Error-Sensitive Evaluation for Ordinal Target Variables

David Z. Chen **Maury Courtland** **Adam Faulkner** **Aysu Ezen Can**
Capital One, Vision and Language Technologies
{david.chen2, maury.courtland, adam.faulkner, aysu.ezencan}
@capitalone.com

Abstract

Product reviews and satisfaction surveys seek customer feedback in the form of ranked scales. In these settings, widely used evaluation metrics including F1 and accuracy ignore the rank in the responses (e.g., 'very likely' is closer to 'likely' than 'not at all'). In this paper, we hypothesize that the order of class values is important for evaluating classifiers on ordinal target variables and should not be disregarded. To test this hypothesis, we compared Multi-class Classification (MC) and Ordinal Regression (OR) by applying OR and MC to benchmark tasks involving ordinal target variables using the same underlying model architecture. Experimental results show that while MC outperformed OR for some datasets in accuracy and F1, OR is significantly better than MC for minimizing the error between prediction and target for all benchmarks, as revealed by error-sensitive metrics, e.g. mean-squared error (MSE) and Spearman correlation. Our findings motivate the need to establish consistent, error-sensitive metrics for evaluating benchmarks with ordinal target variables, and we hope that it stimulates interest in exploring alternative losses for ordinal problems.

1 Introduction

Organizations have a vested interest in ensuring customer happiness. To measure this quantity, analysts often use surveys containing numerical Likert scales addressing various aspects of the customer experience (Allen and Seaman, 2007). One popular question asks customers the likelihood that they will recommend a product or service to others. From these answers, analysts calculate a "Net Promoter Score" (NPS) representing the percentage of customers who will recommend a product or service to others minus those who will recommend against it (Reichheld, 2003). Additionally, many companies are also interested in tracking product reviews (Keung et al., 2020). Collecting, measuring, and analyzing customer feedback is essential to the profitability and long-term success of many companies, but it is prohibitively expensive to survey the entire customer base and even the feedback that a company does receive is often too massive for systematic human evaluation. Therefore, it is important to develop effective machine learning models for predicting customer satisfaction and to maintain consistent and accurate methods and metrics for evaluating their performance.

An important aspect of modeling customer sentiment is the subjective numerical ranking of the customer response. Feedback is often in the form of ranked scales, e.g. rating scales 1-5 or 1-10, or textual "Strongly Agree," "Disagree," etc. Crucially, these Likert scales are ordinal and should not be confused with scalar values: a rating of 5 or "Strongly Agree" is not necessarily 5 times greater than a rating of 1 or "Strongly Disagree" (Allen and Seaman, 2007). To predict ordinal target variables from textual input, we explored a variety of commonplace and cutting-edge NLP techniques, ranging from linear models such as Naive Bayes and Logistic Regression to Transformer-based approaches such as BERT (Devlin et al., 2019) and "Performer" (Choromanski et al., 2021). One of the most striking observations from our experiments was that the most impactful experimental variable was not necessarily the model architecture itself, but rather the loss function employed and classification scheme, i.e. target variable encoding. Specifically, we highlight our use of Ordinal Regression (OR), which is an approach that is underutilized in the field of NLP. This approach has a long history (Frank and Hall, 2001; Graepel and Obermayer, 1999; Gutiérrez et al., 2016; Baccianella et al., 2009)

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 189–199
July 5–10, 2020.
https://doi.org/10.26615/978-954-452-056-4_019

and has recently been applied to deep neural network models predicting ordinal labels (Cao et al., 2020). Here, we extend this framework into the domain of NLP and apply it to train transformer models as a novel application of previous work on OR and transformers. Our results showed that OR produced a distribution of predictions significantly closer to ground truth distributions than Multi-class Classification (MC) for both NPS and survey ratings (i.e. lower K-L divergence), while producing similar accuracies and F1. Additionally, we found that OR improved the correlation between model predictions and ground truth. Our findings highlight that common NLP metrics are insufficient to distinguish the better model(s) for our task, which motivated us to explore and identify error-sensitive metrics more consistent and effective for evaluating models with ordinal target variables.

2 Related work

There has been a longstanding debate (Knapp, 1990; Joshi et al., 2015) within the survey, psychometrics, and crowdsourcing methodology community regarding the use of Likert scales. In particular, a debate on whether the level of measurement in question is interval or ordinal (McCall, 2001) and whether parametric or nonparametric tests should should be used (Kuzon et al., 1996). In a classification setting involving ordinal data, a related question involves the use of multinomial versus ordinal modeling for such data. Intuitively, the inclusion of ordinality in the classification model should improve performance relative to a multinomial approach, as shown in Campbell and Donner 1989. Despite the prevalence of ordinally-scaled tasks in NLP, such as in sentiment analysis (Jiang et al., 2019; Pang and Lee, 2005), stance classification (Sobhani et al., 2015), lexical specificity (Gao et al., 2019), political bias (Baly et al., 2018), and commonsense inference (Zhang et al., 2017), approaches to such tasks have tended to ignore the ordinal nature of the data, treating them as MC tasks. For example, only 2 of 11 participants in subtask C of SemEval 2016 (Rosenthal et al., 2017) (a 5-point-scale twitter sentiment classification task) chose to exploit the ordinal nature of the task in their models. Justifying this choice would involve systematically comparing the use of ordinal versus MC models for these tasks, yet most such comparisons have been reported as an incidental part of experiments for tasks such as sentiment analysis in tweets (Saad and Yang, 2019) and classification of psychiatric symptom severity in clinical notes (Rios and Kavuluru, 2017).

