EasyAug: An Automatic Textual Data Augmentation Platform for Classification Tasks

Siyuan Qiu[*]
Ant Financial Services Group

Binxia Xu[*]
Ant Financial Services Group

Jie Zhang
Ant Financial Services Group

Yafang Wang[†]
Ant Financial Services Group

Xiaoyu Shen
Max Planck Institute for Informatics

Gerard de Melo
Rutgers University

Chong Long
Ant Financial Services Group

Xiaolong Li
Ant Financial Services Group

Abstract

Imbalanced data is a perennial problem that impedes the learning abilities of current machine learning-based classification models. One approach to address it is to leverage data augmentation to expand the training set. For image data, there are a number of suitable augmentation techniques that have proven effective in previous work. For textual data, however, due to the discrete units inherent in natural language, techniques that randomly perturb the signal may be ineffective. Additionally, due to the substantial discrepancy between different textual datasets (e.g., different domains), an augmentation approach that facilitates the classification on one dataset may be detrimental on another dataset. For practitioners, comparing different data augmentation techniques is non-trivial, as the corresponding methods might need to be incorporated into different system architectures, and the implementation of some approaches, such as generative models, is laborious. To address these challenges, we develop EasyAug, a data augmentation platform that provides several augmentation approaches. Users can conveniently compare the classification results and can easily choose the most suitable one for their own dataset. In addition, the system is extensible and can incorporate further augmentation approaches, such that with minimal effort a new method can comprehensively be compared with the baselines.

Keywords

imbalanced data, data augmentation, text generation, model fusion, text classification

ACM Reference Format:
Siyuan Qiu, Binxia Xu, Jie Zhang, Yafang Wang, Xiaoyu Shen, Gerard de Melo, Chong Long, and Xiaolong Li. 2020. EasyAug: An Automatic Textual Data Augmentation Platform for Classification Tasks. In *Companion Proceedings of the Web Conference 2020 (WWW '20 Companion), April 20–24, 2020, Taipei, Taiwan.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3366424.3383552

1 Introduction

Motivation and Problem. The effectiveness of machine learning-based classifiers largely depends on the size and quality of the training data. A common reason for a classifier to exhibit dismal accuracy is a lack of sufficient training data for specific categories. Indeed, even with large amounts of data, many learning algorithms remain sensitive to imbalances in the distribution of target classes. To address this *imbalanced data* issue without simply discarding large amounts of majority-class training data, a common option is to adjust the class distribution by adding more data samples to the minority classes. Unfortunately, using human-generated data can be expensive and difficult to scale, making automated approaches desirable. A substantial number of recent studies have considered automatic data augmentation for computer vision tasks. In image classification, very simple data augmentation tricks such as rotation, translation, and flipping can engender considerable performance gains. Beyond these, further more complex operations such as RGB channel intensity alterations [9] and adding noise directly to features [17] have as well been explored to improve the model's performance.

In real use, however, due to the discrepancies between different datasets, an augmentation approach proving effective on one dataset may turn out to be detrimental on others and it is non-trivial to find the most suitable method by re-implementing all of the existing approaches. In light of this, image data augmenting libraries such as Albumentations [1], Augmentor [2], and Imgaug [6], which integrate various augmentation techniques have been very useful for practitioners. For natural language text, although certain augmentation approaches such as Random Duplication, Easy Data Augmentation (EDA) [15], and generative models [3, 5] have been put forth, to the best of our knowledge, there is only one augmentation library assembling different methods for textual data: NLPAug [10]. This library provides a repertoire of textual augmentation techniques at the character and word level, which can be regarded as improved variants of EDA. However, generative methods, which have been widely explored in recent years remain absent in this library, and it does not provide any evaluation tools to assess and compare the effectiveness of different techniques. Inspired by the image augmentation libraries mentioned above, we develop an easy-to-use platform within the PyTorch framework: **EasyAug**, which incorporates all prominent textual data augmentation approaches and also is able to evaluate the effectiveness of

