08	1.05	1.09	1.	-	1.	-
	RAPS$_{\text{SIZE}}$	$\overline{y \in \mathcal{C}}$	0.97	0.91	0.93	0.90	0.95	0.91	0.94	0.50	0.91	0.50
		$\overline{\lvert\mathcal{C}\rvert}$	1.00	1.00	1.00	1.00	1.00	1.00	1.	-	1.	-
	APS	$\overline{y \in \mathcal{C}}$	0.98	0.95	0.97	0.95	0.97	0.95	0.96	0.47	0.95	0.46
		$\overline{\lvert\mathcal{C}\rvert}$	1.02	1.12	1.08	1.10	1.05	1.11	1.	-	1.	-
	LOCAL$_{\text{CONF}}$	$\overline{y \in \mathcal{C}}$	0.97	0.94	0.96	0.95	0.96	0.94	0.95	0.43	0.94	0.44
		$\overline{\lvert\mathcal{C}\rvert}$	1.10	1.17	1.10	1.13	1.10	1.15	1.	-	1.	-
	ADMIT	$\overline{y \in \mathcal{C}}$	0.98	0.94	0.97	0.92	0.98	0.93	0.95	0.41	0.93	0.39
		$\overline{\lvert\mathcal{C}\rvert}$	1.20	1.14	1.18	1.25	1.19	1.19	1.	-	1.	-
	VENN-ADMIT	$\overline{y \in \mathcal{C}}$	0.98	0.94	0.97	0.92	0.98	0.93	0.94	0.37	0.93	0.39
		$\overline{\lvert\mathcal{C}\rvert}$	1.20	1.33	1.18	1.25	1.19	1.29	1.	-	1.	-
GRAMMAR ($N = 92597$)		n/N in bin	0.19	0.74	0.03	0.04	0.22	0.78	-	-	-	-
	$f(x)_{\text{tr}}^{\text{KNN}}$	ACC.	1.00	0.98	0.25	0.28	0.90	0.94	-	-	-	-
	CONF$_{\text{BASE}}$	$\overline{y \in \mathcal{C}}$	1.00	0.98	0.28	0.32	0.91	0.95	0.98	0.92	0.26	0.06
		$\overline{\lvert\mathcal{C}\rvert}$	1.00	1.01	1.04	1.07	1.01	1.01	1.	-	1.	-
	RAPS$_{\text{ADAPT}}$	$\overline{y \in \mathcal{C}}$	0.97	0.98	0.50	0.59	0.91	0.96	0.97	0.78	0.34	0.05
		$\overline{\lvert\mathcal{C}\rvert}$	1.18	1.15	1.28	1.35	1.19	1.16	1.	-	1.	-
	RAPS$_{\text{SIZE}}$	$\overline{y \in \mathcal{C}}$	0.97	0.98	0.49	0.58	0.91	0.96	0.97	0.79	0.34	0.05
		$\overline{\lvert\mathcal{C}\rvert}$	1.17	1.15	1.26	1.34	1.19	1.16	1.	-	1.	-
	APS	$\overline{y \in \mathcal{C}}$	0.97	0.98	0.50	0.58	0.91	0.96	0.97	0.79	0.34	0.05
		$\overline{\lvert\mathcal{C}\rvert}$	1.18	1.15	1.27	1.34	1.19	1.16	1.	-	1.	-
	LOCAL$_{\text{CONF}}$	$\overline{y \in \mathcal{C}}$	1.00	1.00	0.35	0.38	0.91	0.96	1.00	0.85	0.19	0.05
		$\overline{\lvert\mathcal{C}\rvert}$	1.16	1.06	1.18	1.26	1.17	1.07	1.	-	1.	-
	ADMIT	$\overline{y \in \mathcal{C}}$	1.00	0.98	0.93	0.94	0.99	0.98	0.93	0.22	0.77	0.02
		$\overline{\lvert\mathcal{C}\rvert}$	1.83	1.75	1.72	1.73	1.81	1.75	1.	-	1.	-
	VENN-ADMIT	$\overline{y \in \mathcal{C}}$	1.00	1.00	0.99	1.00	1.00	1.00	1.00	0.09	0.88	<0.01
		$\overline{\lvert\mathcal{C}\rvert}$	1.87	1.91	1.88	1.97	1.87	1.91	1.	-	1.	-

REFERENCES

Anastasios N. Angelopoulos and Stephen Bates. A Gentle Introduction to Conformal Prediction and Distribution-Free Uncertainty Quantification. *CoRR*, abs/2107.07511, 2021. URL https://arxiv.org/abs/2107.07511.