3 Methods

3.1 Datasets

We hypothesize that taking ordinal rankings into account would provide more consistent and better results. To test this, we experimented on 3 common benchmark datasets for sentiment analysis, and 1 additional dataset on Twitter specificity.

Movie Reviews (MR): We used the dataset named “scale dataset v1.0” (Pang and Lee, 2005). It contains full-length movie reviews from 4 authors on RottenTomatoes.com, and we used the 4-class variant, which is obtained by segmenting the ratings from the authors' normalized numerical scale into 4 ranks. Each author a, b, c, and d has 1770, 902, 1307 and 1027 reviews, respectively. The mean number of words per review for each author is 435, 374, 455 and 292, respectively, but the tail is long, with some reviews having 3k words or more. For this dataset, no test data was provided, so we report results on a test set from an 80%/20% train/test split of the dataset.

IMDb: This dataset is titled IMDb Large Movie Dataset, and includes 50k movie reviews written by users on the website IMDb.com (25k train, 25k test) (Maas et al., 2011). Each review had a rating of between 1-4 and 7-10 for low and high sentiment, respectively. Each movie in the dataset is reviewed less than 30 times and no movies reviewed in the training set also appears in the test set. For our experiments, we make use of the fine-grained labels, 1-4 and 7-10, which creates an 8-class classification problem. The reviews vary greatly by length, with some as short as 6 words and others up to 2.4k words.

SST-5: The SST-5 dataset is obtained from Socher et al. 2013, which consists of 215,154 unique phrases parsed from the corpus of 11,855 sentences (averaging 17 words each) from Pang and Lee 2005. Each sentence is labeled by 3 annotators. The labeling interface utilizes a continuous sliding bar with guiding ticks indicating “Very negative,” “Negative,” “Somewhat negative,” “Neutral,” “Somewhat positive,” “Positive,” and “Very positive.” For the SST-5 fine-grained sentiment classification version, the slider responses are collapsed down to 5 ranks.

Specificity: This is a corpus of 7,267 tweets that were sampled by taking 2 tweets (excluding re-tweets) from users who have posted at least 4 tweets (3,665 users) (Gao et al., 2019). The specificity annotations were based on a sliding-scale with 5 guiding options: “1 - Very General,” “2 - General,” “3 - Specific,” “4 - Very Specific,” and “5 - Extremely Specific,” where general refers to posts that do not make references to any specific person, object or event, and specific refers to posts that do. Each tweet was annotated by at least 5 workers on Amazon Mechanical Turk after filtering for low quality labels, with a resulting intra-class correlation coefficient of 0.575. In order to arrive at ordinal labels, we bin and collapse the specificity ratings from a continuous 1-5 scale down to ranks of 1, 2, 3, and 4, where each continuous numerical value, i, is rounded down, i.e. $floor(i)$, in order to reduce class imbalance.

3.2 Model architectures and parameters

All our models share the same underlying architecture in order to minimize differences resulting from model parameterization and feature generation. Model sizes (i.e. # of parameters) differ for each dataset, as the average input sequence lengths are different (e.g. Tweets vs. movie reviews). In general, the model parameters of “sequence length,” “embedding dimension,” and “feature size” are scaled to the maximum number of tokens contained in any given input text from that particular dataset. Because both the MR and IMDb datasets contain reviews that are longer than typical pre-trained Transformer input sequence sizes, we leverage the “Performer” model architecture (Choromanski et al., 2021) for all models in order to accommodate these inputs without modifying the underlying model architecture. Finally, we train our models without any pre-training and without pre-trained embeddings so that we can have meaningful comparisons between the OR and MC methodologies with fewer confounding variables, as adding pre-training or pre-trained embeddings may change how effectively each methodology learns from the data.

3.3 Loss functions and label encodings

For MC models, we use the Performer architecture with cross-entropy loss. The ordinal labels are encoded as one-hot vectors of length K, where K is the number of classes, i.e. ratings. For OR models, we also use the Performer architecture with cross-entropy, but over $K-1$ classes, where each class represents a threshold decision of whether the rating is predicted to be greater than a value, e.g. is the rating greater than 2. To accommodate the OR loss function, which we adopt from COnsistent RAnk Logits (CORAL) (Cao et al., 2020), we encode the ordinal target variables as vectors, $\boldsymbol{v}$, with length $N-1$, and each index v_k represents a binary indicator of rank threshold, i.e. $v_1 = 1$ if $\hat{y}^1 > 0.5$ and $v_2 = 1$ if $\hat{y}^2 > 0.5$, etc. CORAL seeks to optimize the ordinal rank by penalizing misclassifications of thresholds, and it has theoretical guarantees for rank-consistency (rank-monotonicity), e.g. in order to predict 5 (i.e. >4), the model must also predict >1, >2, >3. This is achieved by allowing the $K-1$ binary thresholds to share the same weight parameters, $\boldsymbol{W}$, but independent biases, b_k. Specifically, we seek to minimize:

