Ranking-Constrained Learning with Rationales for Text Classification

Juanyan Wang
Illinois Institute of Technology
Chicago, IL USA
jwang245@hawk.iit.edu

Manali Sharma
Samsung Semiconductor, Inc
San Jose, CA USA
manali.s@samsung.com

Mustafa Bilgic
Illinois Institute of Technology
Chicago, IL USA
mbilgic@iit.edu

Abstract

We propose a novel approach that jointly utilizes the labels and elicited rationales for text classification to speed up the training of deep learning models with limited training data. We define and optimize a ranking-constrained loss function that combines cross-entropy loss with ranking losses as rationale constraints. We evaluate our proposed rationale-augmented learning approach on three human-annotated datasets, and show that our approach provides significant improvements over classification approaches that do not utilize rationales as well as other state-of-the-art rationale-augmented baselines.

1 Introduction

Text classification has been used for numerous applications including sentiment analysis (Hemmatian and Sohrabi, 2019), information retrieval (Aggarwal and Zhai, 2012), and language identification (Jauhiainen et al., 2019). When presented with a large number of labeled documents, common text classification models demonstrate impressive results. In practical settings, however, labeled data is often scarce. Labeling documents is a tedious task that requires time and effort, thus curating a large labeled corpus can be expensive and even unrealistic.

There is a wide range of use cases for businesses and industry that require curating a labeled dataset for the current task before the need to move on to the next task arises. For example, consider legal case document classification where documents need to be labeled as relevant/not-relevant to the current case at hand. The next legal case requires labeling the documents as relevant/not-relevant for that particular case, and so on. Similarly, several fast-response tasks such as immediate analysis of news and social media posts for a breaking news, for a recently released product, for a policy announcement, etc., require fast curation of a small and yet informative labeled dataset.

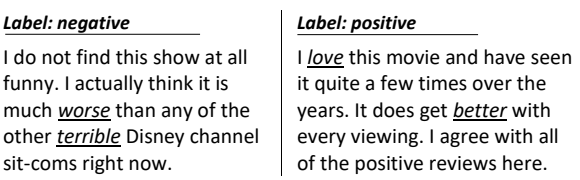

Figure 1: Rationales annotated on a negative movie review and a positive movie review.

An effective approach to make the best use of the human's time and maximize classifier performance with a small labeled dataset is to elicit rich feedback, in the form of rationales for classification, during the labeling process (Zaidan et al., 2007, 2008; Donahue and Grauman, 2011; Sharma and Bilgic, 2018). For sentiment classification, for example, the annotators might highlight certain segments of the text that convinced them to label the review as positive or negative (Figure 1). Unlike humans, a classifier will not know which segments of the document are responsible for its label during training, until it has been presented with many training samples. Since the human annotators read the document to decide its label in the first place, they have already spent the time to find the justifications for their labeling decision; hence, previous studies have shown that the extra time needed to highlight a piece of the text as a rationale for its label is not high and is often worth more (for improving the classifier) than spending that time to label an additional document. Zaidan et al. (2007) showed that rationale annotation has low overhead, roughly twice the time required for annotating only the labels. Sharma and Bilgic (2018) showed that annotating a *single* document with rationales can be worth as many as 20 documents that are simply annotated with labels.

Prior work on learning with rationales focused on one-hot encoding of the text in combination with logistic regression and support vector machines

Findings of the Association for Computational Linguistics: ACL 2022, pages 2034 - 2046

(Zaidan et al., 2007; Sharma and Bilgic, 2018), deep learning with multi-task learning (Melamud et al., 2019), and rationale-augmented attention-based models (Bahdanau et al., 2014), which still required a large set of labeled documents. We propose a general approach that is applicable to both one-hot encoding as well as deep learning embedding representations and that is highly effective under limited labeling settings.

The rationale supervision can be understood as an expectation that a document should have a higher probability of belonging to its class than the same document from which the rationale(s) are removed. Motivated by this intuition, we formulate a hybrid loss function to combine classification loss with ranking constraints for rationale supervision, which serves as an effective way of directing the model's focus to rationales during training. Our contributions in this paper include:

- We formulate a general and effective learning-with-rationales method for text classification.
- We study its empirical effectiveness on three human-annotated text classification datasets (sentiment analysis, aviation safety, and scientific articles).
- We compare our method to several baselines, and empirical findings show that it achieves the state-of-the-art results. For example, our proposed method is able to achieve 80% accuracy on the IMDb movie review dataset (Zaidan et al., 2007) with as few as 23 documents, whereas a fine-tuned BERT model that does not use rationales required 73 documents, and the most competitive rationale-augmented baseline required 63 documents to achieve the same level of accuracy.
- We annotate a new text classification dataset with rationales and make it publicly available.

The rest of the paper is organized as follows. We first discuss related work and how our work differs from previous work in Section 2. We formalize our learning with rationales approach in Section 3 and detail the experimental methodology in Section 4, followed by a discussion of the results in Section 5. We discuss the limitations and future work in Section 6 and then conclude.

2 Related Work

Zaidan et al. (2007) presented one of the first approaches to learning with rationales for text classification. They proposed to utilize human-provided rationales by converting the rationales into constraints for training support vector machines. They later extended the framework to a rationale-constrained probabilistic model (Zaidan and Eisner, 2008). Sharma and Bilgic (2018) proposed a general method to incorporate rationales into the training of any classifier by weighting the rationale features higher than the non-rationale features. However, their method relied on using a bag-of-words representation of the documents.

As deep learning achieved the state-of-the-art performance on text classification (e.g., (Sun et al., 2019; Devlin et al., 2019; Zhang et al., 2015; Yang et al., 2016)), recent work proposed methods specifically for training deep learning models using rationale supervision. Some methods utilized the rationales to generate rationale-augmented representations of the text while others utilized the rationales for richer supervision of the model. For instance, Zhang et al. (2016) proposed a Rationale-Augmented CNN (RA-CNN) that jointly learns from the labels of the documents as well as the labels at the sentence level, by using a two-step approach. However, their approach still requires sufficient amounts of data for training a model at the sentence level to learn a valid rationale-augmented representation of a document. Errica et al. (2021) proposed a representation learning approach to leverage rationales by learning to focus on relevant input tokens in the embedding space. Bao et al. (2018) proposed a framework to derive machine attentions from human-provided rationales. Sastry and Milios (2020) defined a new attribution score for words by computing the partial derivative of the output with respect to the input in the word embedding space, and used misattribution error as an additional supervision in the loss function. Our method has two major differences from these work: i) our approach can use but does not require an attention mechanism to focus on the rationales and ii) our approach does not require learning a separate representation for the rationales.

