Exploiting Proximity Search and Easy Examples to Select Rare Events

Daniel Kang, Alex Derhacobian, Kaoru Tsuji, Trevor Hebert, Peter Bailis, Tadashi Fukami, Tatsunori Hashimoto, Yi Sun, Matei Zaharia

Abstract

A common problem practitioners face is to select rare events in a large dataset. Unfortunately, standard techniques ranging from pre-trained models to active learning do not leverage proximity structure present in many datasets and can lead to worse-than-random results. To address this, we propose EZMODE, an algorithm for iterative *selection* of rare events in large, unlabeled datasets. EZMODE leverages active learning to iteratively train classifiers, but chooses the *easiest* positive examples to label in contrast to standard uncertainty techniques. EZMODE also leverages proximity structure (e.g., temporal sampling) to find difficult positive examples. We show that EZMODE can outperform baselines by up to 130× on a novel, real-world, 9,000 GB video dataset.

1 Introduction

A common problem practitioners wish to use ML to solve is to select rare events in large, bespoke datasets, i.e., find $\mathcal{D}_+ = \{x \in \mathcal{D} : \mathcal{O}(x) = 1\}$ for some oracle $\mathcal{O}$. For example, ecologist researchers may wish to select hummingbirds in field videos, but hummingbird visits can be less than 0.2% of the video. This problem is common across many domains: political scientists may also be interested in selecting newspaper articles about the New Deal from large corpora of newspaper scans; moderators may be interested in finding social media posts about vaccine misinformation among all posts. This data can subsequently be used for manual analysis or for training classifiers downstream [1].

Unfortunately, standard practice in ML, ranging from executing pre-trained models on the dataset to active learning to train high-quality classifiers, can be ineffective in these scenarios. For example, we found that pre-trained models can be *worse than random* on these datasets due to distribution shift. Active learning, in which a classifier is iteratively trained on the labeled data from the original dataset, can also fail to produce good classifiers for selection.

One reason is that neither of these methods leverage inherent *proximity structure* present in many datasets. Abstractly, many hard-to-classify examples are close in some auxiliary space to easy-to-classify examples. For example, many easy examples of hummingbirds are temporally close to difficult examples of hummingbirds (Figure 1), so selection should leverage the temporal nature of these datasets. Newspaper columnists may write about similar topics, so once an article by a particular author is found to be about the New Deal, it may be beneficial to look for other articles by that author.

To make selection more efficient, we propose an algorithm EZMODE for selecting rare events. In contrast to many ML settings, ranging from active learning to co-training, the goal is explicitly not to train a high quality classifier: EZMODE aims to *select* rare events and does so by leveraging proximity structure (e.g., temporal or spatial closeness). Given an oracle (typically an expert human labeler) and a method of ranking data (typically an ML model's confidence), EZMODE will iteratively: 1) execute the oracle on the most confident unlabeled examples, 2) execute the oracle on examples found using proximity structure on any positive examples found in the previous step, 3) regenerate rankings (typically by retraining a model), and 4) repeat the process.

Preprint. Under review.

(a) Easy example.

(b) Hard example.

Figure 1: An example of an easy and difficult to detect hummingbird (as determined by an expert labeler) temporally close. Our best model ranks the left bird in the 99th percentile and right bird is ranked in the 91th percentile. In contrast, the left example can be found via temporal sampling, lowering the need to label 8% of the dataset.

We further show that iteratively training classifiers using previously found data strongly boosts performance. However, in contrast to standard active learning, EZMODE does not aim to train a high quality classifier over all the data. Instead, EZMODE aims to train a classifier that ranks easy examples from the rare class as high as possible (e.g., Figure 1). For example, a hummingbird may leave the frame and reappear in motion (e.g., Figure 1). In this example, the hummingbird may leave for many frames and reappear later in the video, making it difficult to incorporate this knowledge in a classifier, but is easy to select using temporal information. Given that many difficult examples can be found via proximity structure, EZMODE uses the example difficulty to iteratively train classifiers to select the easy examples, which we find lead to higher precision classifiers.

We deploy EZMODE on a 9,000 GB novel, real world video dataset. The dataset consists of continuous video taken of flowering plants in an outdoor setting with the goal of finding frames showing hummingbirds visiting individual flowers to feed on nectar. We first show that EZMODE is 130$\times$ more efficient than pretrained detectors at selecting rare events. We then show that EZMODE strongly outperforms detectors trained on both easy and hard examples (by up to 7$\times$) and that using temporal information also greatly aids in selection (by up to $1.8\times$).