$$L(\boldsymbol{W}, \boldsymbol{b}) = \\ -\sum_{i=1}^{N}\sum_{k=1}^{K-1} \lambda^k(\log(\hat{y}_i^k)y_i^k + \log(1-\hat{y}_i^k)(1-y_i^k))$$

where N is the number of training examples, λ^k is the weight associated with the k^{th} rank threshold, and $\hat{y}_i^k = \sigma(g(x_i, W) + b_k)$. Here, $\sigma()$ is the logistic sigmoid function, x_i is the input example, $\boldsymbol{W}$ are the model weight parameters, which are shared for all the binary rank thresholds, and b_k is the independent bias unit for threshold at rank k. In optimizing this loss function, one can prove that the independent bias units will rank order and result in overall rank-monotonicity, i.e. $b_1 \geq b_2 \geq b_3, etc.$ (see Theorem 1 in Cao et al. 2020). Due to this rank-consistency, a model prediction of $\hat{y}^3 > 0.5$ is always accompanied by both $\hat{y}^2 > 0.5$ and $\hat{y}^1 > 0.5$. In other words, for $v_3 = 1$, both

Dataset	Scheme	MSE	Spearman Rho	K-L divergence	Accuracy	F1(macro)	F1(weighted)
MR	MC	0.577	0.611	58.73	50.4	43.9	51.8
	OR	**0.491**	**0.684**	**6.47**	**56.9**	**55.3**	**57.1**
IMDb	MC	3.878	0.719	324.64	**35.0**	**28.1**	**36.4**
	OR	**2.855**	**0.763**	**179.28**	31.2	27.8	30.1
SST-5	MC	2.520	0.308	264.32	**33.2**	**29.9**	**35.2**
	OR	**2.051**	**0.362**	**136.29**	31.3	29.4	31.6
Specificity	MC	0.438	0.502	1057.67	59.5	37.4	62.6
	OR	**0.367**	**0.591**	**18.16**	**64.2**	**50.0**	**65.0**

Table 1: Experimental results for the benchmark datasets comparing OR to MC. Bolded numbers represent higher values except MSE and K-L divergence, where lower is better.

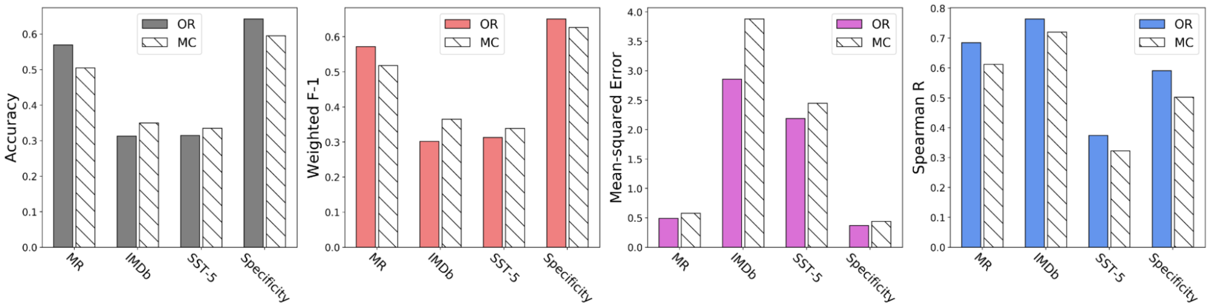

Figure 1: Across datasets, OR outperforms MC on MSE (purple; **lower is better**) and Spearman Rho (blue). The results are mixed for Accuracy (gray) and Weighted-F1 (red).

$v_2 = 1$ and $v_1 = 1$ are also true, and we encode a rating of 4 out of 5 as $\boldsymbol{v} = [1110]$. The model prediction, p, is then the first index at which $v_i = 0$, $argmin(\boldsymbol{v})$, or 5 if $v_i = 1$ for all i in [1..4].

3.4 Parameter tuning

For all experiments, we use existing train/test splits for the benchmark dataset. If no train/test splits exist, we generate a 80%/20% train/test split and randomly select 20% of the training split as a validation split for tuning model parameters and early-stopping. To determine the early-stopping point, we select the training epoch at the inflection point where a 5-epoch moving average of the validation loss no longer improves.

4 Results

Our experiments with MC vs OR classifiers for benchmark datasets (Table 1 and Figure 1) show: 1) Performance, measured in accuracy and F1, varies depending on the underlying dataset, with IMDb and SST-5 favoring MC and MR and Specificity favoring OR. Also, accuracy and F1 are correlated across models and datasets. 2) In contrast, mean-squared error (MSE), Spearman's Rho, and K-L divergence consistently favor OR, with OR achieving the lowest MSE and K-L divergence and highest Spearman Rho across all datasets. Here, MSE could be replaced by mean absolute error (MAE) or root-mean-squared error (RMSE), both of which also select OR as the better methodology over MC (not shown). These observations suggest that typical model evaluation metrics such as accuracy and F1 score, frequently used in benchmarks and leaderboards for sentiment analysis (Ribeiro et al., 2016; Ruder, 2021; Barbieri et al., 2020), may not successfully select the best performing models for classifying ordinal target variables.