Anastasios Nikolas Angelopoulos, Stephen Bates, Michael Jordan, and Jitendra Malik. Uncertainty Sets for Image Classifiers using Conformal Prediction. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=eNdiU_DbM9.

Rina Foygel Barber, Emmanuel J. Candès, Aaditya Ramdas, and Ryan J. Tibshirani. Predictive inference with the jackknife+. *The Annals of Statistics*, 49(1):486 – 507, 2021. doi: 10.1214/20-AOS1965. URL https://doi.org/10.1214/20-AOS1965.

Helen M Berman, John Westbrook, Zukang Feng, Gary Gilliland, Talapady N Bhat, Helge Weissig, Ilya N Shindyalov, and Philip E Bourne. The protein data bank. *Nucleic acids research*, 28(1): 235–242, 2000.

Ciprian Chelba, Tomás Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, Phillipp Koehn, and Tony Robinson. One Billion Word Benchmark for Measuring Progress in Statistical Language Modeling. In Haizhou Li, Helen M. Meng, Bin Ma, Engsiong Chng, and Lei Xie (eds.), *INTERSPEECH 2014, 15th Annual Conference of the International Speech Communication Association, Singapore, September 14-18, 2014*, pp. 2635–2639. ISCA, 2014. URL http://www.isca-speech.org/archive/interspeech_2014/i14_2635.html.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423.

Luc Devroye, László Györfi, and Gábor Lugosi. A Probabilistic Theory of Pattern Recognition. In *Stochastic Modelling and Applied Probability*, 1996.

Sara El-Gebali, Jaina Mistry, Alex Bateman, Sean R Eddy, Aurélien Luciani, Simon C Potter, Matloob Qureshi, Lorna J Richardson, Gustavo A Salazar, Alfredo Smart, Erik L L Sonnhammer, Layla Hirsh, Lisanna Paladin, Damiano Piovesan, Silvio C E Tosatto, and Robert D Finn. The Pfam protein families database in 2019. *Nucleic Acids Research*, 47(D1):D427–D432, 2019. ISSN 0305-1048. doi: 10.1093/nar/gky995. URL https://academic.oup.com/nar/article/47/D1/D427/5144153.

Rina Foygel Barber, Emmanuel J Candès, Aaditya Ramdas, and Ryan J Tibshirani. The limits of distribution-free conditional predictive inference. *Information and Inference: A Journal of the IMA*, 10(2):455–482, 08 2020. ISSN 2049-8772. doi: 10.1093/imaiai/iaaa017. URL https://doi.org/10.1093/imaiai/iaaa017.

Leying Guan. Localized Conformal Prediction: A Generalized Inference Framework for Conformal Prediction. *Biometrika*, 07 2022. ISSN 1464-3510. doi: 10.1093/biomet/asac040. URL https://doi.org/10.1093/biomet/asac040. asac040.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On Calibration of Modern Neural Networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, pp. 1321–1330. JMLR.org, 2017.

Ulf Johansson, Tuve Löfström, and Håkan Sundell. Venn predictors using lazy learners. In *The 2018 World Congress in Computer Science, Computer Engineering & Applied Computing, July 30-August 02, Las Vegas, Nevada, USA*, pp. 220–226. CSREA Press, 2018.

Divyansh Kaushik, Eduard Hovy, and Zachary Lipton. Learning The Difference That Makes A Difference With Counterfactually-Augmented Data. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=Sklgs0NFvr.

Michael Schantz Klausen, Martin Closter Jespersen, Henrik Nielsen, Kamilla Kjaergaard Jensen, Vanessa Isabell Jurtz, Casper Kaae Soenderby, Morten Otto Alexander Sommer, Ole Winther, Morten Nielsen, Bent Petersen, et al. Netsurfp-2.0: Improved prediction of protein structural features by integrated deep learning. *Proteins: Structure, Function, and Bioinformatics*, 2019.

Antonis Lambrou, Ilia Nouretdinov, and Harris Papadopoulos. Inductive Venn Prediction. *Annals of Mathematics and Artificial Intelligence*, 74(1):181–201, 2015. doi: 10.1007/s10472-014-9420-z. URL https://doi.org/10.1007/s10472-014-9420-z.

Jing Lei and Larry Wasserman. Distribution-free prediction bands for non-parametric regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 76(1):71–96, 2014. doi: https://doi.org/10.1111/rssb.12021. URL https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/rssb.12021.

Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization, 2019.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning Word Vectors for Sentiment Analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 142–150, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. URL https://aclanthology.org/P11-1015.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger (eds.), *Advances in Neural Information Processing Systems 26*, pp. 3111–3119. Curran Associates, Inc., 2013. URL http://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality.pdf.

John Moult, Krzysztof Fidelis, Andriy Kryshtafovych, Torsten Schwede, and Anna Tramontano. Critical assessment of methods of protein structure prediction (CASP)-Round XII. *Proteins: Structure, Function, and Bioinformatics*, 86:7–15, 2018. ISSN 08873585. doi: 10.1002/prot.25415. URL http://doi.wiley.com/10.1002/prot.25415.

Harris Papadopoulos, Kostas Proedrou, Volodya Vovk, and Alex Gammerman. Inductive confidence machines for regression. In *Proceedings of the 13th European Conference on Machine Learning*, ECML'02, pp. 345–356, Berlin, Heidelberg, 2002. Springer-Verlag. ISBN 3540440364. doi: 10.1007/3-540-36755-1_29. URL https://doi.org/10.1007/3-540-36755-1_29.

John C. Platt. Probabilistic Outputs for Support Vector Machines and Comparisons to Regularized Likelihood Methods. In *Advances in Large Margin Classifiers*, pp. 61–74. MIT Press, 1999.

Aleksandr Podkopaev and Aaditya Ramdas. Distribution-free uncertainty quantification for classification under label shift. In Cassio de Campos and Marloes H. Maathuis (eds.), *Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence*, volume 161 of *Proceedings of Machine Learning Research*, pp. 844–853. PMLR, 27–30 Jul 2021. URL https://proceedings.mlr.press/v161/podkopaev21a.html.

Roshan Rao, Nicholas Bhattacharya, Neil Thomas, Yan Duan, Xi Chen, John Canny, Pieter Abbeel, and Yun S Song. Evaluating Protein Transfer Learning with TAPE. In *Advances in Neural Information Processing Systems*, 2019.

Marek Rei and Helen Yannakoudakis. Compositional Sequence Labeling Models for Error Detection in Learner Writing. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1181–1191, Berlin, Germany, August 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1112. URL https://www.aclweb.org/anthology/P16-1112.

Yaniv Romano, Matteo Sesia, and Emmanuel J. Candès. Classification with valid and adaptive coverage. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, Red Hook, NY, USA, 2020. Curran Associates Inc. ISBN 9781713829546.

Sara Rosenthal, Noura Farra, and Preslav Nakov. SemEval-2017 Task 4: Sentiment Analysis in Twitter. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pp. 502–518, Vancouver, Canada, August 2017. Association for Computational Linguistics. doi: 10.18653/v1/S17-2088. URL https://www.aclweb.org/anthology/S17-2088.

Mauricio Sadinle, Jing Lei, and Larry A. Wasserman. Least ambiguous set-valued classifiers with bounded error levels. *Journal of the American Statistical Association*, 114:223 – 234, 2018.

Allen Schmaltz. Detecting Local Insights from Global Labels: Supervised and Zero-Shot Sequence Labeling via a Convolutional Decomposition. *Computational Linguistics*, 47(4):729–773, December 2021. doi: 10.1162/coli_a_00416. URL https://aclanthology.org/2021.cl-4.25.

Allen Schmaltz and Andrew Beam. Exemplar Auditing for Multi-Label Biomedical Text Classification. *CoRR*, abs/2004.03093, 2020. URL https://arxiv.org/abs/2004.03093.

Ryan J Tibshirani, Rina Foygel Barber, Emmanuel Candes, and Aaditya Ramdas. Conformal Prediction Under Covariate Shift. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/8fb21ee7a2207526da55a679f0332de2-Paper.pdf.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30, pp. 6000–6010. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Vladimir Vovk. Conditional validity of inductive conformal predictors. In Steven C. H. Hoi and Wray Buntine (eds.), *Proceedings of the Asian Conference on Machine Learning*, volume 25 of *Proceedings of Machine Learning Research*, pp. 475–490, Singapore Management University, Singapore, 04–06 Nov 2012. PMLR. URL https://proceedings.mlr.press/v25/vovk12.html.

Vladimir Vovk. Cross-conformal predictors. *Annals of Mathematics and Artificial Intelligence*, 74(1):9–28, 2015. doi: 10.1007/s10472-013-9368-4. URL https://doi.org/10.1007/s10472-013-9368-4.

Vladimir Vovk and Ivan Petej. Venn-Abers Predictors. In *UAI*, 2014.

