Sequential Targeting: an incremental learning approach for data imbalance in text classification

Joel Jang[a,1], Yoonjeon Kim[b], Kyoungho Choi[c], Sungho Suh[d,e,*]

[a]Department of Computer Science and Engineering, Korea University, 02841 Seoul, Republic of Korea.
[b]Department of Applied Statistics, Yonsei University, 03722 Seoul, Republic of Korea.
[c]Media Tech Group, Naver Corp., 13561 Seongnam-si, Gyeonggi-do, Republic of Korea.
[d]Department of Computer Science, TU Kaiserslautern, 67663 Kaiserslautern, Germany
[e]Smart Convergence Group, Korea Institute of Science and Technology Europe Forschungsgesellschaft mbH, 66123 Saarbrücken, Germany.

Abstract

Classification tasks require a balanced distribution of data to ensure the learner to be trained to generalize over all classes. In real-world datasets, however, the number of instances vary substantially among classes. This typically leads to a learner that promotes bias towards the majority group due to its dominating property. Therefore, methods to handle imbalanced datasets are crucial for alleviating distributional skews and fully utilizing the under-represented data, especially in text classification. While addressing the imbalance in text data, most methods utilize sampling methods on the numerical representation of the data, which limits its efficiency on how effective the representation is. We propose a novel training method, Sequential Targeting(ST), independent of the effectiveness of the representation method, which enforces an incremental learning setting by splitting the data into mutually exclusive subsets and training the learner adaptively. To address problems that arise within incremental learning, we apply elastic weight consolidation. We demonstrate the effectiveness of our method through experiments on simulated benchmark datasets (IMDB) and data collected from NAVER.

Keywords: data imbalance, incremental learning, information extraction, text classification, text mining, sentiment analysis

1. Introduction

Classification is an important task of knowledge discovery in databases and data mining. It is a task of learning a discriminative function from the given data that classifies previously unseen data to the correct classes. Current research trends in natural language processing focus on developing deep neural network(DNN) models such as BERT [1] that have been pre-trained with a large text corpus and thus show immense improvement in different text classification tasks. Despite the success of large pre-trained models, DNNs still suffer from generalizing to a balanced testing criterion in cases of data imbalance [2]. In realistic settings, it is rarely the case where the discrete distribution of the data acquired is perfectly balanced across all classes. Realistic settings are prone to be skewed to specific classes while such classes are often the class of interest. Some situations may be binary, as in detecting spams in forums [3]. The majority of the contents posted from users are not spams and is in accordance with the intended goal. As a result, the number of spam samples is sparse in comparison to non-spam samples. Imbalanced data may also occur in a multi-classification setting such as classifying articles into different categories [4].

Text classification can be used for numerous application purposes. In this paper, we address the problem of detecting sexual harassment and toxicity in comments from news articles. In the name of anonymity, online discussion platforms have become a place where people undermine, harass, humiliate, threaten, and bully others [5] based on their superficial characteristics such as gender, sexual orientation, and age [6]. Each toxic comment can further be classified into classes based on their degree of toxicity [7]. Figure 1 shows the overall procedure of detecting sexual harassment and performing sentimental analysis on comment data in the wild. When collecting and annotating comments, data skewness occurs naturally since users do not consider data imbalance levels when writing toxic or non-toxic comments. Classifiers trained in imbalanced settings tend to become biased toward the class with more samples in the training data. This is because standard deep learning architectures [8] do not take the data imbalance level into consideration. In order to develop intelligent classifiers, methods to temper the classifier from biasing towards certain classes are of great importance.

Previous methods addressing data imbalance in the text can be divided into data-level and algorithm-level methods. Data-level methods [9, 10] apply manipulation on the data by undersampling majority classes or oversampling minority classes. However, most of the methods require an effective numerical representation algorithm since methods are applied directly to the representation instead of on the actual text. Algorithm-level

*Corresponding Author.
Email addresses: wkddydpf@korea.ac.kr (Joel Jang), yoonkim313@yonsei.ac.kr (Yoonjeon Kim), k.h.choi@navercorp.com (Kyoungho Choi), s.suh@kist-europe.de (Sungho Suh)

[1]This work was done while the author was at NAVER.