For all of our experiments, we determined that our OR models are significantly different from MC models, with OR and MC predictions producing statistically distinct distributions as determined by a paired t-test with p values less than 0.05.

4.1 Dataset benchmarks

While the goal of our experiments is to compare OR vs MC in a baseline setting and not to challenge current state-of-the-art models, we provide our results against benchmarks on the sentiment analysis datasets in order to give additional context to our models' performance.

MR: Benchmarks and leaderboards on this specific dataset are sparse, as most researchers have opted to use the sentence-level polarity dataset from the same authors. The best-performing model we have found is from Bickerstaffe and Zukerman 2010, which achieved author-level accuracies of 65.72%, 52.89%, 66.99%, and 51.87%, respectively (from 10-fold CV). This, on average, out-performed our best-performing OR model, which achieved author-level accuracies of 62.12%, 54.78%, 54.85%, and 52.29%, respectively (from 20% test split). Specifically, Bickerstaffe and Zukerman 2010 out-performed on the majority-authors a and c while OR had a slight edge in the minority-authors b and d. OR out-performed the original models shown in Pang and Lee 2005.

IMDb: There is a lack of available benchmark data on the fine-grained 8-class version of the IMDb dataset, as most researchers opt to experiment on collapsing the labels to a binary classification problem. We obtain 34.98% and 31.19% accuracies for MC and OR, respectively on the fine-grained 8-class task. In an attempt to compare our result with previous work on the binary task, we collapse our predictions into binary format, by mapping predictions 1-4 to 0 and 7-10 to 1. With binary-mapping, we obtain 86.71% and 79.93% accuracies for OR and MC. The OR binary-mapped accuracy is on-par with previous benchmark results on binary IMDb predictions with vanilla CNNs and LSTMs (Camacho-Collados and Pilehvar, 2018). Interestingly, OR outperforms MC for the binary-mapped accuracy while the 8-class accuracy favors MC. This further suggests that OR is more effective at minimizing rank-error compared to MC, as errors in the binary-mapped case represent greater magnitudes than in the 8-class case.

SST-5: Our OR and MC models obtained 31.27% and 33.21% accuracy, respectively. This is substantially lower than baseline results in the literature, which typically achieve >40% accuracy (Socher et al., 2013), with current SOTA in the 50%'s (Khodak et al., 2018). The most comparable model from the literature is the "VecAvg" variant in the original SST-5 paper, which is a word-embedding model that has fine-grained accuracy of 32.7% (Socher et al., 2013). The poor performance exhibited by our OR and MC models on SST-5 might be due to a couple factors: 1) our omitting pre-training in the form of a pre-trained language model or word embeddings, whereas typical protocols for SST-5 train embedding representations on additional data (Khodak et al., 2018) or on the sub-phrases in the SST-5 training set (Socher et al., 2013; Le and Mikolov, 2014), 2) the Performer architecture may not be ideal for modeling very short input sequences, as it was designed to approximate the Attention matrix in order to have favorable time and memory scaling for long input sequences (Choromanski et al., 2021), and 3) 56% of the words in the SST-5 training examples appear only once, which makes pre-training in the form of language models or word embeddings especially important for improving model performance.

Specificity: This dataset is relatively new, and to our knowledge, there has not been any additional published work using this dataset for benchmarking. In addition, it is difficult to compare directly with the authors' results, as they used continuous target values and trained a Support Vector Regression model (Gao et al., 2019).

For our use case of predicting NPS, we greatly value the accuracy of predicted survey ratings not only in absolute class agreement, but also in how closely the predicted ratings distribution matches the actual distribution. For accurate NPS prediction, our model rating distribution needs to reflect the actual distribution of NPS not just in aggregate, but across various business segments. To probe deeper into the performance differences between OR and MC, we examined the predicted distributions of Movie Ratings compared to ground truth.

Overall distributions for the MR and IMDb datasets (Figure 2) show: 1) OR produces more accurate rating distributions, as measured by smoothed K-L divergence (Table 1). 2) MC over-predicts majority classes in both datasets (2s and 3s for MR and 1s and 10s for IMDb) while under-predicting the others (except 2s and 3s in IMDb). These results are in line with the common observation that MC models tend to overfit on the majority classes in im-

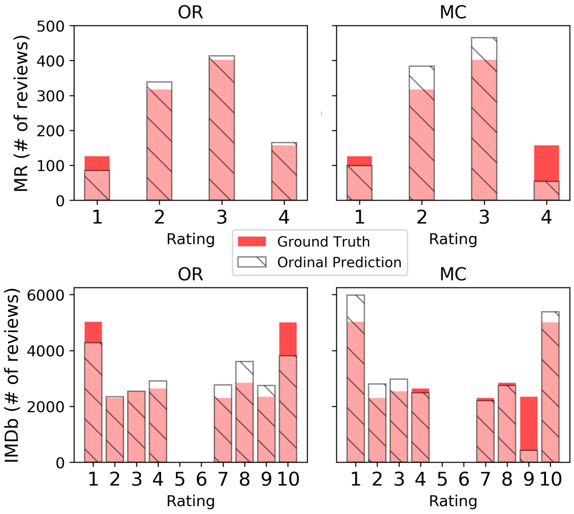

Figure 2: OR (left) outperforms MC (right) with respect to representing whole distributions of movie ratings for MR (top) and IMDb (bottom) datasets as measured by divergence.