Vladimir Vovk, Glenn Shafer, and Ilia Nouretdinov. Self-calibrating Probability Forecasting. In S. Thrun, L. Saul, and B. Schölkopf (eds.), *Advances in Neural Information Processing Systems*, volume 16. MIT Press, 2003. URL https://proceedings.neurips.cc/paper/2003/file/10c66082c124f8afe3df4886f5e516e0-Paper.pdf.

Vladimir Vovk, Alex Gammerman, and Glenn Shafer. *Algorithmic Learning in a Random World*. Springer-Verlag, Berlin, Heidelberg, 2005. ISBN 0387001522.

Vladimir Vovk, Ivan Petej, and Valentina Fedorova. Large-scale probabilistic predictors with and without guarantees of validity. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015. URL https://proceedings.neurips.cc/paper/2015/file/a9a1d5317a33ae8cef33961c34144f84-Paper.pdf.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. Transformers: State-of-the-Art Natural Language Processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 38–45, Online, October 2020. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/2020.emnlp-demos.6.

Helen Yannakoudakis, Ted Briscoe, and Ben Medlock. A New Dataset and Method for Automatically Grading ESOL Texts. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 180–189, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P11-1019.

Matthew D. Zeiler. ADADELTA: An Adaptive Learning Rate Method. *CoRR*, abs/1212.5701, 2012. URL http://arxiv.org/abs/1212.5701.

A APPENDIX: CONTENTS

Appendix B provides guidelines on controlling for—and conveying the variance of—the sample size. We provide pseudo-code in Appendix B. In Appendix D, E, and F, we provide additional details for each of the tasks.

B CONTROLLING FOR SAMPLE SIZE

Given a single sample from P_{XY} (i.e., our single $\mathcal{D}_{ca}$ of some fixed size), we need to convey the variance due to the observed sample size. We opt for a simple hard threshold, κ, given that the distribution of split-conformal coverage is Beta distributed (Vovk, 2012). With, for example $\kappa = 1000$, assuming exchangeability, the finite-sample guarantee then implies $\approx \pm \leq 0.02$ coverage variation within a conditioning band of size ≥ 1000 with $\alpha = 0.1, |\mathcal{D}_{te}| = \infty$. See the comprehensive tutorial Angelopoulos & Bates (2021) for additional details. In our experiments, if the size of at least 1 label-specific band for a given point falls below κ, we revert to a set of full cardinality. For the PROTEIN, SENTIMENT, and SENTIMENTOOD tasks, we set $\kappa = 1000$. With the low accuracy and low frequency of the minority class in the GRAMMAR task, the $q = K$ partition is comparatively small. As such, for the GRAMMAR task, we set $\kappa = 100$ to avoid heavily censoring the $q = K$ partition, at the expense of potentially higher variability.

C PSEUDO-CODE

Algorithm 1 constructs an ADMIT prediction set for a test point, x_t. Standard, unweighted calibration via the VENN-ADMIT Predictor appears in Algorithm 2. This assumes prediction sets have been constructed for the test point and each calibration point, $x_j \in \mathcal{D}_{\text{ca}}$. When computationally feasible to use the additional KNN localizer, $f(x)_{\text{ca}}^{\widehat{\text{KNN}}}$, as in the experiments in the main text, a simple re-weighting of the augmented test point can add additional robustness across distribution shifts, without harming in-distribution effectiveness, as shown in Algorithm 3.

Algorithm 1 ADMIT Prediction Sets via Neural Model Approximations

Input: $\mathcal{D}_{\text{ca}}$, $(x_t \in \mathcal{D}_{\text{te}}, q_t, d_t)$, band radius ω, $f(x)_{\text{tr}}^{\text{KNN}}$, α