2 Related Work

Selection. Work in the data analytics community has aimed to accelerate selection assuming access to an oracle. This work generally focuses on creating or leveraging a cheap approximation to the oracle [2, 3, 4, 5]. However, this work assumes a large enough sample of positive examples can easily be found (e.g., from pretrained labels or if the prevalence of positive examples is high) and uses these positive examples to train a cheap approximation. Our setting focuses on the case where it is difficult to obtain a sufficient number of positive samples to train a cheap approximation well.

Anomaly detection. The goal of anomaly detection is to find patterns in a dataset that do not conform to some expected behavior [6, 7, 8]. Our problem setting most closely resembles collective anomaly detection, where the goal is to find anomalous instances in a dataset. Similar to ranking models for anomaly detection, EZMODE attempts to find anomalous events through sorting the dataset [9, 10, 11]. In contrast, EZMODE does not aim to learn an optimal ranking. Rather, EZMODE iteratively uses a rank-ordering of the dataset coupled with problem structure to find rare events.

Co-training and active learning. Both co-training and active learning aim to train high quality classifiers. Co-training uses two (ideally independent) views of the same dataset, e.g., the textual data on a webpage and the hyperlink graph for classifying if a webpage is an academic home page or not [12]. Active learning iteratively chooses data points to label using strategies from uncertainty-based [13, 14], representation-based [15, 16, 17], and hybrid approaches [18, 19, 20].

In contrast to these approaches, EZMODE does not aim to train a high quality classifier. Instead, EZMODE leverages problem structure to select rare events iteratively (as in active learning) using independent views (as in co-training). Importantly, EZMODE does not aim to classify difficult positive examples and instead selects them via problem structure.

Algorithm 1 Pseudocode for EZMODE. EZMODE iteratively trains models and selects examples based on model confidence. Upon sampling a positive example, EZMODE will also recursively do distance-based sampling for nearby positive examples. SubsetAndTrain subsets by dropping hard examples and trains a model. DistanceSampling recursively samples by distance for positive examples. We omit these two functions for brevity.

function EZMODE($\mathcal{D}, \mathcal{O}$)
  $X_{\text{all}}, Y_{\text{all}} \leftarrow$ Init($\mathcal{D}, \mathcal{O}$) ▷ Init is usually done via random sampling
  $\mathcal{M} \leftarrow$ SubsetAndTrain($X_{\text{all}}, Y_{\text{all}}$) ▷ Train the initial model
  for Rare events remaining, $i = 1$ **do**
    $X_i, Y_i \leftarrow$ Select($\mathcal{D} \setminus X_{\text{all}}, \mathcal{O}, \mathcal{M}$)
    $X_{\text{all}} = X_{\text{all}} \cup X_i, Y_{\text{all}} = Y_{\text{all}} \cup Y_i$
    $\mathcal{M} \leftarrow$ SubsetAndTrain($X_{\text{all}}, Y_{\text{all}}$)
  return $\{(x, y) \in (X_{\text{all}}, Y_{\text{all}}) : y = 1\}$ ▷ Return all positive examples

function SELECT($\mathcal{D}, \mathcal{O}, \mathcal{M}$, $K = 1000, R = 9$)
  $\mathcal{S} \leftarrow \mathcal{M}(\mathcal{D})$ ▷ Generate confidence scores from the model
  $\mathcal{R} \leftarrow$ ArgSort($\mathcal{S}$) ▷ Sort by confidence
  $X = \emptyset, Y = \emptyset$
  for $k = 1, ..., K$ **do**
    $y_k = \mathcal{O}(\mathcal{R}(x_k))$
    $X = X \cup \{x_k\}, Y = Y \cup \{y_k\}$
    if $y_k = 1$ **then**
      $X_D, Y_D \leftarrow$ DistanceSampling(x_k)
      $X = X \cup X_D, Y = Y \cup Y_D$
  return X, Y

Data programming. Work on data programming aims to leverage rules to generate "weak labels" for the purpose of training high accuracy classifiers [21]. This work ranges from learning structures from unlabeled data to generating rules automatically [22, 23]. Similarly to active learning, EZMODE's goal is not to learn a high quality classifier, but to select data. We see leveraging data programming to selection as an exciting area of future work.

3 EZMODE

Problem statement. We are given a dataset $x \in \mathcal{D}$, an oracle $\mathcal{O}(x) \in \{0, 1\}$, and hardness $\mathcal{H}(x)$. Furthermore, we assume there is an undirected graph $\mathcal{G}$ over the dataset with weighted edges e_{ij}. We iteratively select *unique* data points $x_1, ..., x_T$ and we define the loss of selecting a point as $\ell(x_t) = 1 - \mathcal{O}(x_t)$. Define the loss over T steps of some decision algorithm H as $L_H^T = \sum_{t=1}^{T} \ell(x_t)$. The goal is to minimize the loss over some fixed time horizon T, i.e., find $\min_H L_H^T$.