The work most closely related to ours is the model proposed by Melamud et al. (2019), which jointly learns to predict the labels for text as well as the labels for each token of every input sentence by determining whether the token is part of the rationales or not. Our approach differs from theirs as our ranking loss is calculated by using only the model's predictions, rather than introducing auxiliary learning tasks. Moreover, the approach we propose is more general: it can be used for any

model that can utilize a logistic loss, ranging from a logistic regression model coupled with a one-hot encoding of words to a Long Short-Term Memory (LSTM) model coupled with word embeddings. In their same paper, Melamud et al. (2019) proposed another method that utilizes rationales by constructing rationale prototypes and rationale-biased text vectors. However, these vectors are computed using a rationale-bias function to directly estimate the similarity between words and annotated rationales without incorporating any learning, and thus this method works well only for few-shot learning.

3 Learning with Rationales

Let $\mathcal{D} = \{x_1, x_2, \cdots, x_n\}$ be a set of documents. A small subset of the documents, $\mathcal{L} \subset \mathcal{D}$, are annotated with labels, $\langle x_i, y_i \rangle$ where the value of y_i belongs to a label space, $\mathcal{C} = \{c_1, c_2, \cdots, c_k\}$. y_i is unknown for a much larger set of unlabeled documents, $\mathcal{U} = \mathcal{D} \setminus \mathcal{L}$, represented as $\langle x_i, ? \rangle$. Each document, x_i, contains a number of sentences, $\{s_{i1}, s_{i2}, ..., s_{im}\}$, each of which is represented as a sequence of words: $s_{ij} = \{q^1_{ij}, q^2_{ij}, \cdots, q^l_{ij}\}$.

In the learning with rationales framework, a subset of the words is marked by the human annotator as rationales (i.e., justifications for the document's assigned label). Let $r_i = \bigcup q^l_{ij}$ be the set of all words that are marked as rationales within a document, x_i. It is possible that none of the words are marked as rationales, and hence, $r_i = \emptyset$ for such documents. In the learning-with-rationales setting, $\mathcal{L}$ is modified to contain $\langle x_i, r_i, y_i \rangle$ and $\mathcal{U}$ represents $\langle x_i, \emptyset, ? \rangle$. The objective is to train a model, f, that utilizes the documents x_i, their labels y_i, and their rationales r_i during training, and uses only the documents x_i at prediction time, as rationales are naturally not available for the test documents.

3.1 Our Approach – *LwR-RC*

We first describe our proposed approach, *Learning with Rationales – Ranking-Constrained (LwR–RC)*, and then illustrate how it can be specialized for training deep learning models. To illustrate the motivation behind our approach, consider an example document, *D*, that contains three sentences: *"s1: The movie came out last year. s2: The plot was decent. s3: Acting was superb."*, which is labeled as 'positive' by the annotator. Assume for the sake of example, the annotator highlights only *s3* as the rationale. Let *M* be a masked document that is same as the original document *D*, but from which the sentences containing the rationale phrases are removed. In this case, *M* would be missing *s3*. We postulate that the model should be more sure about the positive label of document *D* than the label of document *M*, since *D* contains the essential evidence, '*Acting was superb*', for the 'positive' label, whereas *M* lacks that evidence. Similarly, let *R* be the document that contains only the rationale sentence *s3*. We postulate that the model should be more sure about the label 'positive' of *R* than the label of *M*, since *R* provides strong evidence for the label, whereas *M* lacks that evidence.[1]

Traditional learning without rationales approaches optimize a loss function to compute the model's error on its predictions, e.g., a binary cross-entropy classification loss, L_{clf}, is defined as:

$$L_{clf} = -\frac{1}{|\mathcal{L}|} \sum_i (y_i \cdot log(p(y_i|x_i)) + (1 - y_i) \cdot log(1 - p(y_i|x_i))) \quad (1)$$

In order to leverage the annotated rationales, we formalize our postulations by providing the model with two additional objectives during training. The first objective is to train the model to be more confident about the label of a document (*D*) than the label of the same document in which the rationales are masked (*M*). The second objective is to train the model to be more confident about the label of document that contains only the rationales (*R*) than the label of the same document in which the rationales are masked (*M*). We achieve these objectives by using a ranking-constrained classification approach, as described next.

Let $\langle x_i, r_i, y_i \rangle \in \mathcal{L}$ be a training document. First, we construct an artificial document x'_i by masking out all the sentences that contain rationales r_i. We construct another artificial document x^r_i consisting of only the sentences that contain rationales r_i. The ranking-constrained classification approach incorporates the rationales into learning by modeling two expectations: (i) the model should be more sure of assigning the correct label y_i to x_i than assigning y_i to x'_i, because x'_i represents a document from which the rationales have been removed, and we refer to this objective as 'Document versus Masked document' (*DvM*), where *D* represents x_i and *M* represents x'_i, and (ii) the model should be more sure of assigning the correct label y_i to x^r_i than assigning y_i to x'_i, and we refer to this objec-

[1] It is possible that the annotator might pick both *s2* and *s3* as rationales; the same arguments that *D* and *R* should be more positive than *M* still applies.

tive as 'Rationale versus Masked document' (*RvM*), where *R* represents x_i^r and *M* represents x_i'.

Another possible objective can be 'Rationale versus Document' (*RvD*), however, we excluded *RvD* objective from our approach for the following reason. Consider the following cases for a binary (positive/negative) classification task:

- Case 1: *D* = *R*+*M* is positive; *R* is positive; *M* is neutral or it contains a small amount of leftover positive. In this case, *RvD* requires $R > R+M$, which forces *M* to be negative, whereas *RvM* requires $R > M$, which does not necessarily require *M* to be negative. Thus, *RvD* is guaranteed to be the wrong approach. *RvM* forces $R > M$, but gives the model the flexibility to decide whether *M* is a small positive, neutral, or negative.
- Case 2: *D* = *R*+*M* is positive; *R* is positive; *M* is negative. In this case, *RvD* requires $R > R+M$, which forces *M* to be negative, whereas *RvM* simply requires $R > M$. In this case, *RvD* is the correct choice, but *RvM* cannot be called the guaranteed wrong choice.
- Remaining cases: The cases where *D* and *R* are negative are similar.

As the cases above show, *RvM* is more flexible: *RvM* simply nudges the model in the correct direction and leaves the judgement about *M* to the data. *RvD*, on the other hand, is a more forceful approach; it forces the model to always make a judgement about *M*, which is the incorrect judgement in case 1. Thus, we include only the *RvM* and *DvM* objectives in our proposed approach.