procedure _THRESHOLD($\mathcal{I}', \alpha$) ▷ Standard split-conformal if $\mathcal{I}' = \mathcal{D}_{\text{ca}}$
  $\mathcal{S}_j \leftarrow s(x_j) = 1 - \hat{\pi}^y(x_j)_{\text{tr}}^{\text{KNN}}, \ \forall\, x_j \in \mathcal{I}'$ ▷ Conformity scores over calibration subset
  $\hat{l}^\alpha \leftarrow \lceil (|\mathcal{I}'| + 1)(1-\alpha) \rceil / |\mathcal{I}'|$ quantile of $\mathcal{S}$
  return $\hat{\tau}^\alpha \leftarrow 1 - \hat{l}^\alpha$
procedure ADMIT($\mathcal{D}_{\text{ca}}, x_t, q_t, d_t, \omega, f(x)_{\text{tr}}^{\text{KNN}}, \alpha$)
  $\hat{\mathcal{C}}(x_t) \leftarrow \{\hat{y}_t^{\text{KNN}}\}$
  $\mathcal{I} \leftarrow \mathcal{B}(x_t, \omega, q_t, d_t; \mathcal{D}_{\text{ca}})$ ▷ Calibration points in band centered at x_t (Eq. 11)
  for $c \in \{1, \ldots, C\}$ **do**
    $\mathcal{I}^c \leftarrow \{x_j : x_j \in \mathcal{I}, y_j = c\}$ ▷ Subset of band for which true class is c
    $\hat{\tau}_c^\alpha \leftarrow$ _THRESHOLD($\mathcal{I}^c, \alpha$)
    $\hat{\mathcal{C}}(x_t) \leftarrow \hat{\mathcal{C}}(x_t) \cup \left\{ c : \hat{\pi}^c(x_t)_{\text{tr}}^{\text{KNN}} \geq \hat{\tau}_c^\alpha \right\}$
  return $\hat{\mathcal{C}}(x_t)$

Output: $\hat{\mathcal{C}}(x_t)$, prediction set with coverage of Eq. 5

D TASK: PROTEIN SECONDARY STRUCTURE PREDICTION (PROTEIN)

In the supervised sequence labeling PROTEIN task, we seek to predict the secondary structure of proteins. For each amino acid, we seek to predict one of three classes, $y \in \{$**H**ELIX, **S**TRAND, **O**THER$\}$.

For training and evaluation, we use the TAPE datasets of Rao et al. (2019).[5] We approximate the Transformer of Rao et al. (2019), which is *not* SOTA on the task; while not degenerate, this

[5]TAPE provides a standardized benchmark from existing models and data (El-Gebali et al., 2019; Berman et al., 2000; Moult et al., 2018; Klausen et al., 2019).

Algorithm 2 Calibration via Inductive VENN-ADMIT Predictor (standard, unweighted)

Input: $\mathcal{D}_{\text{ca}}$, $(x_t \in \mathcal{D}_{\text{te}}, q_t, d_t)$, band radius ω, $f(x)_{\text{tr}}^{\text{KNN}}$

procedure _CATEGORY($\mathcal{I}, x_t, y'$)
  $\mathcal{T} \leftarrow \{(x_t, y')\}$ ▷ y' is the *assumed* true label for x_t
  for $x_j \in \mathcal{I}$ **do**
    if $\hat{y}_t^{\text{KNN}} = \hat{y}_j^{\text{KNN}} \wedge \hat{\mathcal{C}}(x_t) = \hat{\mathcal{C}}(x_j)$ **then** ▷ $\hat{\mathcal{C}}$ calculated as in Alg. 1
      $\mathcal{T} \leftarrow \mathcal{T} \cup \{(x_j, y_j)\}$ ▷ y_j is the true label for x_j
  return $\mathcal{T}$
procedure VENN-ADMIT($\mathcal{D}_{\text{ca}}, x_t, q_t, d_t, \omega, f(x)_{\text{tr}}^{\text{KNN}}$)
  $\mathcal{I} \leftarrow \mathcal{B}(x_t, \omega, q_t, d_t; \mathcal{D}_{\text{ca}})$ ▷ Calibration points in band centered at x_t (Eq. 11)
  for $c \in \{1, \ldots, C\}$ **do**
    $\mathcal{T} \leftarrow$_CATEGORY($\mathcal{I}, x_t, c$)
    for $c' \in \{1, \ldots, C\}$ **do**
      $p_c(c') \leftarrow \frac{|\{(x^*,\, y^*) \in \mathcal{T}\colon\, y^* = c'\}|}{|\mathcal{T}|}$ ▷ Empirical probability
  for $c' \in \{1, \ldots, C\}$ **do**
    $\overline{p(c')} \leftarrow \frac{1}{C} \sum_{c=1}^{C} p_c(c')$ ▷ Mean probability for each class (across augmented distributions)
  return $\overline{p(\cdot)}$

Output: $\overline{p(\cdot)}$, calibrated distribution for x_t.