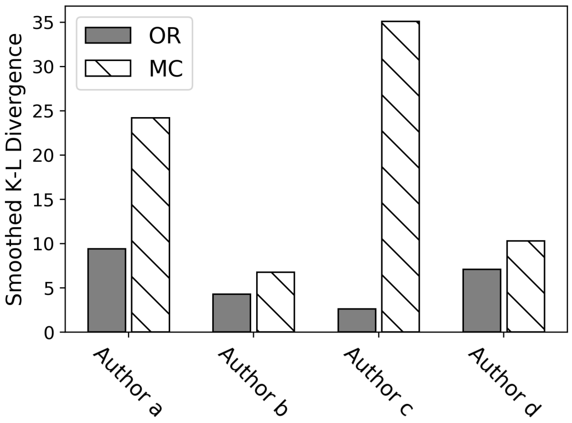

Figure 3: OR outperforms MC across authors in the MR dataset as measured by smoothed K-L divergence (lower is better). This is particularly evident in author c (10x) and author a (2x).

balanced datasets, which motivates the use of "oversampling" or class balancing(Buda et al., 2018; Chawla et al., 2002; Tepper et al., 2020; Gao et al., 2020). OR, in contrast, provides a better fit for MR (slightly under-predicting for 1s), but significantly under-predicts on IMDb majority classes, displaying a much flatter distribution of predictions. 3) Lastly, in MC, for the IMDb distribution, where there are a greater number of classes (8 ratings considered), we observe a tendency to "drop" or ignore a particular class resulting in significant under-prediction, for example the 9s in the bottom right of Figure 2. We observe this throughout training, and, depending on the epoch, have observed the model dropping other minority classes (2's, 3's, 6's, etc.), where we observe a recall less than 3%. This suggests that MC and OR lead to different behaviors with respect to predictive representation, as we discuss later.

To explore the performance of our model with respect to different data subsets, we calculate the smoothed K-L divergence for each of the four authors in the MR dataset (Figure 3). We find that OR greatly out-performs MC: while there are modest improvements in K-L divergence for authors b and d, we observe a two- to ten-fold increase for authors a and c, respectively. We hypothesize that MC is learning associations between review text and ratings as a whole by optimizing for overall accuracy in the predicted ratings, which may resemble learning how an amalgamation (or weighted average) of the 4 authors would rate a given movie. On the other hand, OR appears more sensitive to author-specific language, resulting in far lower K-L divergence values, but this may simply be due to the lower overall K-L divergence in OR calculated over the entire dataset. Lower overall and author-specific K-L divergence may be beneficial for applications with personalized predictions, i.e. in cases where performance on various data segments is important. Notably, authors a and c are the majority segments of the MR dataset, where each author a, b, c, and d has 1770, 902, 1307 and 1027 reviews, respectively. This may partially explain the improved K-L divergence of OR compared to MC for those segments, as there are more training examples for how authors a and c express their opinions on movie ratings.

Table 1 showed that raw accuracy is insufficient to distinguish better performance when distribution fit and error distance are important. In the SemEval-2016 Task for 5-point sentiment analysis of Twitter posts, some contributors used macro-averaged mean absolute error (MAE^M) (Rosenthal et al., 2017), which performs a macroaveraging of absolute errors between prediction and targets across all classes. This metric breaks down the error across target classes and can be better-suited for cases of class imbalance. While MAE^M, MSE, and correlation are all error-sensitive, they do not give

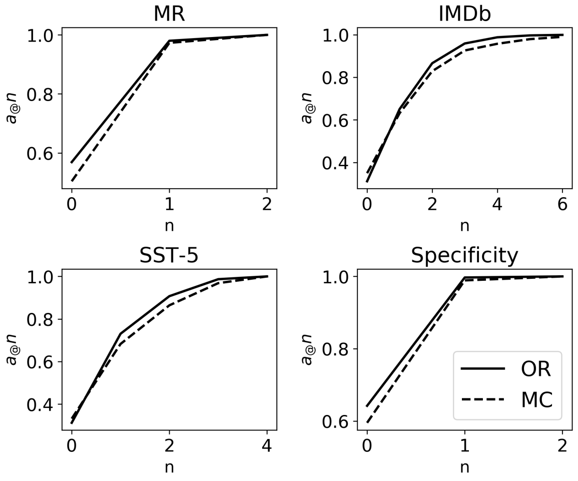

Figure 4: To visualize the impact of optimizing for ordinal rank, we define "accuracy at n" or $a_{@}n$, which considers the closeness of predictions to truth. $a_{@}n$ shows that for all benchmark datasets, OR outperforms MC for $n > 0$, where n is the absolute error.