[*] The first authors contributed equally.
[†] Corresponding author: Yafang Wang, Email: yafang.wyf@antfin.com

WWW '20 Companion, April 20–24, 2020, Taipei, Taiwan
© 2020 IW3C2 (International World Wide Web Conference Committee), published under Creative Commons CC-BY 4.0 License.
ACM ISBN 978-1-4503-7024-0/20/04.
https://doi.org/10.1145/3366424.3383552

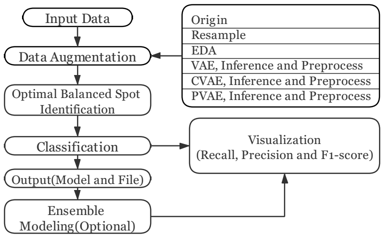

Input Data
Data Augmentation
Optimal Balanced Spot Identification
Classification
Output(Model and File)
Ensemble Modeling(Optional)
Origin
Resample
EDA
VAE, Inference and Preprocess
CVAE, Inference and Preprocess
PVAE, Inference and Preprocess
Visualization
(Recall, Precision and F1-score)

Figure 1: The architecture of EasyAug, the data augmentation system for textual data.

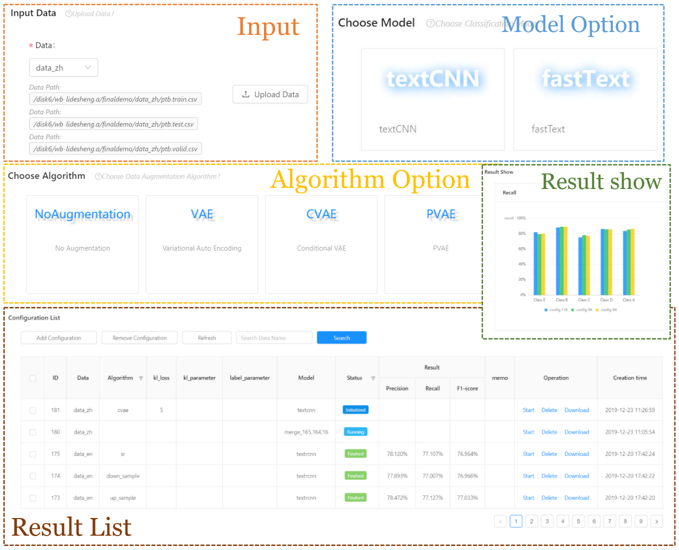
each method with several popular text classification models on datasets uploaded by users such that they can easily choose the most effective technique for their dataset. In addition, our system provides an API to add further data augmentation methods, so existing methods can serve as baselines to evaluate the merits of a new technique. This augmentation platform is currently applied in the intelligent customer services system of Ant Financial Services Group to facilitate the question classification tasks, as some types of questions are asked more often than other types. Our system is able to mitigate this issue and help to achieve a better classification result by comparing the effectiveness of different augmentation techniques and classification models.

System Overview. The current version of our platform supports input data files in two languages (English and Mandarin Chinese). While the system provides a selection of example datasets, users are encouraged to upload their own domain-specific dataset. Subsequently, the training data is processed and extended using the selected augmentation method. Given that the resulting classification results are the most reliable means of assessing the effectiveness of an augmentation approach, users are requested to choose a classification learning algorithm so that the evaluation results can be presented. Additionally, since model fusion has proven beneficial in automated classification, we also offer model ensembling, i.e., merging user-chosen models, as a further option to improve the classification results.