Algorithm 3 Calibration via Inductive VENN-ADMIT Predictor (weighted)

Input: $\mathcal{D}_{\text{ca}}$, $(x_t \in \mathcal{D}_{\text{te}}, q_t, d_t)$, band radius ω, $f(x)_{\text{tr}}^{\text{KNN}}$, localizer $f(x)_{\text{ca}}^{\widehat{\text{KNN}}}$

procedure _CATEGORY($\mathcal{I}, x_t, y'$) ▷ Same as in Alg. 2
procedure WEIGHTED-VENN-ADMIT($\mathcal{D}_{\text{ca}}, x_t, q_t, d_t, \omega, f(x)_{\text{tr}}^{\text{KNN}}, f(x)_{\text{ca}}^{\widehat{\text{KNN}}}$)
  $\mathcal{I} \leftarrow \mathcal{B}(x_t, \omega, q_t, d_t; \mathcal{D}_{\text{ca}})$ ▷ Calibration points in band centered at x_t (Eq. 11)
  for $c \in \{1, \ldots, C\}$ **do**
    $\mathcal{T} \leftarrow$_CATEGORY($\mathcal{I}, x_t, c$) $\setminus \{(x_t, c)\}$ ▷ Exclude x_t from the category
    $\psi' \leftarrow \sum_{\{j\colon\, x_j \in \mathcal{T}\}} \psi_j$ ▷ Sum of weights from KNN localizer Eq. 12; $\psi' \in (0, 1]$
    for $c' \in \{1, \ldots, C\}$ **do**
      $p_c(c') \leftarrow \frac{|\{(x^*,\, y^*) \in \mathcal{T}\colon\, y^* = c'\}| + [c = c'] \cdot (\frac{1}{\psi'})}{|\mathcal{T}| + \frac{1}{\psi'}}$ ▷ Test-point weighted empirical probability
  for $c' \in \{1, \ldots, C\}$ **do**
    $\overline{p(c')} \leftarrow \frac{1}{C} \sum_{c=1}^{C} p_c(c')$ ▷ Mean probability for each class (across augmented distributions)
  return $\overline{p(\cdot)}$

Output: $\overline{p(\cdot)}$, calibrated distribution for x_t.

fine-tuned self-supervised model was outperformed by models with HMM alignment-based input features in the original work. Of interest in the present work is whether coverage can be obtained with a neural model with otherwise relatively modest overall point accuracy. We use the publicly available model and pre-trained weights[6].

D.1 MEMORY LAYER

The base network consists of a pre-trained Transformer similar to $\text{BERT}_{\text{BASE}}$ with a final convolutional classification layer, consisting of two 1-dimensional CNNs: The first over the final hidden layer of the Transformer corresponding to each amino acid (each hidden layer is of size 768), using 512 filters of width 5, followed by ReLU and a second CNN using 3 filters of width 3. Batch normalization is applied before the first CNN, and weight normalization is applied to the output of each of the CNNs. The application of the 3 filters of the final CNN produces the logits, $\mathbb{R}^3$, for each amino acid.

The MEMORY LAYER consists of an additional 1-dimensional CNN, which uses 1000 filters of width 1. The input to the MEMORY LAYER corresponding to each amino acid is the concatenation of the final hidden layer of the Transformer, the output of the final CNN of the base network, and a randomly initialized 10-dimensional word-embedding. The output of the CNN is passed to a LinearLayer of

[6] https://github.com/songlab-cal/tape

dimension 1000 by 3. (Unlike the sparse supervised sequence labeling task of GRAMMAR, we use neither a ReLU, nor a max-pool operation. The sequences are very long in this setting—up to 1000 used in training and 1632 at inference to avoid truncation—so removing the max-pool bottleneck enables keeping the number of filters of the CNN lower than the total number of amino acids. In this way, we also do not use the decomposition of the CNN with the LinearLayer, as in the GRAMMAR task, since the sparsity over the input is not needed for this task.) The exemplar vectors for the KNNs are then the $\boldsymbol{r} \in \mathbb{R}^{1000}$ filter applications of the CNN corresponding to each amino acid.

We fine-tune the base network and train the MEMORY LAYER in an iterative fashion. Each epoch we either update the gradients of the base network, or those of the MEMORY LAYER, freezing the counter-part each epoch. We start by updating the base network (and freezing the MEMORY LAYER), and we use separate optimizers for each: Adadelta (Zeiler, 2012) with a learning rate of 1.0 for the MEMORY LAYER and Adam with weight decay (Loshchilov & Hutter, 2019) with a learning rate of 0.0001 and a warmup proportion of 0.01 for the base network. For the latter, we use the BertAdam code from the HuggingFace re-implementation of Devlin et al. (2019). We fine-tune for up to 16 epochs, and we use a standard cross-entropy loss.