Formally, let $y_i \in \{0, 1\}$: $f(x_i) = p(y_i = 1 \mid x_i) = sigmoid(W_z z_i)$ for some parameter matrix W_z, where z_i is the vector representation of x_i. For modeling the *DvM* objective, let $\mu_i = W_z z_i$ and $\mu_i' = W_z z_i'$ where z_i' is the vector representation of x_i'. If the correct label is $y_i = 1$, we would like $\mu_i > 0$ and $\mu_i > \mu_i'$. If the correct label is $y_i = 0$, we would like $\mu_i < 0$ and $\mu_i < \mu_i'$. We convert this constraint into a logistic loss, as follows:

$$L_{DvM}^i = \begin{cases} log(1 + exp(-(\mu_i - \mu_i'))), y_i = 1 \\ log(1 + exp(-(\mu_i' - \mu_i))), y_i = 0 \end{cases} \quad (2)$$

Summing L_{DvM}^i over all the training instances and reorganizing the terms, we get:

$$L_{DvM} = -\frac{1}{|\mathcal{L}|} \sum_i (y_i \cdot log(p(y_i|x_i, x_i')) + (1 - y_i) \cdot log(1 - p(y_i|x_i, x_i'))) \quad (3)$$

where,

$$p(y_i|x_i, x_i') = \frac{1}{1 + e^{-(\mu_i - \mu_i')}} \quad (4)$$

We define the ranking loss similarly for the *RvM* component, using documents *R* and *M* and their respective scores $\mu_i^r = W_z z_i^r$ and $\mu_i' = W_z z_i'$, where z_i^r is the vector representation of x_i^r. The ranking loss L_{RvM} is then defined as:

$$L_{RvM} = -\frac{1}{|\mathcal{L}|} \sum_i (y_i \cdot log(p(y_i|x_i^r, x_i')) + (1 - y_i) \cdot log(1 - p(y_i|x_i^r, x_i'))) \quad (5)$$

where,

$$p(y_i|x_i^r, x_i') = \frac{1}{1 + e^{-(\mu_i^r - \mu_i')}} \quad (6)$$

We combine the classification loss L_{clf} with the ranking losses, L_{DvM} and L_{RvM}, resulting in the main objective function for our approach:

$$L = (1 - \lambda_1 - \lambda_2) L_{clf} + \lambda_1 L_{DvM} + \lambda_2 L_{RvM} \quad (7)$$

where, $0 \leq \lambda_1 \leq 1$, $0 \leq \lambda_2 \leq 1$, and $\lambda_1 + \lambda_2 \leq 1$. λ_1 and λ_2 are two hyper-parameters that control the importance of the classification loss and the ranking losses relative to one another. We study the effect of these hyper-parameters on the model's performance and provide insights into their relative importance in Section 5.2. We next describe how *LwR-RC* can be implemented through a neural network architecture, which can be specialized to a logistic regression or to a deep learning model.

3.1.1 *LwR-RC* with Deep Learning

Figure 2 shows the deep learning architecture illustrating how the *LwR-RC* approach can minimize the loss function of Equation (7). For every sentence $\{s_{i1}, s_{i2}, ..., s_{im}\}$ within a document x_i, we use an embedding model to create sentence embedding vectors $\{t_{i1}, t_{i2}, ..., t_{im}\}$, and pass them through an average pooling layer to create a single vector, z_i, representing a document. Similarly, the same sentence embedding vectors are passed through two different pooling layers to create two masked averages, z_i' and z_i^r, representing the document without rationales and the document containing only the rationales, respectively. There are several strategies for aggregating many sentence vectors into a single document vector; we use the average pooling strategy for the experiments.

The *LwR-RC* approach can be used to train any model that uses cross-entropy loss functions, including logistic regression and deep neural networks. It can also work with several representations, including one-hot encoding of the words, word2vec (Mikolov et al., 2013), and doc2vec (Le

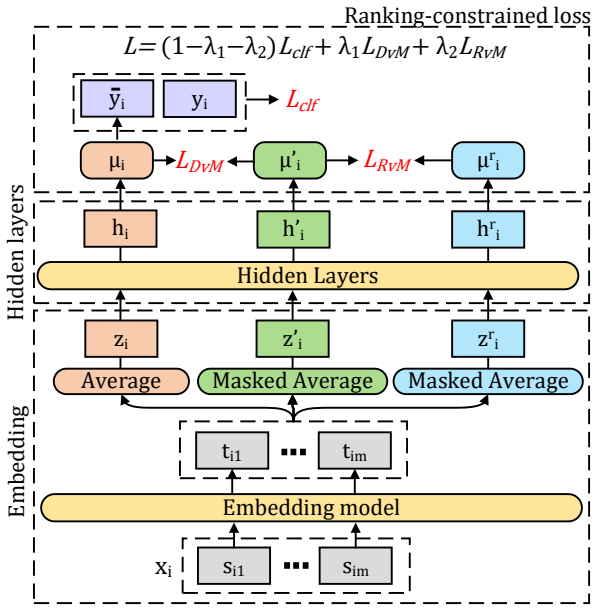

Figure 2: Architecture of the *LwR-RC* model for deep learning using one input document, x_i, as an example.

and Mikolov, 2014), as well as more recent language models such as BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019), and XLNet (Yang et al., 2019). For example, if we remove the embedding layer and the hidden layers, and represent the sentences using one-hot encoding of the words, we would get a simple logistic regression classifier. If we use BERT for encoding the sentences in the embedding layer, then we can either use BERT embeddings directly or fine-tune the BERT model on downstream classification tasks by optimizing the ranking-constrained loss function.

4 Experimental Setup

In this section, we describe the three datasets, several baselines, and the experimental settings.

4.1 Datasets

We used two publicly available datasets: a sentiment classification dataset and an aviation safety dataset. Both datasets were annotated with labels and rationales. Additionally, we introduce a new scientific article classification dataset that we annotated with labels and rationales.

IMDb is a movie review dataset annotated by Zaidan et al. (2007). It consists of 1,800 documents. We used 600 reviews as the training set, 600 reviews as the validation set, and 600 reviews as the test set.

ASRS is an Aviation Safety Reporting System dataset. We used the same balanced binary classification dataset created by Melamud et al. (2019), consisting of reports labeled with either 'Proficiency' or 'Physical Environment.' The original split had 386 documents for training and 392 documents for testing. We split the test set into two and use 196 documents for validation set and 196 documents for test set.

AIvsCR contains scientific articles that we collected from arXiv and annotated with rationales. This dataset contains 2,394 documents from Artificial Intelligence (cs.AI) and Cryptography and Security (cs.CR) categories. Two annotators independently annotated 394 documents with rationales for the ground truth label, and we computed the inter-annotator agreement for the rationales in the same manner as Zaidan et al. (2007). We used 394 human-annotated documents as the training set, 1,000 documents as the validation set, and 1,000 documents as the test set. Note that the validation and test sets do not need rationales; they only need the documents and their labels for evaluation. We make this dataset publicly available, and provide a complete description of this dataset in the appendix.