Outline. In the following, the architecture of our system will be described with an illustration of each main component. Subsequently, we explicitly showcase the user interfaces and demonstrate how to use the platform. A screencast of our system is available online at: https://youtu.be/0acM_ez9nJE

2 Proposed System

The overall framework of EasyAug is illustrated in Figure 1. The input of the overall system consists of three data files: the training set, testing set, and validation set, which should already be split. When the three sets are fed into the framework, the training set is extended by means of the user-selected augmentation approach, while the testing and validation set are kept for the classification step. Upon generating the artificial data samples with a specific augmentation method, we do not simply oversample the minority classes, so that all classes have the same amount of data as the largest majority one. Instead, we combine the operations of oversampling and undersampling by considering several balance levels between the amount of data in the largest and smallest categories, as this tends to yield better results. For each balance level, one



* Data: data_zh

Data Path: /disk6/wb-lidesheng.a/finaldemo/data_zh/ptb.train.csv

Data Path: /disk6/wb-lidesheng.a/finaldemo/data_zh/ptb.test.csv

Data Path: /disk6/wb-lidesheng.a/finaldemo/data_zh/ptb.valid.csv

□	ID	Data	Algorithm	kl_loss	kl_parameter	label_parameter	Model	Status	Result			memo	Operation	Creation time
									Precision	Recall	F1-score			
□	181	data_zh	cvae	5			textcnn	Initialized					Start | Delete | Download	2019-12-23 11:26:59
□	180	data_zh					merge_165,164,16	Running					Start | Delete | Download	2019-12-23 11:05:54
□	175	data_en	sr				textrcnn	Finshed	78.120%	77.107%	76.964%		Start | Delete | Download	2019-12-20 17:42:24
□	174	data_en	down_sample				textrcnn	Finshed	77.893%	77.007%	76.966%		Start | Delete | Download	2019-12-20 17:42:22
□	173	data_en	up_sample				textrcnn	Finshed	78.472%	77.127%	77.033%		Start | Delete | Download	2019-12-20 17:42:20

< 1 2 3 4 5 6 7 8 9 >

Figure 2: The screenshot of our system.

augmented training set is generated, and the classification task is conducted over all training sets with different balance levels so that the most optimal balance level can be identified. Additionally, we provide XGBoost-based [4] ensemble modeling as an optional operation so that the classification results can be improved by integrating the advantages of various classifiers.

The evaluation metrics we use for classification in this system include: precision P, recall R, and F_1 score. EasyAug not only provides macro-averaged indicators, but also visualizes the precision, recall, and F_1 score for each individual category through bar charts, such that the effects on the classification results for each class can easily and intuitively be compared, which benefits the analysis of the imbalance issue on a specific dataset. For all augmentation approaches and classifiers, the system provides the augmented data file with the optimal balance level. In addition, for generative methods, our system is able to provide the trained models.

2.1 Augmentation approaches

As our system aims at providing a platform for testing the effectiveness of different sampling and augmentation approaches, a new method can easily be incorporated into our system. At present, we provide three types of methods to show the merits of the pipeline of our system. These are: random resampling, word-level transformations, and VAE-based text generation models. Further methods will be integrated in future versions.

Random Resampling. There are two widely used resampling operations: random undersampling and random oversampling. In our system, we provide an approach that randomly drops out some data samples in the majority classes so that all classes have the same amount of data as the smallest one, which we refer to as *downsampling*. Since mere downsampling might discard too much of the original data and thus impede the generalization ability of the classifier, we provide another approach: *random duplication*, which invokes a combination of undersampling and oversampling by dropping some data samples in the majority classes, while duplicating data samples in minority classes arbitrarily. One point to be noted is that the degree of undersampling and oversampling for each class is decided by the aforementioned optimal balance level identification.

Word-level Transformation. Word-level transformations generate new sentences while largely preserving the semantic features of the original data sample, and have proven effective in a number of works. Synonym replacement (SR) [8] is the most intuitive technique, as it entails replacing one or more random words in a sample with one of their respective synonyms to construct a new sentence. Compared with SR, the EDA method involves applying one of several word-level operations to a sentence arbitrarily, including synonym replacement, random insertion, random swap, and random deletion. Previous experimental results [15] suggest that EDA is able to improve over SR on the considered dataset.