For reference, Figures 3 and 4 show results for the PROTEIN task at the $\alpha = 0.05$ level. The results are in the same direction as with the $\alpha = 0.1$ level shown in the main text.

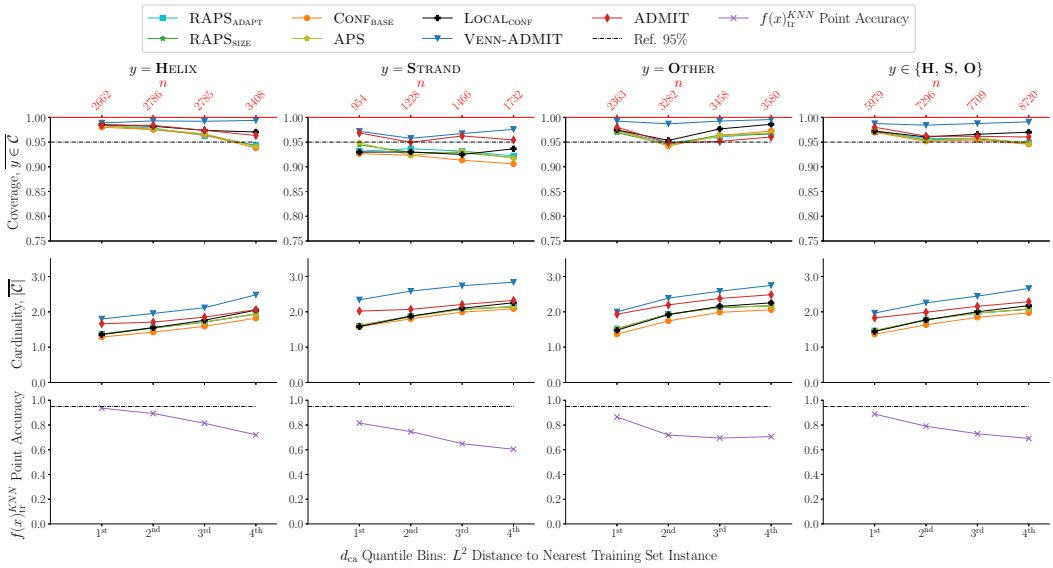

Figure 3: Coverage, cardinality, and point accuracy for the TS115 test set from the PROTEIN task, $\alpha = 0.05$.

E SUPERVISED GRAMMATICAL ERROR DETECTION (GRAMMAR)

The GRAMMAR task is a binary sequence labeling task in which we aim to predict whether each word in the input does ($y_t = 1$) or does not ($y_t = 0$) have a grammatical error. $\mathcal{D}_{\text{tr}}$ and $\mathcal{D}_{\text{ca}}$ consist of essays written by second-language learners (Yannakoudakis et al., 2011; Rei & Yannakoudakis, 2016) and $\mathcal{D}_{\text{te}}$ consists of student written essays *and* newswire text (Chelba et al., 2014). The test set is the FCE+NEWS2K set of Schmaltz (2021).

The test set is challenging for two reasons. First, the $y = 1$ class appears with a proportion of 0.07 of all of the words. This is less than our default value for α, with the implication that marginal coverage can potentially be obtained by altogether ignoring that class. Second, the in-domain task itself is relatively challenging, but it is made yet harder by adding newswire text, as evident in the large $F_{0.5}$ score differences across $\mathcal{D}_{\text{ca}}$ and $\mathcal{D}_{\text{te}}$ in Table 2.

The exemplar vectors, $\boldsymbol{r} \in \mathbb{R}^{1000}$, used in the KNNs are extracted from the filter applications of a penultimate CNN layer over a frozen BERT$_{\text{LARGE}}$ model, as in Schmaltz (2021).