EZMODE. EZMODE aims to minimize the loss via an iterative procedure outlined in Algorithm 1. Intuitively, EZMODE will train a classifier to select *easy* examples as defined by $\mathcal{H}(x)$ (e.g., as in Figure 1) and select data via classifier confidence. Upon finding a positive example, EZMODE will use the distance induced by $\mathcal{G}$ to label nearby points. After some number of steps, EZMODE will iterate from the beginning.

EZMODE contains several hyperparameters that must be tuned (e.g., the number of samples between retraining, the distance threshold). We leave the tuning of EZMODE to future work.

4 Experiments

We evaluate EZMODE on a novel, real-world, 9,000 GB field video dataset. We split the video into ∼8M two-second clips (approximately six months of continuous video). The goal is to find all clips containing bird visits. We used Faster R-CNN [24] as our base model and selected data using EZMODE. We take the maximum confidence across frames from Faster R-CNN as the score of a clip. The oracle was an expert human labeler and the distance between two clips is the time between them.

Model	Fraction needed	Precision @ 1000
R-CNN (MS-COCO, bird)	98%	10%
MegaDetector (animal)	99%	3%

Table 1: Two state-of-the-art object detection methods for finding birds. We show the fraction needed to find all the bird clips (discovered so far) and precision at the top 1,000 ranked clips. As shown, the oracle would have to label over 98% of the data when using pretrained models to find the rare events.

Oracle Calls	Precision (w/ Temporal Sampling)	Precision (rank only)
250	26.4%	23.6%
500	29.6%	18.2%
1000	25.8%	14.2%

Table 2: Precision at top 250, 500, and 1,000 for selection using problem structure and without using problem structure. Namely, we show sampling using temporal sampling and using confidence only. Using problem structure outperforms in all cases, by up to 11%. As expected (if the classifier is a monotone transformation of calibrated), precision decreases when using rank only.

Model	Precision @ 250 (rank only)	Precision @ 250 (w/ temporal sampling)
Only easy	23.6%	26.4%
Easy and hard	3.2%	19.6%

Table 3: Precision of the top 250 clips with a model trained on a) only easy and b) easy and hard examples. Easy and hard were determined by a human labeler As shown, the model trained with only easy examples outperforms the other model by over 7×.

To date, we have selected 7,451 clips of birds, or 56% of the total birds in the video ([45, 100]% 95th percentile CI). We report numbers below using the 7,451 clips.

Pre-trained detectors are insufficient. We first show that pretrained detectors are insufficient. We used a state-of-the-art Mask R-CNN (the most accurate model provided by Detectron2 [25]) trained on MS-COCO and a wildlife-specific detector provided by Microsoft [26]. We rank ordered the clips by maximum bird or animal confidence and computed the total number of clips necessary to select the bird clips we have discovered so far. We further measure the precision of the top ranked 1000 clips. As shown in Table 1, pretrained detectors would need to select up to 99% of the dataset to select the bird clips, which is 130× worse than EZMODE.

Leveraging problem structure is critical. We further show that leveraging problem structure is critical. In particular, temporal sampling allows EZMODE to discover difficult positive examples. To demonstrate this, we take our model with the largest number of training examples and compute the total number of clips necessary to select the bird clips we have discovered so far. Even our best performing model requires scanning 99% of the dataset to select the bird clips, compared to the 61,660 total samples needed to obtain the clips so far.

We further measured the precision at 1,000 clips when using and not using temporal sampling in Table 2. As shown, using temporal sampling increases precision by up to 11.6%.

Training on hard examples hurts performance. We show that training on hard examples hurts performance. We trained two models: 1) with 310 easy birds frames, 350 human frames, and 415 empty frames and 2) with 310 easy bird frames, 210 hard bird frames, 350 human frames, and 415 empty frames. Easy and hard were determined by a human labeler. We measured the precision in the top 250 clips ranked by confidence for both models. As shown in Table 3, the model trained with difficult examples is substantially worse than the model trained on only easy examples.

5 Conclusion

Selecting rare events in large datasets is of great interest. Unfortunately, standard methods (pretrained models and active learning) fail to leverage proximity structure. To address this, we introduce EZMODE, an iterative algorithm that leverages both active learning and proximity structure to select rare events in large datasets. We show that proximity structure is critical: not leveraging problem structure can be up to 130× more inefficient. Furthermore, we show that training on only easy examples can substantially outperform training on both easy and hard examples.

References

[1] Cody Coleman, Edward Chou, Sean Culatana, Peter Bailis, Alexander C Berg, Roshan Sumbaly, Matei Zaharia, and I Zeki Yalniz. Similarity search for efficient active learning and search of rare concepts. *arXiv preprint arXiv:2007.00077*, 2020.