4.2 Experimental Settings

For training *LwR-RC*, we fine-tuned a pre-trained 'bert-base-uncased' version of the BERT (Devlin et al., 2019) model on downstream classification task using our ranking-constrained loss function. We used a TensorFlow implementation of BERT[2]. We input each sentence within a document to BERT and used the '[CLS]' logits from the last hidden layer as the sentence embeddings. To fit the model into GPU (NVIDIA Quadro RTX 5000) memory, we truncated each input sentence to at most 48 tokens (including two special tokens '[CLS]' and '[SEP]'), and each document to at most 64 sentences. We used only one hidden layer with 100 nodes in the hidden layers section of Figure 2, and used `tanh` as the activation function. The total number of model parameters for *LwR-RC* is 109,559,241. The running time of training *LwR-RC* is similar to training a fine-tuned BERT model without using rationales; *LwR-RC* needs to make two more forward passes to compute μ'_i and μ^r_i for x'_i and x^r_i, respectively.

We present average learning curves over 5 different runs to assess how the models would per-

[2] https://tfhub.dev/tensorflow/bert_en_uncased_L-12_H-768_A-12/3

form under varying labeling regiments, and plot error bars showing the standard error. Each learning curve starts with a bootstrap of 5 randomly selected documents from each label. Each step of the learning curve corresponds to labeling 20 additional documents. For a fair comparison between various learning strategies, all learning strategies (our approach and the baselines) are fed the same sequence of documents. After the bootstrap phase, we run 10 more steps, and hence the budget of learning curves runs up to $10 + 20 \times 10 = 210$ documents.

Tuning Hyper-parameters. For a fair comparison between our method and the baselines, at each iteration of learning, we performed grid search to optimize the tunable hyper-parameters of each method using the held-out validation set. For *LwR-RC*, we experimented with different pairs of hyper-parameters, λ_1 and λ_2, whose values were selected from the set {0, 0.125, 0.25, 0.5}. We fine-tuned BERT model for *LwR-RC* for 10 epochs, and selected the best model across different epochs using the held-out validation set. We next discuss the details of the baselines.

4.3 Baselines

We compare our approach with one *Learning without Rationales* (*Lw/oR*) baseline and four *Learning with Rationales* (*LwR*) baselines.

Learning without Rationales. The ***Lw/oR-BERT*** baseline fine-tunes the BERT model for downstream classification tasks, and optimizes the model by only minimizing the classification loss function, L_{clf}, without utilizing any ranking constraints, L_{DvM} or L_{RvM}, according to Equation (1). It is worth noting that traditional *Lw/oR* approaches that fine-tune BERT model on classification tasks have shown impressive performances, and therefore, *Lw/oR-BERT* is a strong baseline. For example, Sun et al. (2019) achieved the state-of-the-art performances on eight text classification tasks by fine-tuning the BERT model, outperforming both CNN and LSTM based models as well as using just pre-trained BERT embeddings. We observed similar trends in our experiments.

Learning with Rationales Baselines. We conducted experiments using four learning-with-rationales baselines from the literature.

1) ***Rationale-Augmented SVM (RA-SVM)***: This approach is Zaidan et al. (2007)'s model that translates the importance of rationales into additional constraints for training support vector machines. This method requires three hyper-parameters: regularization C for the original samples, regularization $C_{contrast}$ for the contrast samples, and margin μ between the original and contrast samples. We optimized these hyper-parameters using grid search, and selected the values of both C and $C_{contrast}$ from the set {0.01, 0.1, 1, 10, 100} and the value of μ from the set {0.01, 0.1, 1, 10}.

2) ***Rationale-Augmented LR (RA-LR)***: This approach is Sharma and Bilgic (2018)'s approach that emphasizes the rationales and de-emphasizes non-rationales in the vectorized feature matrix representation of the documents. It has three hyper-parameters, weight r for the rationale terms, weight o for the non-rationale terms, and regularization C. We selected the value of r from the set {1, 10, 100}, the value of o from the set {0.01, 0.1, 1}, and the value of C from the set {0.01, 0.1, 1, 10, 100} to optimize the hyper-parameters using grid search.

3) ***RB-BOW-PROTO*** and 4) ***RB-WAVG-BERT***: These are two models proposed by Melamud et al. (2019) that achieved the state-of-the-art performance in their experiments compared to Rationale-Augmented CNN (Zhang et al., 2016), Rationale-Augmented SVM (Sharma and Bilgic, 2018), and ULMFiT (Howard and Ruder, 2018). *RB-BOW-PROTO* uses a pre-trained word2vec embedding to construct rationale-biased text vectors for each class as prototypes, and then uses nearest-neighbor classification, instead of training a model to fine-tune the embeddings. This method has one hyper-parameter, α, that controls the impact of rationale biases on the rationale-bias function. We selected the value of α from the set {1, 3, 6, 12} to optimize it using grid search. The second approach, *RB-WAVG-BERT*, which is a strong baseline more closely related to our work, fine-tunes BERT model to jointly learn the labels on documents and the labels on tokens. We fine-tuned this model for 10 epochs and selected the best model across different epochs, using the learning rate of 5e-6, as suggested by the paper. Melamud et al. (2019) found that *RB-BOW-PROTO* performed better under extremely-limited labeling settings, and that *RB-WAVG-BERT* performed better when the training size was larger; hence, we included both approaches as baselines.

5 Results

We first present results comparing *LwR-RC* with the baselines, and then discuss the effects of the

two ranking-constrained losses on the performance of *LwR-RC*.

5.1 Comparison with the Baselines

Figure 3 presents learning curves comparing the average accuracy of the methods over five different runs with up to 210 documents for improved readability. The learning curves with a larger budget of up to 310 documents are included in the appendix.

BERT vs. *LwR* without BERT. The *Lw/oR-BERT* baseline that did not use rationales but fine-tuned BERT outperforms on the IMDb and AIvsCR datasets the two *LwR* frameworks (*RA-SVM* and *RA-LR*) that used rationales but did not use BERT embeddings. Zaidan et al. (2007) and Sharma and Bilgic (2018) showed that *RA-SVM* and *RA-LR* outperformed several *Lw/oR* approaches, and hence these two are strong *LwR* baselines. Still, a fine-tuned BERT model that does not use rationales is able to outperform these two strong baselines that used rationales but did not utilize the BERT embeddings. This result highlights the added benefit of the "existing knowledge" that pretrained embeddings provide.