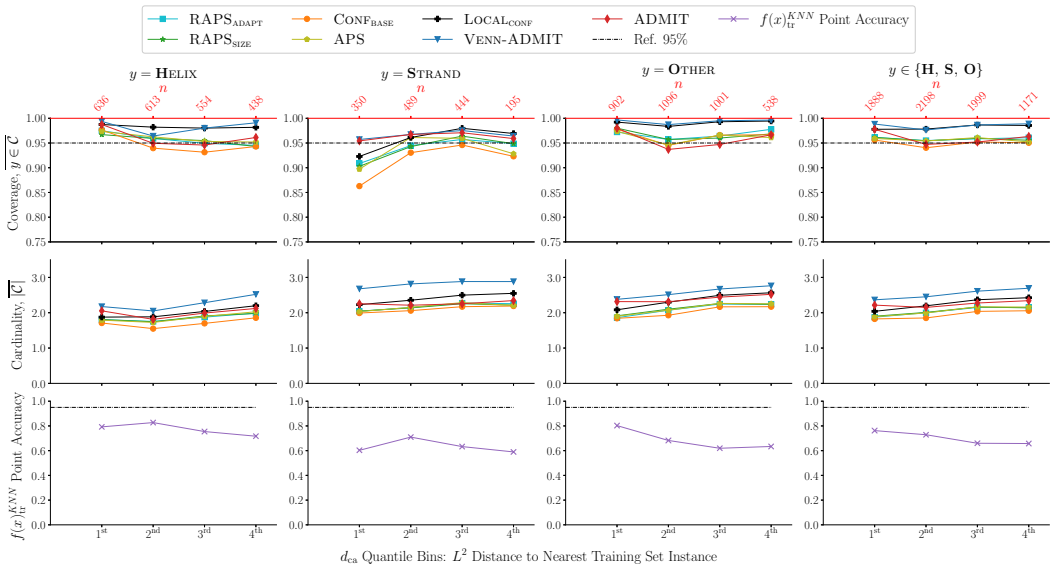

Figure 4: Coverage, cardinality, and point accuracy for the CASP12 test set from the PROTEIN task, $\alpha = 0.05$.

F TASKS: SENTIMENT CLASSIFICATION (SENTIMENT) AND OUT-OF-DOMAIN SENTIMENT CLASSIFICATION (SENTIMENTOOD)

SENTIMENT and SENTIMENTOOD are document-level binary classification tasks in which we aim to predict whether the document is of negative ($y = 0$) or positive ($y = 1$) sentiment. The training and calibration sets, as well as the base networks, are the same for both tasks, with the distinction in the differing test sets. The training set is the 3.4k IMDb movie review set used in Kaushik et al. (2020) from the data of Maas et al. (2011). For calibration, we use a disjoint 16k set of reviews from the original training set of Maas et al. (2011). The test set of SENTIMENT is the 488-review in-domain test set of original reviews used in Kaushik et al. (2020), and the test set of SENTIMENTOOD consists of 5k Twitter messages from SemEval-2017 Task 4a (Rosenthal et al., 2017).

Similar to the GRAMMAR task, the exemplar vectors, $\boldsymbol{r} \in \mathbb{R}^{2000}$, are derived from the filter applications of a penultimate CNN layer over a frozen $\text{BERT}_{\text{LARGE}}$ model. However, in this case, the vectors are the concatenation of the document-level max-pooled vector, $\boldsymbol{r} \in \mathbb{R}^{1000}$, and the vector associated with a single representative token in the document, $\boldsymbol{r} \in \mathbb{R}^{1000}$. To achieve this, we model the task as multi-label classification and fine-tune the penultimate layer CNN and a final layer consisting of two linear layers with the combined min-max and global normalization loss of Schmaltz & Beam (2020). In this way, we can associate each word with one of (or in principle, both) positive and negative sentiment, or a neutral class, while nonetheless having a single exclusive global prediction. This provides sparsity over the detected features, and captures the notion that a document may, in totality, represent one of the classes (e.g., indicate a positively rated movie overall) while at the same time including sentences or phrases that are of the opposite class (e.g., aspects that the reviewer rated negatively). This behavior is illustrated with examples from the calibration set in Table 6. We use the max scoring word from the "convolutional decomposition", a hard-attention-style approach, for the document-level predicted class as the single representative word for the document. For the document-level prediction, we take the max over the multi-label logits, which combine the global and max local scores.

Table 6: Model feature detections from snippets from $\mathcal{D}_{\text{ca}}$ for SENTIMENT and SENTIMENTOOD, for which prediction sets are constructed for the binary document-level predictions. Most documents only have features of a single class detected (as in the example in the final row), but our modeling choice (Section F) does enable multi-label detection as in the first example, for which the true document label is positive sentiment, and the second example, for which the true document label is negative sentiment. The max scoring word for each document is underlined.

Model predictions over $\mathcal{D}_{\text{ca}}$
What an amazing film. *[...]* My only gripe is that it has not been released on video in Australia and is therefore only available on TV. What a waste.
[...] But the story that then develops lacks any of the stuff that these opening fables display. *[...]* I will say that the music by Aimee Mann was great and I'll be looking for the Soundtrack CD. *[...]*
Kenneth Branagh shows off his excellent skill in both acting and writing in this deep and thought provoking interpretation of Shakespeare's most classic and well-written tragedy. *[...]*