VAE-based Models. Variational Autoencoding (VAE) [7] assumes a text generation process in two steps: 1) A latent code z is sampled from a prior distribution $p(z)$; 2) The corresponding text is then generated based on the conditional distribution $p_\theta(x \mid z)$. Compared with a vanilla language model, the VAE variants are able to generate text with superior diversity due to the prior space of the additional latent code. This higher degree of diversity is considered beneficial for data augmentation. Since the exact value of $p_\theta(x \mid z)$ cannot be analytically derived, VAE circumvents this issue by resorting to the evidence lower bound (ELBO):

$$\begin{aligned}\mathcal{L}(\theta,\phi) &= \log \mathbb{E}_{z\sim p(z)}\, p_\theta(x|z) = \log \mathbb{E}_{z\sim q_\phi(z|x)} \frac{p_\theta(x,z)}{q_\phi(z|x)} \\ &\geq \mathbb{E}_{z\sim q_\phi(z|x)} \log p_\theta(x|z) - \mathrm{KL}(q_\phi(z|x)||p(z)),\end{aligned} \tag{1}$$

where $q_\phi(z|x)$ is an encoder trained to map each text into its posterior latent code space.

Eq. 1 can also be modified to a label-dependent format. This essentially turns it into a conditional variational autoencoder (CVAE) [13, 14, 16, 18]. The objective function is changed accordingly to condition on an extra label l:

$$\begin{aligned}\mathcal{L}(\theta,\phi) &= \log \mathbb{E}_{z\sim p_\theta(z|l)} p_\theta(x|z,l) \\ &\geq \mathbb{E}_{z\sim q_\phi(z|x,l)} \log p_\theta(x|z,l) - \mathrm{KL}(q_\phi(z|x,l)||p_\theta(z|l)).\end{aligned} \tag{2}$$

Here, $p_\theta(z|l)$ is parametrized as a label-dependent Gaussian distribution with a diagonal covariance matrix.

There are two strategies to generate augmented text via VAE: 1) train separate **unconditional VAEs** (Eq. 1) for each class; 2) train a single **conditional VAE** (Eq. 2) by taking the class information as an additional input. Moreover, for the sampling strategy of the latent code, we also provide two options: sampling from the **prior distribution** or from the **posterior distribution** for each training data point. As the lower variance of the posterior distribution corresponds to text semantically similar to the training data, while the prior distribution exhibits a greater diversity [3, 12], we combine these variants as the following three kinds of models for data augmentation:

(1) Unconditional VAE + prior sampling (**VAE**)
(2) conditional VAE + prior sampling (**CVAE**)
(3) conditional VAE + posterior sampling (**PVAE**)

Since we observe that these models would generate a lot of repeated sentences or a sentence containing several continuously repeated words, some preprocessing operations such as deduplication are applied immediately after the inference procedure.

2.2 Ensembling

After the input data has been augmented, the processed data is fed into one of the neural classifiers provided in the system, which include BiLSTM, TextCNN, TextRCNN, and FastText ones at present. Apart from the discrepancy between different dataset, the characteristics of different classification models can also affect the performance of augmentation techniques. For example, an augmentation method might facilitate one classifier more than other classifiers. To combine the advantages of multiple neural architectures and achieve better classification results, the tool provides an ensemble modeling operation in the last step of our framework. This operation is flexible and optional, which means that users are able to individually choose the ensembled models, e.g., BiLSTM and FastText ones, to be fused. They can also choose not to ensemble any model.

In this work, we leverage XGBoost [4], a very fast and highly scalable tree boosting system as a top-level classifier based on the classification results of the individual models by concatenating the results predicted by each model as the input features.

2.3 Experiment Implementation

Table 1: Class distribution of training data in the Intelligent Customer Service (ICS) and News datasets

Dataset	Class Distribution				
ICS (Zh)	Class A	Class B	Class C	Class D	Class E
	62,739	59,693	49,084	77,680	76,716
News (En)	Politics	Wellness	Entertainment	Style & Beauty	Travel
	23728	14292	10756	7672	6945

To elucidate the potential of our platform, we conduct experiments over two example datasets: the Intelligent Customer Service Dataset (ICS) with Mandarin Chinese documents, and the News Category Dataset (News) [11] with English ones. The class distribution in the training split is given in Table 1. The ICS dataset contains more than 400K user questions collected from a real intelligent customer service system in five different categories from the finance domain. The News corpus is an archive of Huffington Post articles from the years 2012 to 2018, comprising 200K news headlines and short descriptions along with their corresponding categories. In our experiment, we utilize short descriptions to predict the labels, and thus data samples without such short descriptions are omitted. We construct our News dataset by selecting the 5 largest categories.