BERT Baselines. *RB-WAVG-BERT*, the baseline that fine-tuned BERT model and utilized rationales, outperforms *Lw/oR-BERT*, the baseline that did not use rationales, showing the benefits of utilizing rationales with recent deep learning models. However, the improvements provided by *RB-WAVG-BERT* become noticeable only after the model has seen enough data (e.g., more than 50 documents), which was also noted by Melamud et al. (2019).

***LwR-RC* vs. the Best Baseline.** We next turn our attention to a fairer comparison: *LwR-RC* versus *RB-WAVG-BERT*; both used and fine-tuned BERT embeddings and both utilized rationales. *LwR-RC* provides statistically significant improvements[3] over *RB-WAVG-BERT*, with a p–value of less than 0.05, especially when the annotation budget is small, and it performs comparably at larger budgets. For IMDb, *LwR-RC* provides up to 22.3% improvements in accuracy over *RB-WAVG-BERT*; for ASRS, *LwR-RC* provides up to 21.7% improvements in accuracy over *RB-WAVG-BERT*. For AIvsCR dataset, *Lw/oR-BERT* can quickly reach 90% accuracy even without utilizing rationales, and thus the improvements provided by *LwR-RC* on this dataset for most training budgets are not as large as the improvements on the other two datasets; however, *LwR-RC* can still provide up to 8.67% improvements in accuracy over *RB-WAVG-BERT*. Regarding *RB-BOW-PROTO*, as Melamud et al. (2019) also observed, it performs well only under extremely-limited budget settings.

[3]The complete t-test results are presented in the appendix.

Dataset	Method	Target Accuracy (%)					
		65	**70**	**75**	**80**	**85**	**90**
IMDb	*Lw/oR-BERT*	14	36	52	73	148	N/A
	RB-WAVG-BERT	9	32	43	63	97	208
	LwR-RC	5	9	15	23	36	220
ASRS	*Lw/oR-BERT*	43	69	N/A	N/A	N/A	N/A
	RB-WAVG-BERT	36	57	87	192	N/A	N/A
	LwR-RC	12	19	27	44	90	N/A
AIvsCR	*Lw/oR-BERT*	5	7	8	10	28	93
	RB-WAVG-BERT	4	6	8	10	28	73
	LwR-RC	2	3	5	8	13	29

Table 1: Comparison between the number of annotated documents needed to achieve a target accuracy by the three methods. 'N/A' represents that a target accuracy could not be achieved by a method even with 310 training documents.

Corresponding to the learning curves presented in Figure 3, Table 1 shows the number of annotated documents needed for training *LwR-RC* as well as the two fine-tuned BERT baselines, *Lw/oR-BERT* and *RB-WAVG-BERT*, to achieve a target accuracy (ranging from 65% to 90%). As Table 1 shows, *LwR-RC* usually needs 2 and sometimes 3 times fewer number of annotated documents compared to *Lw/oR-BERT* and *RB-WAVG-BERT* to achieve the same level of accuracy.

5.2 The Effects of the Loss Functions

We further investigate the effects of the two ranking-constrained losses. Specifically, we want to understand how *LwR-RC* behaves with the two ranking-constrained losses: *LwR-RC*$_{DvM}$ that uses only L_{DvM} (setting λ_1 to 0.25 and λ_2 to 0 in Equation (7)), and *LwR-RC*$_{RvM}$ that uses only L_{RvM} (setting λ_1 to 0 and λ_2 to 0.25 in Equation (7)). Figure 4 presents the learning curves for these settings. For the IMDb dataset, *LwR-RC*$_{RvM}$ achieves a slightly higher accuracy than *LwR-RC*$_{DvM}$ after 100 training documents. For ASRS dataset, *LwR-RC*$_{DvM}$ performs the best, and for AIvsCR dataset, *LwR-RC*$_{RvM}$ performs the best.

To investigate it further, we provide average statistics for the number of sentences, the number of rationale sentences, and the percentage of rationale sentences within the documents for each dataset in Table 2. We observe that *LwR-RC*$_{RvM}$ performs better when the percentage of rationale sentences in documents is high, e.g., IMDb and

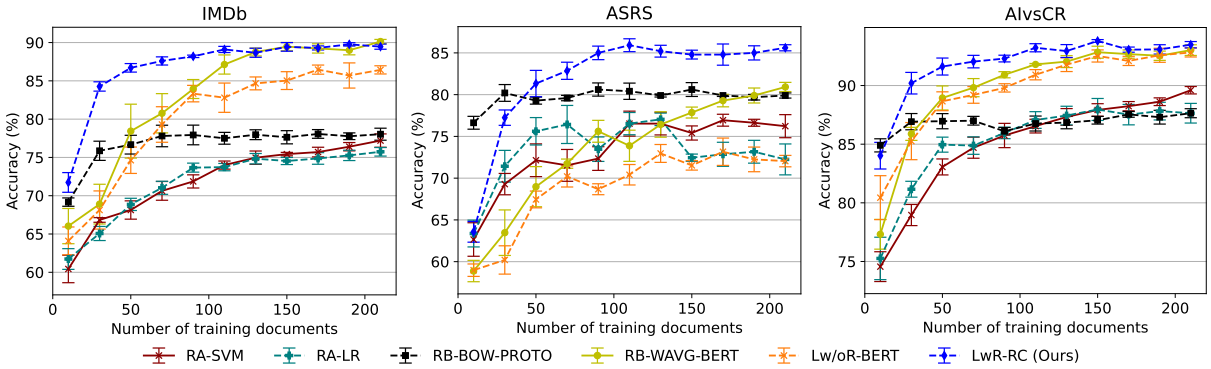

Figure 3: Comparison between our approach, *LwR-RC*, and the five baselines using the best hyper-parameter setting for each method.

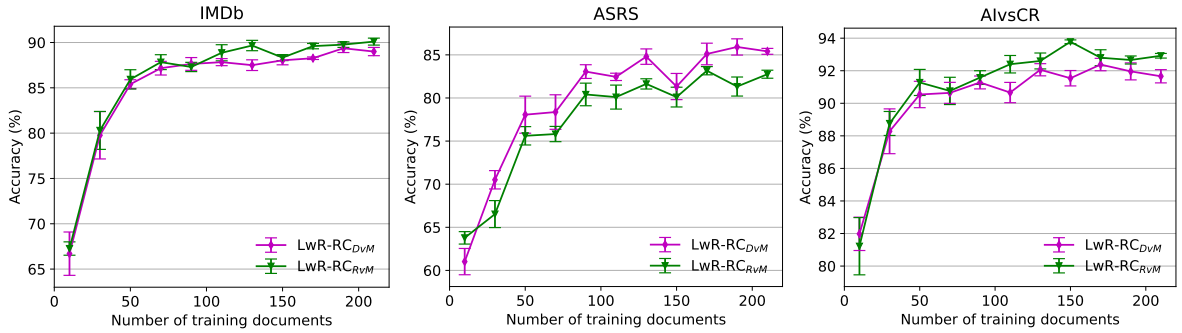

Figure 4: Comparison between different ranking constrained losses for *LwR-RC*. *LwR-RC*$_{DvM}$ represents using the parameter setting (λ_1=0.25, λ_2=0), and *LwR-RC*$_{RvM}$ represents using the parameter setting (λ_1=0, λ_2=0.25) in Equation (7).