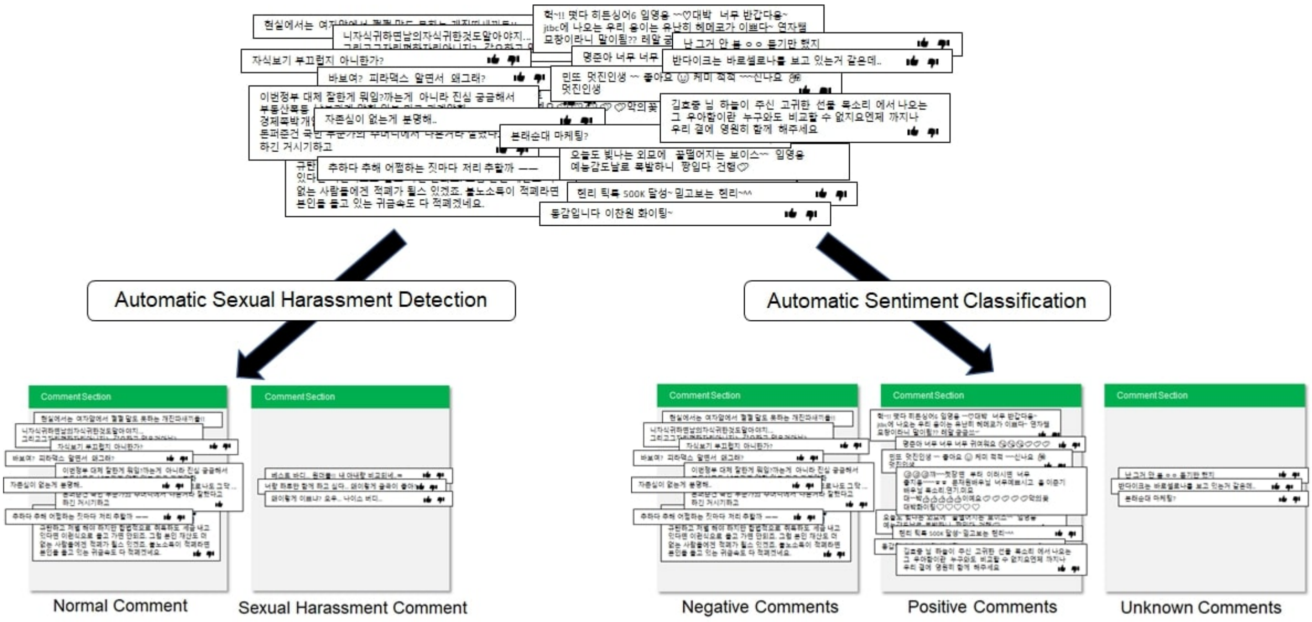

Figure 1: Detecting sexual harassment and sentiment in Korean comment data.

methods modify the underlying learner or its output to reduce bias towards the majority group. However, these methods are task-sensitive and somewhat heuristic since it requires the researchers to modify the classifier considering the innate properties of the task. This property leads to the inefficiency in training the learner since heuristic approaches are often time-consuming and arbitrary. Since only traditional oversampling and undersampling methods, which simply duplicate or sample data instances, are independent of these two limitations, methods addressing data imbalance in the text without the utilization of feature spaces or task-dependent is needed.

We propose a novel training architecture, **Sequential Targeting (ST)**, that handles the data imbalance problem by forcing an incremental learning setting. ST divides the entire training data set into mutually exclusive partitions, target-adaptively balancing the data distribution. Target distribution is a predetermined distributional setting that enables the learner to exert maximum performance when trained with. In an imbalanced distributional setting, the target distribution is idealistic to follow a uniform distribution where all the classes hold equal importance. Optimal class distribution may differ by innate property of the data but research shows that a balanced class distribution has an overall better performance compared to other distributions [11]. The remaining partitions are then sorted in the magnitude of similarity with the target distribution which is measured by KL-divergence. The first partition of the split data is imbalanced while the last partition is arbitrarily modeled to be uniform across classes and all the partitions are utilized to train the learner sequentially.

We handle the issue of catastrophic forgetting [12], which is an inevitable phenomenon when transfer learning, by utilizing Elastic Weight Consolidation(EWC) [13] to stabilize the knowledge attained from the previous tasks. This allows the discriminative model to learn from the incoming data while not forgetting the previously inferred parameters from previous tasks.

Our proposed method is both independent of the numerical representation method and the task at hand. We validate our method on simulated datasets(IMDB) with varying imbalance levels and apply our method to a real-world application. We annotated and construct three datasets consisting of comments made by users from different social platforms of NAVER[2]: two for detecting sexual harassment and one for multiple sentimental analysis. Annotations on the datasets were improved iteratively by in-lab annotations and crowdsourcing. Experimental results show that ST outperforms traditional approaches, with a notable gap, especially in extremely imbalanced cases. Lastly, ST proves to be compatible with previous approaches.

Our contribution in this paper is three-folds:

- We introduce a novel method addressing the data imbalance problem by splitting the data and incrementally training a learner to perform balanced learning by paying equal attention to the minority data as to the majority data. The learner's initial focus on the majority class is compensated by the redistributed task that comes sequentially, resulting in an overall increase of performance. We perform experiments not only on simulated benchmark text datasets but also on datasets constructed from real-world data. Results demonstrate the superiority of our method.

- Incremental learning techniques are utilized to prevent catastrophic forgetting on tasks encountered before. The novelty of our method stands since we applied incremental learning to address the data imbalance problem.

[2]NAVER is the Korean No.1 web search portal where around 16 million users visit every day. *www.naver.com*

- We propose a novel method that is not only free from the dependencies of previous methods addressing the data imbalance in text, but also compatible with the previous methods. The performance increase is shown when utilized together.

The rest of the paper is organized as follows. Section 2 summarizes related works. Section 3 provides the details of the proposed method. Section 4 presents dataset descriptions, experiment setups, and qualitative experimental results on various datasets. Finally, Section 5 concludes the paper.

2. Related Works

2.1. Methods Handling the Data Imbalance Problem in Text Data

Previous researchers have proposed data-level methods and algorithm-level methods to address the data imbalance problem. Data-level techniques such as Synthetic Minority Oversampling Technique (SMOTE) [14, 15] is one of the most used approaches. Its basic idea is to interpolate between the observations of minority classes to oversample the training data. Different variants of SMOTE such as SMOTE-SVM [16]and MWMOTE [17] has also been proposed. Other methods try to augment less-frequently observed classes by translating samples from more-frequently observed classes [18]. Recently, deep learning methods have been applied in imbalanced data setting to generate diverse instances of minority classes by utilizing Generative Adversarial Nets [19, 20, 21].

These approaches are non-heuristic but arbitrary to some extent in that researchers should determine what to resample from the data. A classifier should effectively approximate the hyper-plane which forms a boundary between classes. If the resampled data does not fully represent the quality data points that are crucial in deciding the hyper-plane, the model fails to generalize classification performance on new observations. Moreover, most approaches focus on handling imbalance on images, and only a few [9, 10] have tried applying SMOTE and its variants to text data. However, these methods assume effective numerical representations of data since SMOTE works only in feature spaces. Even though effective numerical representation methods have been recently proposed such as GLOVE [22] and Word2Vec [23], methods that can be utilized without having a dependency on the representation method is still in need.