fine-grained insights into the degree of errors, e.g. how close did we get to the right answer? To show this, we extend MAE^M and MSE with a metric that allows us to visualize degree of error: "accuracy at n" or $a_{@}n$ (Figure 4). $a_{@}n$ calculates the proportion of predictions that are within an allowable absolute error, n (ranging from 0 to $K-1$), from their targets, y. In this metric, $a_{@}0$ represents traditional accuracy. Accordingly, the graphs in Figure 4 begin at the values reported in Table 1 . However, as we increase n, we observe that across the board the $a_{@}n$ for $n > 0$ is higher for OR compared to MC for all datasets. In other words, while MC predicts the exact rating correctly more often than OR for IMDb and SST-5, when OR predicts incorrectly, it generally gets closer to the target than MC. This is highly desirable for tasks where the degree of error is important.

5 Discussion

We have shown that while MC outperformed OR for some datasets in terms of accuracy and F1, OR is significantly better than MC for minimizing error between predictions and targets for all datasets, as revealed by error-sensitive metrics such as mean-squared error and Spearman Rho. This can lead to better performance in terms of representing distributions, as measured by smoothed K-L divergence, and minimizing the magnitude of errors, as shown by $a_{@}n$.

We attribute OR's superiority over MC on MSE and Spearman correlation to the difference in their loss functions. While both MC and OR utilize cross-entropy, the label encodings and model constraints account for major differences. MC assumes that each rating is independent and uses one-hot encodings. OR encodes the ratings as rank thresholds, v, where each index, v_i, represents a binary indicator of threshold, i.e. $v_1 = 1$ if $\hat{y}^1 > 0.5$ and $v_2 = 1$ if $\hat{y}^2 > 0.5$, etc. This encoding combined with shared weight parameters and independent biases enforces rank-consistency, meaning a model prediction of $\hat{y}^3 > 0.5$ is accompanied by both $\hat{y}^2 > 0.5$ and $\hat{y}^1 > 0.5$ because $b_1 \geq b_2 \geq b_3$ (Cao et al., 2020). Consequently, we have not observed rank-inconsistent predictions in our OR models. This rank-consistency places a constraint on the model, forcing it to learn the ordinal information separating different ratings. In terms of bias-variance tradeoff, MC results in a lower bias, higher variance model, while OR produces a higher bias, lower variance model for ordinal target variables. Therefore, OR optimizes for rank-error between prediction and target, leading to lower MSE and higher Spearman correlations compared to MC.

Despite not directly optimizing for raw accuracy, we observe that OR still outperforms MC in accuracy and F1 for the MR and Specificity datasets. We hypothesize that this is related to the independence assumption that MC places during training, which may omit useful ordinal signal. While Likert-like scales can be highly subjective and inconsistent from one reviewer to another (Liang et al., 2020) due to cultural differences (Lee et al., 2002), among other reasons, there is significant overlap in rating schemes among reviewers (e.g. all reviewers should agree that higher ratings are better than lower). This standardization is particularly apparent when ratings are consistently generated, as is the case with MR, where all reviews originate from four authors. Each author, while unique, has a self-consistent way of expressing their reviews, which allows their readers to understand their reasoning for assigning movie ratings. This intra-author consistency may enhance the ordinal signal contained in the MR dataset. The Specificity dataset provides a related explanation. The ranks in the Specificity and MR datasets derive from labeled ratings that were collapsed from a more-continuous scale into 4 ranks. This may help to reduce the variance among the annotators. Additionally, the creators of the Specificity dataset took care to ensure that annotators agreed with one-another, assigning multiple annotators to label each data point (Gao et al., 2019). OR may also benefit from having fewer ranks, as more ranks create more opportunities for inconsistencies among reviewers as to what each rank represents.

The complement to our previous observation is that MC outperforms OR in accuracy and F1 for IMDb and SST-5. For IMDb, we hypothesize that this is related to the granularity of the ranks (8 total classes), and the large number of reviewers (likely tens of thousands). In this case, each person's different notions of ratings introduces considerable noise into the ordinal signal. It seems that OR may be more sensitive to rater inconsistencies compared to MC because OR fits the model on the ordinal signal whereas MC makes a rank-independence assumption. For SST-5, the answer is less clear, as each sentence is annotated by 3 judges, and the labels are collapsed down to 5-classes from a continuous range obtained from a sliding bar. It is possible that OR may struggle to make the generalizations necessary to successfully distinguish different ranks when the input sequences are short, as words may not appear in more than one example. For SST-5, 56% of the words in the training split appear in only one sentence. This may create difficulties in learning to generalize across ranks. It also highlights the impact of pre-training either in the form of language models or word embeddings for performance (Khodak et al., 2018). This pitfall is not apparent in the Specificity dataset, and we hypothesize that it is because the specificity task has significant correlations with the Tweet length itself (Gao et al., 2019), so that learning word associations is less important.