Average Statistics	**IMDb**	**ASRS**	**AIvsCR**
# Sentences	33.7	15.3	8.5
# Rationale sentences	7.9	2.4	2.5
% Rationale sentences	25.7	19.0	30.5

Table 2: Average statistics per document for the IMDb, ASRS, and AIvsCR datasets. The percentages in the third row are computed by taking an average of the percentages of rationale sentences for all documents within each dataset, instead of dividing the values in the first row by the values in the second row directly.

AIvsCR datasets, and *LwR-RC*$_{DvM}$ performs better when the percentage of rationale sentences is low in the documents, e.g., ASRS dataset.

We hypothesize that different ranking constraints may be affected differently by a number of factors, including the budget for training documents, the diversity of rationales, the number of rationales provided for each document, how thorough the annotator was in providing rationales, and the domain, to name a few. Table 2 provides only a glimpse of such a study. An exhaustive study is needed for making a definitive conclusion about how various document and rationale statistics affect different ranking-constrained losses, which is beyond the scope of this study. However, the tuning strategy that picks the best λ parameters for *LwR-RC* at each iteration of learning using a validation set, and hence chooses the appropriate balance between the two loss functions, works well in practice, as was shown in Figure 3.

6 Limitations and Future Work

We presented experimental results for binary classification tasks in this paper. To the best of our knowledge, prior learning-with-rationales frameworks also focused on binary classification tasks in their experiments. Extending the framework to multi-class settings is a promising future direction. Such an extension would require adapting the loss functions to multi-class settings and creating multi-class classification datasets with rationales. Extending the framework to *multi-label* settings where a document can be assigned more than one label, however, is more challenging, both for formulating the problem as well as annotating the datasets with rationales, because rationales need to be assigned to their respective labels, which might be more than one in a single document.

7 Conclusions

We presented a novel approach to incorporate rationales as ranking-constraints into the training of classification models with cross-entropy loss. The proposed approach is general enough that it can be used for simple models, such as logistic regression with one-hot encoding of documents, as well as deep learning models combined with text embeddings. We conducted empirical evaluations comparing the proposed approach to several baselines and observed that the proposed approach outperformed the baselines in most settings, and was comparable to them at the remaining settings.

Acknowledgments

This material is based upon work supported by Samsung Semiconductor Inc. under a grant titled "Interactive Patent and Scientific Article Classification."

References

Charu C Aggarwal and ChengXiang Zhai. 2012. A survey of text classification algorithms. In *Mining text data*, pages 163–222. Springer.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Yujia Bao, Shiyu Chang, Mo Yu, and Regina Barzilay. 2018. Deriving machine attention from human rationales. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1903–1913, Brussels, Belgium. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Jeff Donahue and Kristen Grauman. 2011. Annotator rationales for visual recognition. In *2011 International Conference on Computer Vision*, pages 1395–1402. IEEE.

Federico Errica, Fabrizio Silvestri, Bora Edizel, Ludovic Denoyer, Fabio Petroni, Vassilis Plachouras, and Sebastian Riedel. 2021. Concept matching for low-resource classification. In *2021 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE.

Fatemeh Hemmatian and Mohammad Karim Sohrabi. 2019. A survey on classification techniques for opinion mining and sentiment analysis. *Artificial Intelligence Review*, pages 1–51.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339, Melbourne, Australia. Association for Computational Linguistics.

Tommi Jauhiainen, Marco Lui, Marcos Zampieri, Timothy Baldwin, and Krister Lindén. 2019. Automatic language identification in texts: A survey. *Journal of Artificial Intelligence Research*, 65:675–782.

Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *International conference on machine learning*, pages 1188–1196. PMLR.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Oren Melamud, Mihaela Bornea, and Ken Barker. 2019. Combining unsupervised pre-training and annotator rationales to improve low-shot text classification. 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 3884–3893, Hong Kong, China. Association for Computational Linguistics.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Chandramouli Shama Sastry and Evangelos E Milios. 2020. Active neural learners for text with dual supervision. *Neural Computing and Applications*, pages 1–20.

Manali Sharma and Mustafa Bilgic. 2018. Learning with rationales for document classification. *Machine Learning*, 107(5):797–824.

Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019. How to fine-tune bert for text classification? In *Chinese Computational Linguistics*, pages 194–206, Cham. Springer International Publishing.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489, San Diego, California. Association for Computational Linguistics.

Omar Zaidan and Jason Eisner. 2008. Modeling annotators: A generative approach to learning from annotator rationales. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 31–40, Honolulu, Hawaii. Association for Computational Linguistics.

Omar Zaidan, Jason Eisner, and Christine Piatko. 2007. Using "annotator rationales" to improve machine learning for text categorization. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 260–267, Rochester, New York. Association for Computational Linguistics.

Omar F Zaidan, Jason Eisner, and Christine Piatko. 2008. Machine learning with annotator rationales to reduce annotation cost. In *Proceedings of the NIPS* 2008 workshop on cost sensitive learning*, pages 260–267.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in neural information processing systems*, pages 649–657.

Ye Zhang, Iain Marshall, and Byron C. Wallace. 2016. Rationale-augmented convolutional neural networks for text classification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 795–804, Austin, Texas. Association for Computational Linguistics.

A Appendix

In this section, we supplement the results presented in the paper with the following:

- In the paper, we focused on experimental results with a budget of up to 210 training documents. Here, we supplement the main results in the paper with a larger budget of up to 310 documents.
- We present the improvements in accuracy provided by *LwR-RC* over the two fine-tuned BERT baselines, *Lw/oR-BERT* and *RB-WAVG-BERT*, for all three datasets at varying budgets.
- We provide the results of paired t-tests comparing *LwR-RC* to *Lw/oR-BERT* and *RB-WAVG-BERT*.
- In the paper, we provided the formulation of *LwR-RC* for binary classification for the ease of exposition. Here, we extend the formulation of *LwR-RC* to multi-class classification.
- We provide a complete description of the AIvsCR dataset that we collected and annotated with rationales for the ground truth labels.
- Additionally, we provide the AIvsCR dataset and the other two datasets (IMDb and ASRS), as well as the source code for all the experiments in our paper with this submission as separate .zip files.