Some text data augmentation methods do not have this limitation and apply augmentation directly on the text data instead of on the numerical representations. Easy data augmentation techniques (EDA) [24] introduces four powerful operations: synonym replacement, random insertion, random swap, and random deletion.

Alternately, algorithm-level methods [25, 26], commonly implemented with a weight or cost schema, modify the underlying learner or its output to reduce bias towards the majority group. Algorithm-level methods modify the structure of the decision process of how much to focus on under-represented samples. This could be implemented by assigning the cost matrix as a penalty [27]. Moreover, loss functions could be modified, such as in the case of focal loss [28], which reshapes the standard cross-entropy loss such that it penalizes the loss assigned to well-classified instances.

2.2. Elastic Weight Consolidation

Catastrophic forgetting [12] is a phenomenon of a learner forgetting about the previously learned task when encountered with another task. This phenomenon occurs naturally when training a learner in an incremental learning setting [29]. One of the major approaches to overcome this issue is to use an ensemble of networks, each trained with individual tasks [30]. However, this approach has a complexity issue. Alternately, Fernando et al. [27] proposed an ensemble approach which attempts to fix the parameters learned from the previous task and train new parameters on consecutive tasks. This method has successfully reduced complexity issues, but performance suffers from the lack of trainable parameters.

Kirkpatrick et al. [13] proposed Elastic Weight Consolidation(EWC) that consider training the neural network from a probabilistic perspective. This approach assumes that neural network parameters follow Gaussian distribution for each task and attempt to find the optimal instance among the distribution that performs well for all tasks. It focuses on developing a regularization term that buffers the model from forgetting the previously trained information. It implements a modified regularization term that consolidates knowledge across tasks by imposing restrictions on the model during training to slow down updating certain important parameters from the previous task.

EWC is implemented extensively in different applications such as in multi-task learning [31] and mining tasks [32]. Unlike the previous application of EWC which focuses on working in different domains, we apply EWC to address the data imbalance problem.

3. Sequential Targeting

We first introduce a broad overview of the novel training architecture: Sequential Targeting. Next, we show how this method has been applied to address the data imbalance problem.

3.1. Matching the Target Distribution with Elastic Weight Consolidation

We propose a novel model architecture of **forced incremental learning** on the imbalanced setting. The term *forced incremental learning* has rarely, if ever, been used since incremental learning is a much more complicated task. In incremental learning, the new data is referred to as different *task* in this paper. The task is consistently provided therefore the learner needs to be constantly updated on the new task. Because of catastrophic forgetting, learners perform much better when trained with an individual task than continually being updated with multiple tasks. However, by applying EWC, we have proven it is beneficial to force an incremental learning setting where the given data distribution varies substantially from

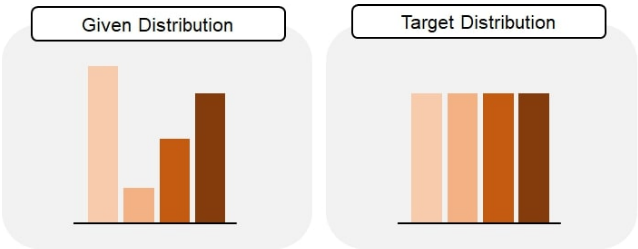

Figure 2: An example of when given data distribution differs from the target data distribution

the target data distribution, such as in the case of data imbalance. The **target distribution**, denoted as $\mathcal{P}_T$, is the idealistic data distribution in which the learner would perform the best if trained on Figure 2 shows an example of a case where the given distribution differs from the target distribution. Left shows the distribution in case of data skewness and right shows the distribution in the data is balanced.

Algorithm 1 Training architecture for Sequential Targeting

Require: Given task $\mathcal{D}_{total}$ divided into multiple tasks, $\mathcal{D}_1, \mathcal{D}_2, ...\mathcal{D}_k$, so that $\bigcup \mathcal{D}_i = \mathcal{D}_{total}$ and $\bigcap \mathcal{D}_i = \emptyset$.
Initialize: Sort the tasks $\mathcal{D}_1, \mathcal{D}_2, ...\mathcal{D}_k$ in the decreasing order of Kullback–Leibler divergence from the target distribution P_T
for $i \leftarrow 1$ to k **do**
  if $i = 1$ **then**
    Randomly initialize θ_i
  else
    Set loss function to $\mathcal{L}(\theta_i) + \sum_j \frac{\lambda}{2} F_j(\theta_{i,j} - \theta^*_{i-1,j})^2$
  end if
  Train θ_i from $\mathcal{D}_i$
  Save optimal θ_i as θ^*_i
end for

Our method effectively improves the model by dividing a given task into multiple tasks so that the union of the tasks is the initially given task and the intersection of the divided tasks is an empty set. Each task is partitioned into varying distributions: $\mathcal{P}_1, \mathcal{P}_2, ...\mathcal{P}_k$ where k is the number of splits. The learner is sequentially trained on these tasks in the order of similarity with the target distribution, which is measured with KL-divergence. The following has to hold:

$$\begin{aligned} KL(P_T \| P_{prev}) &> KL(P_T \| P_{curr}), \\ \text{where } P_i &\xrightarrow{\mathcal{D}} P_T \text{ as } i \rightarrow k. \end{aligned} \tag{1}$$

KL divergence is used to measure the discrepancy between the task distributions and the target distribution.