[2] Daniel Kang, John Emmons, Firas Abuzaid, Peter Bailis, and Matei Zaharia. Noscope: optimizing neural network queries over video at scale. *PVLDB*, 10(11):1586–1597, 2017.

[3] Daniel Kang, Peter Bailis, and Matei Zaharia. Blazeit: Optimizing declarative aggregation and limit queries for neural network-based video analytics. *PVLDB*, 2019.

[4] Kevin Hsieh, Ganesh Ananthanarayanan, Peter Bodik, Paramvir Bahl, Matthai Philipose, Phillip B Gibbons, and Onur Mutlu. Focus: Querying large video datasets with low latency and low cost. *OSDI*, 2018.

[5] Michael R Anderson, Michael Cafarella, Thomas F Wenisch, and German Ros. Predicate optimization for a visual analytics database. *ICDE*, 2019.

[6] Thomas Schlegl, Philipp Seebock, Sebastian M. Waldstein, Ursula Schmidt-Erfurth, and Georg Langs. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. *Information Processing in Medical Imaging*, 10265(2):146–157, 2017.

[7] MohammadNoor Injadat, Fadi Salo, Ali Bou Nassif, Aleksander Essex, and Abdallah Shami. Bayesian optimization with machine learning algorithms towards anomaly detection. pages 1–6. IEEE Global Communication Conference, 2018.

[8] Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM Computer Survey*, 41(3):71–97, 2009.

[9] Xuanhui Wang, Nadav Golbandi, Michael Bendersky, Donald Metzler, and Mark Najork. Position bias estimation for unbiased learning to rank in personal search. In *WSDM*, pages 610–618, 2018.

[10] Tie-Yan Liu. Learning to rank from information retrieval. *Foundations and Trends in Information Retrieval*, 3(3):225–331, 2009.

[11] Xialei Liu, Joost van der Weijer, and Andrew D. Bagdanov. Leveraging unlabeled data for crowd counting by learning to rank. In *CVPR*, pages 7661–7669, 2018.

[12] Avrim Blum and Tom Mitchell. Combining labeled and unlabeled data with co-training. In *Proceedings of the eleventh annual conference on Computational learning theory*, pages 92–100, 1998.

[13] William H Beluch, Tim Genewein, Andreas Nürnberger, and Jan M Köhler. The power of ensembles for active learning in image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9368–9377, 2018.

[14] Keze Wang, Dongyu Zhang, Ya Li, Ruimao Zhang, and Liang Lin. Cost-effective active learning for deep image classification. *IEEE Transactions on Circuits and Systems for Video Technology*, 27(12):2591–2600, 2016.

[15] Samarth Sinha, Sayna Ebrahimi, and Trevor Darrell. Variational adversarial active learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5972–5981, 2019.

[16] Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. *arXiv preprint arXiv:1708.00489*, 2017.

[17] Dan Wang and Yi Shang. A new active labeling method for deep learning. In *2014 International joint conference on neural networks (IJCNN)*, pages 112–119. IEEE, 2014.

[18] Jordan T Ash, Chicheng Zhang, Akshay Krishnamurthy, John Langford, and Alekh Agarwal. Deep batch active learning by diverse, uncertain gradient lower bounds. *arXiv preprint arXiv:1906.03671*, 2019.

[19] Hieu T Nguyen and Arnold Smeulders. Active learning using pre-clustering. In *Proceedings of the twenty-first international conference on Machine learning*, page 79, 2004.

[20] Lin Yang, Yizhe Zhang, Jianxu Chen, Siyuan Zhang, and Danny Z Chen. Suggestive annotation: A deep active learning framework for biomedical image segmentation. In *International conference on medical image computing and computer-assisted intervention*, pages 399–407. Springer, 2017.

[21] Alexander J Ratner, Christopher M De Sa, Sen Wu, Daniel Selsam, and Christopher Ré. Data programming: Creating large training sets, quickly. *Advances in neural information processing systems*, 29:3567–3575, 2016.

[22] Stephen H Bach, Bryan He, Alexander Ratner, and Christopher Ré. Learning the structure of generative models without labeled data. In *International Conference on Machine Learning*, pages 273–282. PMLR, 2017.

[23] Paroma Varma and Christopher Ré. Snuba: Automating weak supervision to label training data. In *Proceedings of the VLDB Endowment. International Conference on Very Large Data Bases*, volume 12, page 223. NIH Public Access, 2018.

[24] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, 2015.

[25] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. `https://github.com/facebookresearch/detectron2`, 2019.

[26] Sara Beery, Dan Morris, and Siyu Yang. Efficient pipeline for camera trap image review. *arXiv preprint arXiv:1907.06772*, 2019.