B Results with Larger Budgets

In the paper, we focused on experimental results with a budget of up to 210 training documents (Figure 3). We supplement the results in Figure 3 with a larger budget of up to 310 training documents in Figure 5. As can be seen in Figure 5, the trends of all the results in the paper remain the same even with larger budgets. For IMDb and AIvsCR datasets, *LwR-RC* still performs better or comparably to the most competitive baseline, *RB-WAVG-BERT*; for ASRS dataset, *LwR-RC* still outperforms all the baselines. However, as the number of labeled documents grows, we expect our models and the baselines to converge to a similar accuracy, as the models no longer need the human-provided rationales and can learn statistically "what is important" from a large collection of documents that are simply annotated with labels.

C Accuracy Improvements

We present the improvements in accuracy provided by *LwR-RC* compared to the baselines for the three datasets across different training budgets. Specifically, we compare *LwR-RC* with the two fine-tuned BERT based approaches, *Lw/oR-BERT* and *RB-WAVG-BERT*. As shown in Table 3, *LwR-RC* provides significant improvements in accuracy over the two baselines across most training budgets: for IMDb, the improvements are up to 23.68%; for ASRS, the improvements are up to 28.31%; for AIvsCR, the improvements are up to 8.67%.

D Statistical Significance Results

In this section, we provide a summary of pairwise one-tailed t-tests comparing *LwR-RC* with the two most competitive baselines, *Lw/oR-BERT* and *RB-WAVG-BERT*, for all three datasets at varying budget regiments. Table 4 shows the p–values of one-tailed paired t-tests with the alternative hypothesis "the performance of *LwR-RC* is better than the baseline approach". As this result shows, *LwR-RC* statistically significantly outperforms both *Lw/oR-BERT* and *RB-WAVG-BERT* at most budget regiments with a p–value of less than 0.05.

E Extension to Multi-class Classification

In our paper, we focused on binary classification. *LwR-RC*, can be extended to multi-class classification with a few modifications. For multi-class classification, let $y_i \in \{c_1, c_2, \cdots, c_k\}$: $f(x_i) = p(y_i = c \mid x_i) = softmax(W_z z_i)$ for some parameter vector/matrix W_z, where c is the correct label for instance x_i and z_i is the vector representation of x_i. Assuming that y_i is encoded as one-hot representation, the classification loss function, L_{clf}, will then change from binary cross-entropy to categorical cross-entropy:

$$L_{clf} = -\frac{1}{|\mathcal{L}|}\sum_i (y_i \cdot log(p(y_i|x_i))) \quad (8)$$

For modeling the *DvM* objective of *LwR-RC*, let $\mu_i = W_z z_i$ and $\mu'_i = W_z z'_i$, where z'_i is the vector representation of x'_i. Then, for the correct label c, we would like $\mu_i^c > 0$ and $\mu_i^c > \mu_i'^c$, which results in the following objective function:

$$L_{DvM} = -\frac{1}{|\mathcal{L}|}\sum_i (y_i \cdot log(p(y_i|x_i, x'_i)) \quad (9)$$

where,

$$p(y_i|x_i, x'_i) = softmax(-(\mu_i - \mu'_i)) \quad (10)$$

We define the ranking loss similarly for the *RvM* component, this time using the *R* and *M* documents and their respective scores $\mu_i^r = W_z z_i^r$ and $\mu'_i = W_z z'_i$, where z_i^r is the vector representation of x_i^r. The ranking loss L_{RvM} is then defined as:

$$L_{RvM} = -\frac{1}{|\mathcal{L}|}\sum_i (y_i \cdot log(p(y_i|x_i^r, x'_i)) \quad (11)$$

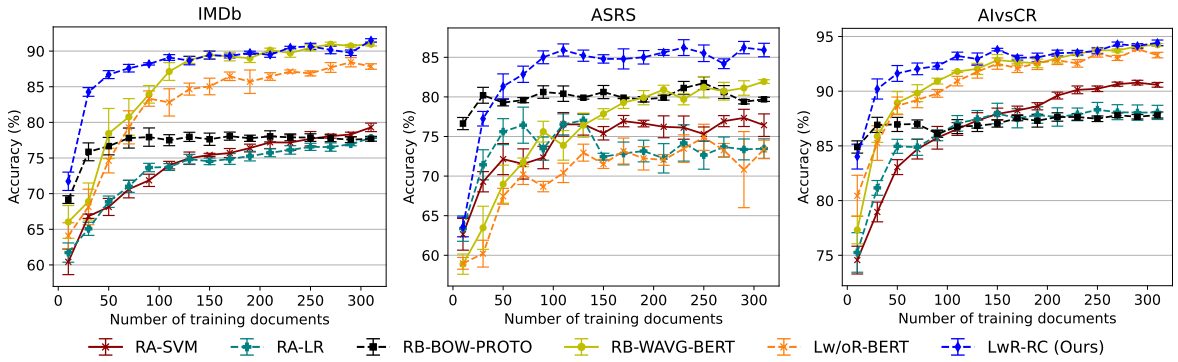

Figure 5: Comparison between our approach, *LwR-RC*, and the five baselines using the best hyper-parameter setting for each method.