Using a single learner over all tasks, we incrementally condition the maximum performance from the previous task and stabilize the learned parameters on the current task. Optimizing the network parameters θ is equivalent to finding their most feasible values given some data D. We can compute this conditional probability $p(\theta|D)$ from the prior probability of the parameters $p(\theta)$ and the probability of the data $p(D|\theta)$ by using Bayes' rule:

$$\log p(\theta|D) = \log p(D|\theta) + \log p(\theta) - \log p(D). \tag{2}$$

By assuming that the data is split into two independent parts, one defining the previous task D_{prev} and the other current task D_{curr}. we can rearrange (2) by applying the Bayesian update rule:

$$\log p(\theta|D) = \log p(D_{curr}|\theta) + \log p(\theta|D_{prev}) - \log p(D_{curr}). \tag{3}$$

Equation (3) shows how the prior distribution learned from the data of the previous task is further enriched by the data given at the current task.

We apply EWC between transfer learning from task to task. Due to the intractability of the posterior distribution, EWC poses an assumption on the prior distribution to follow a Gaussian distribution with the mean as θ^*_{prev} and the Fisher information matrix, F, of the previous task as the precision. To minimize the loss of information, the objective function is defined as follows:

$$\mathcal{L}(\theta) = \mathcal{L}_{curr}(\theta_{curr}) + \sum_i \frac{\lambda}{2} F_i(\theta_{curr,i} - \theta^*_{prev,i})^2, \tag{4}$$

where $\mathcal{L}_{curr}(\theta)$ sets the loss for the current task, λ sets the importance of the previous task and i labels each parameter.

Since the tasks are sorted dependent on the KL divergence value with the target distribution, the learner is adaptively trained on data distribution similar to the target distribution more and more as tasks proceed. At last, the learner is trained on task that is identical to the target distribution ensuring $KL(P_T \| P_k) \approx 0$. As tasks develop, we apply EWC so that the learner does not forget about previous tasks. The training procedure for the proposed method is summarized in Algorithm 1.

The number of data splits, k, and how each task is partitioned to have varying KL-divergence values are both highly dependent on what $\mathcal{P}_T$ is defined to be. How ST has been applied to address the data imbalance problem is explained in the next subsection.

3.2. Adaptive Balancing

Imbalanced data setting tempers the model from learning extensively from under-represented instances. Therefore, it is crucial to enforce the learner to acquire knowledge equally among classes. In an idealistic setting, a learner is trained with equally represented data. However, in a realistic setting, imbalanced data includes some under-represented classes and the learner has difficulty acquiring sufficient knowledge to generalize. ST enables balanced learning by redistributing the task to approach the target distribution as tasks develop.

It is idealistic to assume $\mathcal{P}_T \stackrel{d}{=} Uniform[0, p]$ in an imbalanced data setting therefore when p denotes the number of classes the target distribution is discrete uniform: $\{\frac{1}{p}, \frac{1}{p}, , , , , \frac{1}{p}\}$. In our method, the training data is redistributed into two splits so that

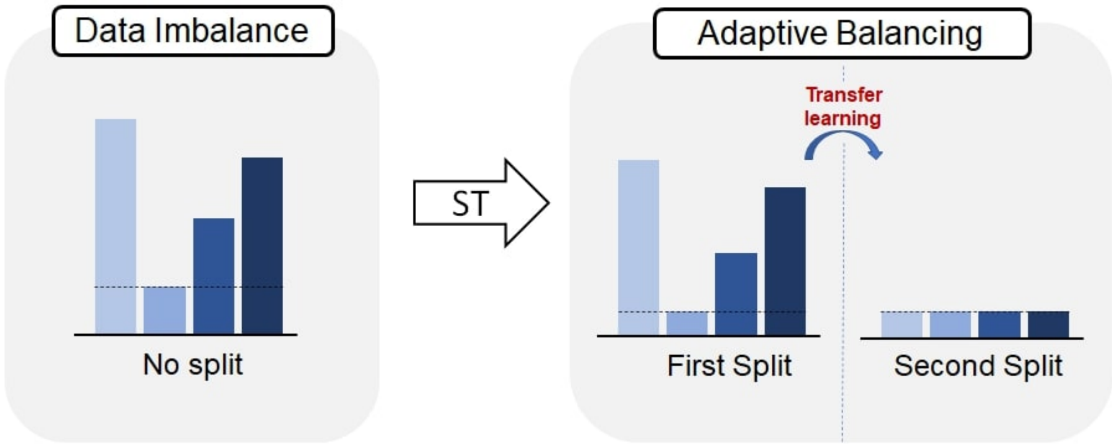

Figure 3: Data distribution before and after applying ST in the case of multi-classification with data imbalance.

the last split is identical to the uniform distribution as shown in Figure 3. The left shows the data distribution setting before ST is applied and the right shows the case after ST is applied. ρ and η are ratios that are explained in Section 4.

This training architecture lets the learner pay more focus to the under-represented data by manipulating the learning sequence. Sequentially training the learner to be exposed to an increasing portion of minority class data benefits the overall performance. Moreover, applying dropout to the layers and implementing EWC during the transfer between tasks proves to help the learner maintain the knowledge acquired from the previous split. We believe this training approach is the first of its kind, with the best of our knowledge.

4. Experiments

4.1. Evaluation Metrics and ratios

Accuracy is commonly used to measure the performance of a classification model. However, when it comes to skewed data, accuracy alone can be misleading and thus other appropriate metrics are needed to correctly evaluate the performance of the model. In this paper, we use precision, recall, and macro F1-score to objectively evaluate the model in a skewed data setting. *Precision* measures the percentage of actual positive among the number of positively predicted samples. *Recall* measures the percentage of the truly positive instances that was correctly predicted by the model. As precision and recall are in a trade-off relationship, selecting a learner that performs well on both metrics would be a reasonable policy. *Macro F1-score* combines both precision and recall as a harmonic mean weighted with equal importance on each class rather it be sparse or rich. In this paper, F1-score is used as the core metric for measuring performance.