In addition to rank-error minimization, OR avoids the class dropping problem we observe in MC. We hypothesize that MC drops classes because probability mass is shared among all classes via the Softmax activation. This results in a model bias to take probability mass away from minority classes to give to majority classes when it is uncertain, which we observe in MC over-predicting majority classes (Figure 2). This is likely to happen in cases where the model cannot find reliable signal for particular minority class(es). In the extreme case, this leads to nearly no predictions for those minority classes, i.e. class dropping. OR, in contrast, avoids class dropping because its predictions do not share probability mass, i.e. each rank threshold represents its own prediction after Sigmoidal activation, without a Softmax. This builds inherent robustness into the prediction. For example, if the model is quite confident that a review is higher than 2 and less than 5, i.e. $\hat{y}^2 >> 0.5$ and $\hat{y}^5 << 0.5$, but unsure if it is a 3 or a 4, i.e. $\hat{y}^3 \approx 0.5$ and $\hat{y}^4 \approx 0.5$, rather than splitting probability mass between 3 and 4 as in MC (thus making other classes more likely in relation via Softmax), it can adjust the probability mass of v_3 independent of the other thresholds by adjusting b_3. For inputs near the decision boundary for both v_3 and v_4, the model will predict 3 and 4 in roughly equal proportions, avoiding a drop of either rank, whereas for MC, the model's bias towards majority classes coupled with Softmax activation may lead to dropping 3 or 4. Therefore, OR may produce better results in tasks with data imbalance, e.g. with highly-skewed or bi-/multi-modal distributions.

While our results are only empirically demonstrated for OR implemented with CORAL, we expect that our observations would likely generalize to other OR implementations, especially latent-variable models and other variants that produce rank-ordered thresholds, as the salient features would be the same.

6 Conclusion

Common model evaluation metrics such as accuracy, precision, recall, and F1 are insufficient for capturing the degree of error between prediction and target for multi-class prediction where the target is ordinal. Therefore, selecting models based on these traditional metrics may result in selecting an underperforming model. Error-sensitive metrics such as MSE (similarly, MAE and RMSE) and Spearman correlation capture the ordinal error, resulting in selecting models that have more representative distributions and improved generalization across data segments, as measured by K-L divergence. This is particularly important in use cases like predictive NPS, where accurate scores are necessary not just in aggregate, but across time and different customer segments as well. The independence assumption made by MC can remove useful ordinal signal, especially in cases where there is greater consistency across reviewers, their language, or their ratings. For benchmarks involving ordinal target variables, it is important to evaluate the MSE (or a similar error-sensitive metric like MAE, $a_{@}n$, and MAE^M) and Spearman correlations in addition to the usual metrics in determining whether a new model outperforms previous models. We hope to see these metrics included in future benchmarks with ordinal target variables.

Additionally, some of our experimental observations appear to be tied to the Softmax activation used in MC compared to the Sigmoidal activation in OR, such as class dropping in MC and substantially lower author-level K-L divergence in OR. These observations motivate exploration into alternative losses and activations for ordinal problems compared to traditional classification. For example, it would be interesting to compare the MC approach to one that does not involve the Softmax, such as "One-vs-Rest" or "One-vs-One," to observe whether class dropping persists.

References

I. Elaine Allen and Christopher A. Seaman. 2007. Likert scales and data analyses. *Quality Progress*.

Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. 2009. Evaluation measures for ordinal regression. In *Proceedings of the 2009 Ninth International Conference on Intelligent Systems Design and Applications*, pages 283–287, USA. IEEE Computer Society.

Ramy Baly, Georgi Karadzhov, Dimitar Alexandrov, James Glass, and Preslav Nakov. 2018. Predicting factuality of reporting and bias of news media sources. *arXiv preprint arXiv:1810.01765*.

Francesco Barbieri, Jose Camacho-Collados, Leonardo Neves, and Luis Espinosa-Anke. 2020. Tweeteval: Unified benchmark and comparative evaluation for tweet classification. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1644–1650. Association for Computational Linguistics.

Adrian Bickerstaffe and Ingrid Zukerman. 2010. A hierarchical classifier applied to multi-way sentiment detection. In *Proceedings of the 23rd International Computational Linguistics (Coling 2010)*, pages 62–70. Association for Computational Linguistics.

Mateusz Buda, Atsuto Maki, and Maciej A. Mazurowski. 2018. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*, pages 249–259.

Jose Camacho-Collados and Mohammad Taher Pilehvar. 2018. On the role of text preprocessing in neural network architectures: An evaluation study on text categorization and sentiment analysis. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 40–46. Association for Computational Linguistics.

M Karen Campbell and Allan Donner. 1989. Classification efficiency of multinomial logistic regression relative to ordinal logistic regression. *Journal of the American Statistical Association*, 84(406):587–591.

Wenzhi Cao, Vahid Mirjalili, and Sebastian Raschka. 2020. Rank consistent ordinal regression for neural networks with application to age estimation. *Pattern Recognition Letters*, 140:325–331.

Nitesh V. Chawla, Kevin W. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. 2002. Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357.

Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, David Belanger, Lucy Colwell, and Adrian Weller. 2021. Rethinking attention with performers. *ICLR*.

Jacob Devlin, Chang Ming-Wei, Lee Kenton, and Toutanova Kristina. 2019. 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)*, pages 41710–4186. Association for Computational Linguistics.