Budget	Dataset	*Lw/oR-BERT*	*LwR-RC*	Abs. Imp.	% Imp.	*RB-WAVG-BERT*	*LwR-RC*	Abs. Imp.	% Imp.
10	IMDb	64.10	71.73	7.63	11.91%	66.03	71.73	5.70	8.63%
	ASRS	58.98	63.57	4.59	7.79 %	58.88	63.57	4.69	7.97%
	AIvsCR	80.44	84.02	3.58	4.45%	77.32	84.02	6.70	**8.67%**
30	IMDb	68.13	84.27	16.13	**23.68%**	68.90	84.27	15.37	**22.30%**
	ASRS	60.20	77.24	17.04	**28.31%**	63.47	77.24	13.78	**21.70%**
	AIvsCR	85.26	90.20	4.94	**5.79%**	85.88	90.20	4.32	5.03%
50	IMDb	74.63	86.70	12.07	16.17%	78.43	86.70	8.27	10.54%
	ASRS	67.45	81.33	13.88	20.57%	68.98	81.33	12.35	17.90%
	AIvsCR	88.66	91.62	2.96	3.34%	88.94	91.62	2.68	3.01%
70	IMDb	79.30	87.60	8.30	10.47%	80.77	87.60	6.83	8.46%
	ASRS	70.20	82.86	12.65	18.02%	71.73	82.86	11.12	15.50%
	AIvsCR	89.20	92.04	2.84	3.18%	89.82	92.04	2.22	2.47%
90	IMDb	83.33	88.20	4.87	5.84%	83.93	88.20	4.27	5.08%
	ASRS	68.67	85.00	16.33	23.77%	75.61	85.00	9.39	12.42%
	AIvsCR	89.80	92.30	2.50	2.78%	90.92	92.30	1.38	1.52%
110	IMDb	82.80	89.10	6.30	7.61%	87.13	89.10	1.97	2.26%
	ASRS	70.41	85.92	15.51	22.03%	73.88	85.92	12.04	16.30%
	AIvsCR	90.92	93.22	2.30	2.53%	91.80	93.22	1.42	1.55%
130	IMDb	84.67	88.67	4.00	4.72%	88.77	88.67	N/A	N/A
	ASRS	72.96	85.20	12.24	16.78%	76.43	85.20	8.78	11.48%
	AIvsCR	91.78	92.94	1.16	1.26%	92.04	92.94	0.90	0.98%
150	IMDb	85.03	89.43	4.40	5.17%	89.47	89.43	N/A	N/A
	ASRS	71.53	84.80	13.27	18.54%	77.86	84.80	6.94	8.91%
	AIvsCR	92.52	93.80	1.28	1.38%	92.84	93.80	0.96	1.03%
170	IMDb	86.47	89.30	2.83	3.28%	89.23	89.30	0.07	0.07%
	ASRS	73.16	84.80	11.63	15.90%	79.29	84.80	5.51	6.95%
	AIvsCR	92.10	93.06	0.96	1.04%	92.68	93.06	0.38	0.41%
190	IMDb	85.70	89.77	4.07	4.75%	89.00	89.77	0.77	0.86%
	ASRS	72.24	85.00	12.76	17.66%	79.90	85.00	5.10	6.39%
	AIvsCR	92.58	93.08	0.50	0.54%	92.54	93.08	0.54	0.58%
210	IMDb	86.43	89.47	3.03	3.51%	90.13	89.47	N/A	N/A
	ASRS	72.04	85.61	13.57	18.84%	80.92	85.61	4.69	5.80%
	AIvsCR	92.82	93.48	0.66	0.71%	93.02	93.48	0.46	0.49%

Table 3: Accuracy results comparing ***LwR-RC*** with the two fine-tuned BERT baselines, ***Lw/oR-BERT*** and ***RB-WAVG-BERT***, at varying budgets. 'Abs. Imp.' represents the absolute accuracy improvements that *LwR-RC* provides over the baselines and '% Imp.' represents the percentage of improvements in accuracy that *LwR-RC* provides with respect to the baselines. 'N/A' represents that *LwR-RC* doesn't provide any improvements over the baselines. For each dataset, the highest improvements that *LwR-RC* provides over the two baselines across all budgets are highlighted in boldface.

where,

$$p(y_i|x_i^r, x_i') = softmax(-(\mu_i^r - \mu_i')) \quad (12)$$

We combine the classification loss L_{clf} with the ranking losses, L_{DvM} and L_{RvM}, resulting in the main objective function for our approach for multi-class classification:

$$L = (1 - \lambda_1 - \lambda_2)L_{clf} + \lambda_1 L_{DvM} + \lambda_2 L_{RvM} \quad (13)$$

Budget	Dataset	p–value	
		Lw/oR-BERT	*RB-WAVG-BERT*
10	IMDb	**0.011**	**0.012**
	ASRS	**0.005**	**0.031**
	AIvsCR	**0.047**	**0.002**
30	IMDb	**0.001**	**0.002**
	ASRS	**0**	**0.005**
	AIvsCR	**0.033**	**0.006**
50	IMDb	**0.001**	**0.04**
	ASRS	**0**	**0**
	AIvsCR	**0.005**	**0.006**
70	IMDb	**0.009**	**0.022**
	ASRS	**0**	**0**
	AIvsCR	**0.028**	**0.027**
90	IMDb	**0.009**	**0.018**
	ASRS	**0**	**0.004**
	AIvsCR	**0.008**	**0.006**
110	IMDb	**0.017**	0.078
	ASRS	**0**	**0.001**
	AIvsCR	**0.004**	**0.011**
130	IMDb	**0.008**	0.574
	ASRS	**0**	**0.001**
	AIvsCR	**0.019**	**0.01**
150	IMDb	**0.028**	0.511
	ASRS	**0**	**0**
	AIvsCR	**0.039**	0.065
170	IMDb	**0.004**	0.456
	ASRS	**0.001**	**0.01**
	AIvsCR	**0.016**	0.185
190	IMDb	**0.049**	0.181
	ASRS	**0**	**0.02**
	AIvsCR	0.231	**0.031**
210	IMDb	**0.001**	0.921
	ASRS	**0**	**0.003**
	AIvsCR	0.086	0.101

Table 4: Statistical significance results comparing *LwR-RC* to the two fine-tuned BERT baselines for all three datasets at varying budget regiments. We report the p–values for one-tailed paired t-tests with the alternative hypothesis "the performance of our approach is better than the baseline approach". The results where *LwR-RC* performs statistically significantly better than the baselines (with a p–value of less than 0.05) are boldfaced.

F AIvsCR Dataset Collection and Annotation

In our study, we experimented with three human-annotated datasets, IMDb, ASRS, and AIvsCR. We collected and annotated the AIvsCR dataset. To construct this dataset, we first collected 6,000 articles equally from two categories, cs.AI and cs.CR, from arXiv.org using a custom search query in the arXiv API. We provide the code, including the custom search queries, that we used to collect the data from arXiv.org with the supplementary material.

For annotating the AIvsCR dataset, two annotators, A1 and A2, were provided with the same instructions as Zaidan et al. (2007) described in their paper: highlight the rationales at your best but do not mark everything.

Statistics	A1	A2
# rationales per document	3.8	8.4
# rationale words per document	17.4	31.3
% rationales overlapping with A1	100	30.5
% rationales overlapping with A2	64.0	100

Table 5: Average statistics for AIvsCR dataset and the two annotators, A1 and A2. The table presents the number of rationales and the number of rationale words per document provided by the two annotators, as well as the inter-annotator agreement for their rationale annotation.

We calculated the inter-annotator agreement for the rationales, where the rationales provided by the two annotators for the same document are considered as overlapping if they have at least one word in common, following the same manner of Zaidan et al. (2007). The relevant statistics are shown in Table 5. To make the best use of each annotator's effort, for every document, we kept the overlapping words, phrases, and sentences between the two annotators' highlighted rationales as the final rationales, as illustrated in the following example:

- A1: ***rectified linear units*** are among the most widely used ***activation function*** in a broad variety of tasks in vision.
- A2: rectified linear units are among the most widely used ***activation function in a broad variety of tasks in vision.***
- Final: rectified linear units are among the most widely used ***activation function*** in a broad variety of tasks in vision.