Following the conventions of the previous research on imbalanced data [30, 33], we employ two distinct ratios used throughout the experiment to represent the imbalanced state of the data. One parameter is a ratio between the number of instances in majority classes and the number of instances in minority classes and defined as follow:

$$\rho = \frac{|\max(C_i)|}{|\min(C_i)|}. \tag{5}$$

where C_i is the set of instances in class i and N is the total number of classes. Another

The other parameter η is a parameter that compares the relative number of minority class instances among splits. For instance, if the first task consists of 100 samples in minority class and the second task consists of 50 samples, then η will be 2:1. We experienced various combinations of these three ratios and concluded that η does not have a significant effect on model performance. Therefore, the number of instances of the minority classes between the splits is fixed as 1:1 throughout the whole experiment.

4.2. Dataset Descriptions

In this paper, we validate and apply our proposed method on four text datasets(IMDB, NAVER-Catcall-Yellow, NAVER-Catcall-Red, NAVER-Posneg). In the case of *IMDB*, the dataset was deliberately made into varying imbalanced states as shown in Table 1. *NAVER-Posneg, NAVER-Catcall-Yellow, and NAVER-Catcall-Orange* are Korean text data of comments crawled and annotated to improve the AI Clean Bot 2.0 [34] used to detect toxic comments at NAVER. All of the data were collected from the news, entertainment, and sports platforms of NAVER. Annotations on the datasets were made through crowdsourcing and improved iteratively by in-lab annotations. All of the datasets have been divided into train, validation, and test sets for objective comparison of methods.

IMDB [35] is a text dataset, which contains 50,000 movie reviews with binary labels (positive/negative). Reviews have been preprocessed, and each review is encoded as a sequence of word indexes. Three different imbalance ratios have been deliberately made ($\rho = 10, 20, 50$) to test how each method performs as the imbalance level worsens. The positive reviews are regarded as the positive class.

NAVER-Catcall-Yellow and NAVER-Catcall-Red consists of comments that have been collected and annotated to train a model that detects sexual harassment in comments. It is made

Table 1: Simulated IMDB Datasets with Variations of ρ ratio and NAVER Datasets.

Dataset	ρ	Train		Validation		Test	
		Minority	Majority	Minority	Majority	Minority	Majority
IMDB	10	1,250	12,500	2,500	2,500	10,000	10,000
	20	625	12,500	2,500	2,500	10,000	10,000
	50	250	12,500	2,500	2,500	10,000	10,000
NAVER-Posneg	8	6,355	48,342	314	2,498	621	1453
NAVER-Catcall-Yellow	10	2,650	26,792	140	1,410	73	427
NAVER-Catcall-Red	15	1,840	27,602	97	1,453	73	427

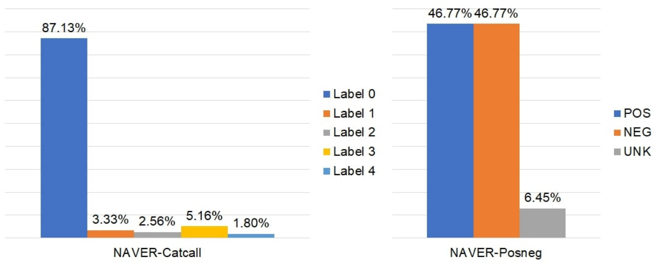

Figure 4: Initial data distribution of collected and annotated comments of *NAVER-Catcall* and *NAVER-Posneg*.

up of 31,392 comments labeled into 5 different classes. The labeling rules used for the annotation process and how the initial annotated data was used to construct *NAVER-Catcall-Yellow* and *NAVER-Catcall-Red* are shown in Table 2. The second column shows the initial labeling by annotators. The third and fourth column shows the binary labeling standards for *yellow* and *red*. As shown, models trained with *yellow* standard was expected to become more sensitive since the initial *label 2* was considered a positive case of sexual harassment. Likewise, models trained with *red* standard was expected to have lower recall, but show higher precision since it only perceives clear sexual harassment comments as positive. The left bar graph in Figure 4 shows the initial data distribution and the percentage of each class labeled by the annotators.

Table 2: Labeling rules used to construct NAVER-Catcall-Yellow and NAVER-Catcall-Red.

Description	**Label**	**Yellow**	**Red**
No one will perceive it as sexual harassment	0	0	0
Contains sexual language	1		
Some may perceive as sexual harassment	2	1	
Clearly intended as sexual harassment	3		1
Shows advocacy of sexual violence	4		

The actual analysis of imbalance found on real comments was much severe ($\rho > 100$) since sexual harassment comments were not common. Because of this, we crawled comments from media sources where sexual harassment comments were more likely to be written. Despite sampling from less imbalanced data sources, the data imbalance level still turned out to be significant: $\rho \approx 10$ for *yellow* and $\rho \approx 15$ for *red*.

NAVER-Posneg was collected to train a sentimental analysis model and consists of 95,875 comments labeled into three different classes: *positive, negative, and unknown*. The *unknown* class was added since some comments were hard to label either as *positive* or *negative* based on the comment itself without knowing the full context. *Positive* and *negative* comments were proportionate. However, skewnewss existed in *unknown* comments ($\rho \approx 8$). The right bar graph in Figure 4 shows the data distribution of the constructed dataset.