Eibe Frank and Mark Hall. 2001. A simple approach to ordinal classification. In *De Raedt L., Flach P. (eds) Machine Learning: ECML 2001.*, volume 2167. ECML 2001. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg.

Yang Gao, Yi-Fan Li, Yu Lin, Charu Aggarwal, and Latifur Khan. 2020. Setconv: A new approach for learning from imbalanced data. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1284–1294. Association for Computational Linguistics.

Yifan Gao, Yang Zhong, Daniel Preotiuc-Pietro, and Junyi Jessy Li. 2019. Predicting and analyzing language specificity in social media posts. *Association for the Advancement of Artificial Intelligence.*

T Graepel and K Obermayer. 1999. *Advances in Large Margin Classifiers*, volume 7, chapter Large margin rank boundaries for ordinal regression. The MIT Press.

Pedro Antonio Gutiérrez, María Pérez-Ortiz, Javier Sánchez-Monedero, Francisco Fernández-Navarro, and César Hervás-Martínez. 2016. Ordinal regression methods: survey and experimental study. *IEEE Transactions on Knowledge and Data Engineering*, 28(1):127–146.

Qingnan Jiang, Lei Chen, Ruifeng Xu, Xiang Ao, and Min Yang. 2019. A challenge dataset and effective models for aspect-based sentiment analysis. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6280–6285. Association for Computational Linguistics.

Ankur Joshi, Saket Kale, Satish Chandel, and D Kumar Pal. 2015. Likert scale: Explored and explained. *British Journal of Applied Science & Technology*, 7(4):396.

Phillip Keung, Yichao Lu, György Szarvas, and Noah A. Smith. 2020. The multilingual amazon reviews corpus. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4563–4568. Association for Computational Linguistics.

Mikhail Khodak, Nikunj Saunshi, Yingyu Liang, Tengyu Ma, Brandon Stewart, and Sanjeev Arora. 2018. A la carte embedding: Cheap but effective induction of semantic feature vectors. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12–22. Association for Computational Linguistics.

Thomas R Knapp. 1990. Treating ordinal scales as interval scales: an attempt to resolve the controversy. *Nursing research*, 39(2):121–123.

William Kuzon, Melanie Urbanchek, and Steven McCabe. 1996. The seven deadly sins of statistical analysis. *Annals of plastic surgery*, 37:265–272.

Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *Proceedings of the 31 st International Conference on Machine Learning*, 32. JMLR.

Jerry W. Lee, Patricia S. Jones, Yoshimitsu Mineyama, and Xinwei Esther Zhang. 2002. Cultural differences in responses to a likert scale. *Research in Nursing and Health*, 25(4):295–306.

Weixin Liang, James Zou, and Zhou Yu. 2020. Beyond user self-reported likert scale ratings: a comparison model for automatic dialog evaluation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1363–1374. Association for Computational Linguistics.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150. Association for Computational Linguistics.

Chester H McCall. 2001. An empirical examination of the likert scale: Some assumptions, development and cautions. In *annual meeting of the CERA Conference, South Lake Tahoe, CA.*

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 115–124. Association for Computational Linguistics.

Frederick F. Reichheld. 2003. The one number you need to grow. *Harvard Business Review.*

Filipe N. Ribeiro, Matheus Araújo, Pollyanna Gonçalves, Marcos André Gonçalves, and Fabrício Benevenuto. 2016. Sentibench - a benchmark comparison of state-of-the-practice sentiment analysis methods. *EPJ Data Science*, 5(23).

Anthony Rios and Ramakanth Kavuluru. 2017. Ordinal convolutional neural networks for predicting rdoc positive valence psychiatric symptom severity scores. *Journal of biomedical informatics*, 75:S85–S93.

Sara Rosenthal, Noura Farra, and Preslav Nakov. 2017. Semeval-2017 task 4: Sentiment analysis in twitter. In *Proceedings of the 11th international workshop on semantic evaluation (SemEval-2017)*, pages 502–518.

Sebastian Ruder. Nlp-progress: Sentiment analysis [online]. 2021.

Shihab Elbagir Saad and Jing Yang. 2019. Twitter sentiment analysis based on ordinal regression. *IEEE Access*, 7:163677–163685.

Parinaz Sobhani, Diana Inkpen, and Stan Matwin. 2015. From argumentation mining to stance classification. In *Proceedings of the 2nd Workshop on Argumentation Mining*, pages 67–77.

Richard Socher, Alex Perelygin, Jean Y. Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Conference on Empirical Methods in Natural Language Processing (EMNLP 2013)*, pages 1631–1642. Association for Computational Linguistics.

Naama Tepper, Esther Goldbraich, Naama Zwerdling, George Kour, Ateret Anaby Tavor, and Boaz Carmeli. 2020. Balancing via generation for multi-class text classification improvement. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1440–1452. Association for Computational Linguistics.

Sheng Zhang, Rachel Rudinger, Kevin Duh, and Benjamin Van Durme. 2017. Ordinal commonsense inference. *Transactions of the Association for Computational Linguistics*, 5:379–395.