The macro-level experimental results on both ICS and News are given in Table 2. We can easily observe that the effectiveness of different data augmentation techniques vary across these two datasets. Comparing the two, the data volume for each category in ICS is much larger than those of News, and the degree of imbalance in much lower in the ICS data, given that in News, the number of data samples in the largest category (Politics) is more than three times that of the smallest one (Travel). Hence, the influence of the imbalance issue on News is more severe, which is reflected in the comparison of gains for different data augmentation techniques on the two datasets: The gains for EDA, the most effective method on

Table 2: Macro F$_1$-score for TextRCNN classification model over the ICS and News test sets comparing different training set sampling/augmentation approaches.

	Original	Down Sample	Combined Sample	SR	EDA	VAE	CVAE	PVAE
ICS (Zh)	0.8363	0.8333	0.8367	0.8337	0.8357	0.8367	0.8350	0.8357
News (En)	0.7525	0.7697	0.7703	0.7688	0.7720	0.7727	0.7680	0.7713

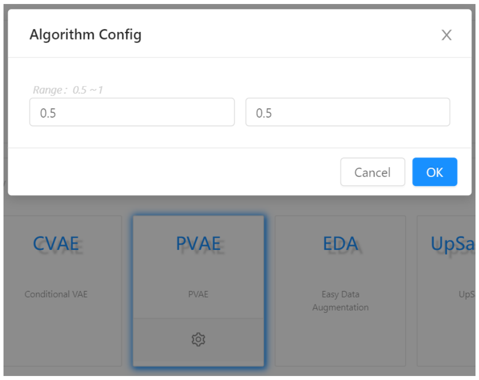
News is near 2% (absolute), while on the more balanced ICS, none of the augmentation approaches greatly improve the results.

3 Demonstration

The Web interface of our system is illustrated in Figure 2.

Input and model configuration. The top left part of Figure 2 shows the format of the input datasets. The system currently supports both English and Chinese data. The data needs to have been split into training, testing, and validation sets based on the user needs. The data currently needs to be packaged as three CSV files inside a single ZIP file. After uploading the data, the user is asked to choose an augmentation approach and a classification model as presented under *Algorithm Option* and *Model Option* in Figure 2.

For some of the augmentation methods such as the VAE variants, there are parameters that can be adjusted. For example, as presented in Fig. 3, when users activate the Setting icon for PVAE, a small window appears, which allows them to alter the parameters of this generative model. For the user's convenience, we have set default values for these parameters according to the experimental results on our datasets. However, we highly recommend that users tune the parameters themselves for their specific dataset. After the configuration is set, users may add the new configuration to the processing list (Figure 2 bottom) and select **Start** to run the task.



Algorithm Config

Range： 0.5 ~1

0.5

0.5

Figure 3: Parameter setting.

Output and evaluation. Once the **Status** of a task transitions from *Running* to *Finished*, the training process is completed and the overall evaluation results are presented, including the macro-averaged precision, recall, and F_1 score. Thus, the effectiveness of different data augmentation techniques on a dataset can easily be compared. However, focusing only on the macro-level classification results may not suffice when dealing with class imbalance. In real-world scenarios, it often makes sense to pay special attention to the improvements on the minority classes, as these are more affected by class imbalance issues. Hence, in our platform, when users select two or more than two finished tasks for the same dataset, a comparison of the performance gains for each category is plotted as a bar chart (Figure 2 middle right). Additionally, when several finished tasks augmented by the same method over various classifiers are chosen and the **Model Merge** button is selected, the ensemble modeling process is triggered.