4.3. Experimental Setup

In our experiments, our proposed method has been extensively compared with two data-level methods, random oversampling (ROS) and random under-sampling (RUS). For ROS, instances from the minority classes were sampled and duplicated to match the number of instances in the majority classes. For RUS, we randomly down-sampled all majority classes so that they had the same number of instances as the minority classes.

We explore the full capability of ST by combination with ROS. ROS is combined with ST by oversampling the first split; no sampling method is applied to the second split. In the case of *NAVER-Catcall-Red*, we perform experiments with EDA [24] as well as combine EDA with ST by applying EDA to the first split. We only utilize synonym replacement and random insertion for EDA.

Even though large models, such as BERT [1], tend to perform better in imbalanced settings, a simple CNN+LSTM model architecture was used since large models suffer from slow inference time [36]. Inference time and model size are important factors when considering actual model deployment that will be used in applications. The details of the model structure used for experiments are shown in Table 3. While BERT has 110M parameters, our model has only 1M parameters (*more than 100 times smaller*) and thus show faster inference time.

At the last FC layer, sigmoid activation is used for binary classification and softmax activation is used for multi-classification. The CNN layer efficiently extracts the higher-level representations while the BiLSTM and UniLSTM layers obtain the sentence representation [37].

For each configuration, five independent trial runs were made with different initial weights. This setting ensures the effect of weight initialization to be ruled out in evaluating model performance. Among the five trials, the model checkpoint with the highest validation f1-score was used for evaluation. All the experimental settings including epochs, learning rate, and model architecture are fixed for each corresponding task.

Table 3: Neural Network Architecture of the CNN+LSTM classifier

Layers	Activation function
Embedding	-
1-Dimension Convolution Layer (Dropout: 20%)	ReLU
BiLSTM + Residual Connection from Embedding	-
UniLSTM (Dropout: 20%)	-
1-Dimension Global Max Pooling	-
1-Dimension Fully Connected	Sigmoid/Softmax

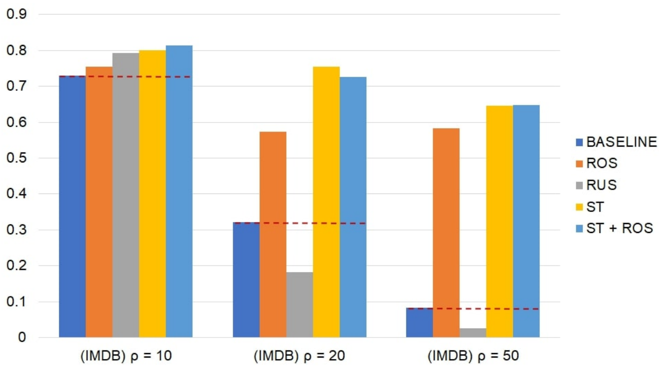

Figure 5: Experimental Results on IMDB Datasets in terms of F1-Score.

4.4. Experiment Results

IMDB Results. Table 4 shows the experimental results on the simulated IMDB datasets. Baseline methods show a substantial performance decrease when the data imbalance level worsens. This is because of the significant decrease in the recall, which signifies only a small portion of positive reviews were actually classified correctly. We observe training the model with ST outperforms, if not on par with, traditional methods. Results show a considerable increase in recall when ST is applied. This is because the model can classify more positive reviews correctly since the focus was put on the under-represented class. It is natural to expect a significant decrease in precision since the focus has been shifted away from the majority class. However, the drop is minimal when EWC is applied. This is because EWC helps the model to remember valuable information obtained during the training of the first split as the model is trained with the balanced second split. As shown in Figure 5, the performance gap between ST and baseline grows substantially as the severity of data imbalance increases: 8.52%, 43.27%, and 56.38%.

NAVER-Catcall-Yellow and NAVER-Posneg Results. Table 5 shows the experimental results on *NAVER-Catcall-Yellow* and *NAVER-Posneg*. In both cases, ST shows the best performance. Applying ST in *NAVER-Catcall* and *NAVER-Posng* shows a performance increase of 16.89% and 2.4% in terms of F1-score, respectively. Applying ST has a much more significant impact in the case with only *NAVER-Catcall* than in the case of *NAVER-Posneg*, as shown in Figure 6. This is because the former is a binary classification task and the latter is a multi-classification task. Considering that most methods are ineffective and even may cause a negative effect on multi-classification tasks [38], the relatively small increase in performance is still meaningful.

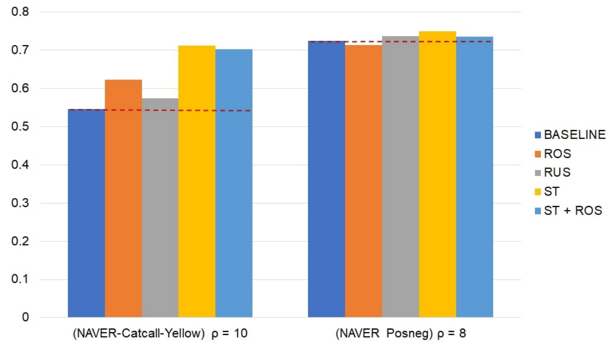

Figure 6: Experimental Results on NAVER-Catcall-Yellow and NAVER-Posneg in terms of F1-Score.

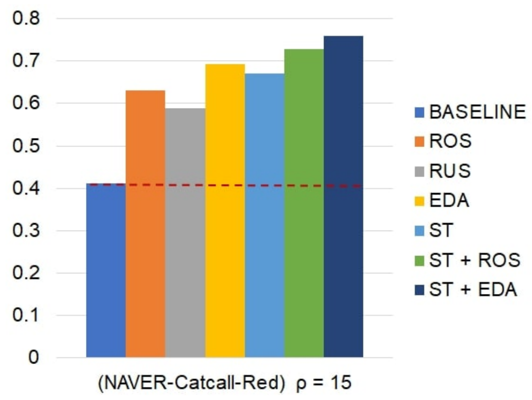

Figure 7: Experimental Results on NAVER-Catcall-Red in terms of F1-Score.

NAVER-Catcall-Red Results Table 6 shows the experimental results on *NAVER-Catcall-Red*. EDA shows a higher F1-Score than ROS. This is because instead of a plain duplication approach for oversampling, EDA utilized different operations that allowed data to be oversampled similar to actual data in the wild. The effect is emphasized when used together with ST as shown in Figure 7. ST+EDA shows the best performance among compared methods and shows an increase of 34.76% in terms of F1-score compared to the Baseline.

5. Conclusion

It is seldom the case data in the wild has a balanced distribution. In realistic settings, there is a limitation of acquiring relatively balanced data through choices of balanced data sources. Handling data skewness is a crucial problem because learning from imbalanced data inevitably brings bias toward frequently observed classes. Data-level manipulation tries to under-sample the majority classes or over-sample the minority classes. But these methods tend to discard valuable information from observations of majority classes or overfit to a sparse representation of minority classes, especially as the imbalance level gets higher. Moreover, recent methods such as SMOTE cannot be applied directly to the text data.

We propose ST, which effectively circumvents these issues by simply decomposing the data into k splits and sequentially

Table 4: Experimental Results on Simulated IMDB Datasets with Varying ρ ratio

Ratios	$\rho = 10$			$\rho = 20$			$\rho = 50$		
Metrics	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall
Baseline	0.7291	**0.8751**	0.6248	0.3216	0.6677	0.2118	0.0833	0.69	0.0443
ROS	0.7548	0.702	0.8162	0.5738	0.5839	0.564	0.5837	0.5618	0.6061
RUS	0.7921	0.7305	0.865	0.1837	0.6758	0.1063	0.0256	0.6065	0.0131
ST	0.8002	0.7181	**0.9035**	**0.7543**	**0.7368**	0.7727	0.6457	**0.7188**	0.586
ST + ROS	**0.8143**	0.7984	0.8307	0.7259	0.6728	**0.788**	**0.6471**	0.6156	**0.6819**

Table 5: Experimental Results on NAVER-Catcall-Yellow and NAVER-Posneg

Ratios	(NAVER-Catcall-Yellow) $\rho = 10$			(NAVER-Posneg) $\rho = 8$		
Metrics	F1	Precision	Recall	F1	Precision	Recall
Baseline	0.5455	**0.8108**	0.411	0.7249	**0.7756**	0.7288
ROS	0.6218	0.8043	0.5068	0.7126	0.7754	0.7182
RUS	0.5733	0.5857	0.5616	0.7363	0.7444	0.7353
ST	**0.7144**	0.6974	0.726	**0.7489**	0.7651	**0.7489**
ST + ROS	0.7018	0.6122	**0.8219**	0.7352	0.7609	0.7365

Table 6: Experimental Results on NAVER-Catcall-Red

Ratios	(NAVER-Catcall-Red) $\rho = 15$		
Metrics	F1	Precision	Recall
Baseline	0.411	**0.9091**	0.274
RUS	0.5893	0.8462	0.4521
ROS	0.6316	0.878	0.4931
EDA [24]	0.6917	0.7667	0.6301
ST	0.671	0.7015	0.6438
ST + ROS	0.7273	0.8136	0.6575
ST + EDA [24]	**0.7586**	0.7639	**0.7534**

training a learner in the decreasing order of KL divergence with the target distribution, which in the case of data imbalance problem is the discrete uniform distribution. Through extensive experiments, we show our architecture proves to be compatible with previous methods and outperforms existing methods when validated on simulated as well as real-application tasks. Our model shows superiority in performance because it enables more focus to be put on minority instances while not forgetting about majority instances. We believe that our work makes a meaningful step towards handling data skewness in text classification and the application of incremental learning methods focused on the data imbalance problem.

For future work, ensemble methods can be used by varying the η ratio to train multiple weak learners. Moreover, since ST can be applied simultaneously with algorithm-level methods, proven methods such as focal loss [28] and cost-sensitive deep neural network [26] could be implemented together to increase optimal performance.

Acknowledgement

This research work is supported by NAVER Corp.

References

[1] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv:1810.04805 (2018).

[2] Q. Dong, S. Gong, X. Zhu, Imbalanced deep learning by minority class incremental rectification, IEEE transactions on pattern analysis and machine intelligence 41 (6) (2018) 1367–1381.

[3] P. Ratadiya, R. Moorthy, Spam filtering on forums: A synthetic oversampling based approach for imbalanced data classification, arXiv preprint arXiv:1909.04826 (2019).

[4] Y. Suh, J. Yu, J. Mo, L. Song, C. Kim, A comparison of oversampling methods on imbalanced topic classification of korean news articles, Journal of Cognitive Science 18 (4) (2017) 391–437.

[5] N. S. Samghabadi, S. Maharjan, A. Sprague, R. Diaz-Sprague, T. Solorio, Detecting nastiness in social media, in: Proceedings of the First Workshop on Abusive Language Online, 2017, pp. 63–72.

[6] M. ElSherief, V. Kulkarni, D. Nguyen, W. Y. Wang, E. Belding, Hate lingo: A target-based linguistic analysis of hate speech in social media, arXiv preprint arXiv:1804.04257 (2018).

[7] P. Fortuna, S. Nunes, A survey on automatic detection of hate speech in text, ACM Computing Surveys (CSUR) 51 (4) (2018) 1–30.