4 Conclusion

To the best of our knowledge, our automatic data augmentation platform, EasyAug, is the first tool in the community to intuitively compare the effectiveness of a diverse set of augmentation approaches for different classification models on uploaded textual data. The system substantially simplifies working on text classification tasks with augmentation, as it is no longer necessary to implement and evaluate individual methods manually. Users can conveniently try different augmentation methods on their data and compare the results. Our system already supports sampling, word-level transformations, and VAE-based methods, and can easily be extended to evaluate further augmentation approaches.

References

[1] E. Khvedchenya V. I. Iglovikov A. Buslaev, A. Parinov and A. A. Kalinin. 2018. Albumentations: fast and flexible image augmentations. *ArXiv e-prints* (2018). arXiv:1809.06839

[2] Marcus D Bloice, Peter M Roth, and Andreas Holzinger. 2019. Biomedical image augmentation using Augmentor. *Bioinformatics* 35, 21 (04 2019), 4522–4524.

[3] Samuel R Bowman, Luke Vilnis, Oriol Vinyals, Andrew M Dai, Rafal Jozefowicz, and Samy Bengio. 2015. Generating sentences from a continuous space. *arXiv preprint arXiv:1511.06349* (2015).

[4] Tianqi Chen and Carlos Guestrin. 2016. XGBoost: A Scalable Tree Boosting System. In *SIGKDD* (San Francisco, California, USA) *(KDD '16)*. ACM, New York, NY, USA, 785–794.

[5] Yutai Hou, Yijia Liu, Wanxiang Che, and Ting Liu. 2018. Sequence-to-sequence data augmentation for dialogue language understanding. *arXiv preprint arXiv:1807.01554* (2018).

[6] Alexander B. Jung, Kentaro Wada, Jon Crall, Satoshi Tanaka, Jake Graving, Sarthak Yadav, Joy Banerjee, Gábor Vecsei, Adam Kraft, Jirka Borovec, Christian Vallentin, Semen Zhydenko, Kilian Pfeiffer, Ben Cook, Ismael Fernández, Weng Chi-Hung, Abner Ayala-Acevedo, Raphael Meudec, Matias Laporte, et al. 2019. imgaug. Online; accessed 25-Sept-2019.

[7] Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* (2013).

[8] Sosuke Kobayashi. 2018. Contextual augmentation: Data augmentation by words with paradigmatic relations. *arXiv preprint arXiv:1805.06201* (2018).

[9] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*. 1097–1105.

[10] Edward Ma. 2019. nlpaug: Data augmentation for NLP.

[11] Rishabh Misra. 2018. *News Category Dataset*. https://doi.org/10.13140/RG.2.2.20331.18729

[12] Iulian Vlad Serban, Alessandro Sordoni, Ryan Lowe, Laurent Charlin, Joelle Pineau, Aaron Courville, and Yoshua Bengio. 2017. A hierarchical latent variable encoder-decoder model for generating dialogues. In *AAAI*.

[13] Xiaoyu Shen, Hui Su, Yanran Li, Wenjie Li, Shuzi Niu, Yang Zhao, Akiko Aizawa, and Guoping Long. 2017. A Conditional Variational Framework for Dialog Generation. In *ACL*. 504–509.

[14] Kihyuk Sohn, Honglak Lee, and Xinchen Yan. 2015. Learning structured output representation using deep conditional generative models. In *Advances in neural information processing systems*. 3483–3491.

[15] Jason W Wei and Kai Zou. 2019. Eda: Easy data augmentation techniques for boosting performance on text classification tasks. *arXiv preprint arXiv:1901.11196* (2019).

[16] Xinchen Yan, Jimei Yang, Kihyuk Sohn, and Honglak Lee. 2016. Attribute2image: Conditional image generation from visual attributes. In *ECCV*. Springer, 776–791.

[17] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. 2017. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412* (2017).

[18] Tiancheng Zhao, Ran Zhao, and Maxine Eskenazi. 2017. Learning Discourse-level Diversity for Neural Dialog Models using Conditional Variational Autoencoders. In *ACL*. 654–664.