[8] J. Moon, W. I. Cho, J. Lee, Beep! korean corpus of online news comments for toxic speech detection, arXiv preprint arXiv:2005.12503 (2020).

[9] W. Satriaji, R. Kusumaningrum, Effect of synthetic minority oversampling technique (smote), feature representation, and classification algorithm on imbalanced sentiment analysis, in: 2018 2nd International Conference on Informatics and Computational Sciences (ICICoS), IEEE, 2018, pp. 1–5.

[10] C. Padurariu, M. E. Breaban, Dealing with data imbalance in text classification, Procedia Computer Science 159 (2019) 736–745.

[11] L. Rokach, Ensemble-based classifiers, Artificial intelligence review 33 (1-2) (2010) 1–39.

[12] R. M. French, Catastrophic forgetting in connectionist networks, Trends in cognitive sciences 3 (4) (1999) 128–135.

[13] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, et al., Overcoming catastrophic forgetting in neural networks, Proceedings of the national academy of sciences 114 (13) (2017) 3521–3526.

[14] N. V. Chawla, K. W. Bowyer, L. O. Hall, W. P. Kegelmeyer, Smote: synthetic minority over-sampling technique, Journal of artificial intelligence research 16 (2002) 321–357.

[15] H. Han, W.-Y. Wang, B.-H. Mao, Borderline-smote: a new over-sampling method in imbalanced data sets learning, in: International conference on intelligent computing, Springer, 2005, pp. 878–887.

[16] H. M. Nguyen, E. W. Cooper, K. Kamei, Borderline over-sampling for imbalanced data classification, International Journal of Knowledge Engineering and Soft Data Paradigms 3 (1) (2011) 4–21.

[17] S. Barua, M. M. Islam, X. Yao, K. Murase, Mwmote–majority weighted minority oversampling technique for imbalanced data set learning, IEEE Transactions on knowledge and data engineering 26 (2) (2012) 405–425.

[18] J. Kim, J. Jeong, J. Shin, M2m: Imbalanced classification via major-to-minor translation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 13896–13905.

[19] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, in: Advances in neural information processing systems, 2014, pp. 2672–2680.

[20] S. Suh, H. Lee, J. Jo, P. Lukowicz, Y. O. Lee, Generative oversampling method for imbalanced data on bearing fault detection and diagnosis, Applied Sciences 9 (4) (2019) 746.

[21] S. Suh, H. Lee, P. Lukowicz, Y. O. Lee, CEGAN: Classification Enhancement Generative Adversarial Networks for unraveling data imbalance problems, Neural Networks (2020).

[22] J. Pennington, R. Socher, C. D. Manning, Glove: Global vectors for word representation, in: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), 2014, pp. 1532–1543.

[23] P. Bojanowski, E. Grave, A. Joulin, T. Mikolov, Enriching word vectors with subword information, Transactions of the Association for Computational Linguistics 5 (2017) 135–146.

[24] J. Wei, K. Zou, Eda: Easy data augmentation techniques for boosting performance on text classification tasks, arXiv preprint arXiv:1901.11196 (2019).

[25] B. Krawczyk, Learning from imbalanced data: open challenges and future directions, Progress in Artificial Intelligence 5 (4) (2016) 221–232.

[26] S. H. Khan, M. Hayat, M. Bennamoun, F. A. Sohel, R. Togneri, Cost-sensitive learning of deep feature representations from imbalanced data, IEEE transactions on neural networks and learning systems 29 (8) (2017) 3573–3587.

[27] C. Fernando, D. Banarse, C. Blundell, Y. Zwols, D. Ha, A. A. Rusu, A. Pritzel, D. Wierstra, Pathnet: Evolution channels gradient descent in super neural networks, arXiv preprint arXiv:1701.08734 (2017).

[28] T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollár, Focal loss for dense object detection, in: Proceedings of the IEEE international conference on computer vision, 2017, pp. 2980–2988.

[29] Z. Li, D. Hoiem, Learning without forgetting, IEEE transactions on pattern analysis and machine intelligence 40 (12) (2017) 2935–2947.

[30] M. Buda, A. Maki, M. A. Mazurowski, A systematic study of the class imbalance problem in convolutional neural networks, Neural Networks 106 (2018) 249–259.

[31] S.-A. Rebuffi, H. Bilen, A. Vedaldi, Efficient parametrization of multi-domain deep neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 8119–8127.

[32] P. Karisani, J. Ho, E. Agichtein, Domain-guided task decomposition with self-training for detecting personal events in social media, in: Proceedings of The Web Conference 2020, 2020, pp. 2411–2420.

[33] J. M. Johnson, T. M. Khoshgoftaar, Survey on deep learning with class imbalance, Journal of Big Data 6 (1) (2019) 27.

[34] K. Lee, K. Choi, S. Kim, Cleanbot 2.0: Detect malicious comments (short sentences) that understand context ai, https://d2.naver.com/helloworld/7753273 (July 2020).

[35] A. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, C. Potts, Learning word vectors for sentiment analysis, in: Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies, 2011, pp. 142–150.

[36] Z. Li, E. Wallace, S. Shen, K. Lin, K. Keutzer, D. Klein, J. E. Gonzalez, Train large, then compress: Rethinking model size for efficient training and inference of transformers, arXiv preprint arXiv:2002.11794 (2020).

[37] C. Zhou, C. Sun, Z. Liu, F. Lau, A c-lstm neural network for text classification, arXiv preprint arXiv:1511.08630 (2015).

[38] Z.-H. Zhou, X.-Y. Liu, Training cost-sensitive neural networks with methods addressing the class imbalance problem, IEEE Transactions on knowledge and data engineering 18 (1) (2005) 63–77.