RIPS 2016 Project:

Optimizing the Quality-Cost Tradeoff of Human Annotation

Industry Mentors:

Jen Bisignani
jenbisignani@google.com

Sarah Ouwayda
ouwayda@google.com

Alexander Vorontsov
vorontsov@google.com

Project overview

Google trains a number of binary classifiers to determine whether a page is about a certain topic. We train the classifier by showing it pages where the correct label (e.g., tobacco or not-tobacco) is known. The classifier looks at the features associated with these example pages, and learns how to label an unseen web page. In order to collect data to train the classifier, we collect a representative sample of web pages and have them manually annotated by human raters. This human-annotated data is then used as the gold standard, and is partitioned for use in training the classifier and in evaluating its performance.

Since the human-annotated data is used as the gold standard, the quality of the human annotation is paramount for training and evaluating a high-quality classifier. However, human raters can and do make mistakes. When collecting golden data, therefore, we can use several strategies to mitigate human error. Some available strategies are:

- Collecting a large amount of training data
- Using "premium" raters known to produce higher quality ratings on average
- Using trained internal raters
- Having the same web page rated by multiple raters
- Using mathematical error detection to clean the data before training and evaluating a classifier

Each of these strategies has its own costs and limitations. Thus, collecting a good dataset on a fixed budget will likely involve using some combination of these techniques. Additionally, there may be external constraints that limit how much we can rely on each of these techniques, such as availability of trained raters.

The goal of this project is to find the correct combination of rating strategies for the data used to train and evaluate the classifier, with the goal of achieving the highest possible classifier performance.

The primary dataset for the project will be either a set of labelled data provided by Google or a publicly available set.

Technical background

The first step in the project involves creating a simulation program that permutes a dataset to simulate human errors, then trains and evaluates a classifier to estimate the impact of the increase in errors. Once such a tool is in place, it can be used to simulate the use of different combinations of rating strategies on datasets with various characteristics meant to imitate real problems we solve at Google. By running many trials with various conditions, the team will be able to predict the best strategy specifying the number of pages rated, the proportion and choice of pages that get sent for triple-rating, and for a given dataset, budget, and distribution of rater quality. Additional details on rater quality, and availability, and budget will be available for the project.

Rater & Time Constraints

For this project, assume we have three types of raters:

- Grade A Raters: High quality raters
    - Accuracy: 90%
    - Monetary cost: x
    - Productivity: 0-8K/week
    - Max number of tasks 8K/week
- Grade B Raters:
    - Decent quality raters
        - Accuracy: 80%
        - Monetary cost 0.6*x
        - Productivity: 2K-30K ratings/week
    - Overflow raters
        - Accuracy: 60%
        - Monetary cost 0.6*x
        - Productivity: 5K-unlimited

For each classifier, we have a monetary budget of 15000*x and a time horizon of one week. Productivity varies depending on the task so optimal rater composition will have to vary with the productivity constraint.

Classifier training and evaluation

Typically, in training a classifier, human-labeled data is split into two sets: a larger "training" set and a smaller "holdout" set used for evaluation. The classifier is trained only on the training set data and then evaluated on the evaluation data.

We evaluate a classifier by measuring the proportion of time it correctly guesses the label for a previously unseen page, i.e. one from the holdout set. The instances that the classifier identifies as positive, and which the human raters have labelled positively are called True Positives. The instances that the classifier identifies as positive which the human rater have labelled as negative are called False Positives. Precision is the proportion of True Positives to all Positives.

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives}$$

The instances that the classifier identifies as negative and which the human raters have also identified as negative are called True Negatives. And the instances that the classifier identifies as negative and which the human raters have labeled as positive are called False Negatives. Recall is the proportion of True Positives to all Positives (which is True Positives + False Negatives).

$$Recall = \frac{True\ Positives}{True\ Positives + False\ Negatives}$$

The harmonic mean of precision and recall, also called the F-score, is commonly used as the single measure a classifier. A higher F-score is an indication of a better classifier.

$$Fscore = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$$

The goal of this project is to maximize a classifier’s F-score by creating the optimal training dataset of singly- and triply-rated instances given a fixed budget. Since this is prohibitively difficult to do analytically, we will do this by creating a number of training datasets with varying proportions of single- and triple-rated data, training classifiers on these data, and evaluating their performance.

We are interested in two evaluations. To simulate real-world outcomes, the classifiers will be evaluated on holdout data that will also be mutated per the simulation strategy. Additionally, we will evaluate the classifier on instances from the originally provided golden dataset that are known to be correct to get the true classifier performance. For a successful data generation strategy, the performance of the classifier on the mutated data should be similar to the performance of the classifier on the true data. Thus, some of the rating budget should be spent on getting a high quality data sample for evaluation.

Prerequisites

This project requires prior familiarity with basic computer science, including coding experience and familiarity with standard algorithms and data structures.Knowledge of a specific programming language is not required; however, candidates should have a working familiarity with at least one language that's appropriate for data mining and analysis, such as Python, R, C/C++, Java, or MATLAB. Each of these has excellent commercial and open-source libraries and packages that are appropriate for this project, such as scikit-learn (Python), the CMU C/Matlab Toolkit, and Weka (Java). Previous familiarity with these specific libraries is not required, as they can be learned as part of this project, but candidates should have prior experience with importing and manipulating data.

While previous experience with machine learning is helpful, it is not required for this project. Interest in machine learning, however, is very much a prerequisite.

Objectives

- Simulate the effect of human raters of varying quality on the training data.
- Use open source software to train and evaluate classifiers using different variations of the training data.
- Develop a model for predicting the optimal proportion of multiple-rated instances, distribution over different levels of rater quality, and number of raters involved for a given classification task, budget, and time constraint for the training and evaluation data.
- Model the marginal impact of additional time and monetary budget on the quality of the classifier.
- Evaluate the possibility of using more sophisticated techniques such as error detection to improve ratings without incurring additional costs.
- Summarize the results in the form of a written report and presentation to the Google LA office.
- *Stretch objective*: Create realistic synthetic datasets to mimic different conditions of the problem, and evaluate whether the resulting outcomes are transferrable to real data.

- *Stretch objective*: Model the impact of using a non-binary task to collect binary data (e.g. using a slider, using a 4-point scale, etc.) on the quality of the classifier.

Expectations

By the end of the project, the team should have a good understanding of how binary classification works, and of the importance of training and evaluation data quality and size in building and evaluating a classifier. Deliverables will include code used to simulate noise, code or protocol followed for evaluating trials, a written report outlining the methods and results, and a presentation at Google's offices in Venice.

Recommended Reading

Important readings

□ Beleites, C., Neugebauer, U., Bocklitz, T., Krafft, C., & Popp, J. (2013). Sample size planning for classification models. Analytica chimica acta, 760, 25-33.
http://arxiv.org/pdf/1211.1323v3.pdf

□ Hodge, Victoria J., and Jim Austin. "A survey of outlier detection methodologies." Artificial Intelligence Review 22.2 (2004): 85-126.
http://rd.springer.com/article/10.1007/s10462-004-4304-y

□ Tallon-Ballesteros, A.J.; Riquelme, J.C. "Deleting or keeping outliers for classifier training?", Nature and Biologically Inspired Computing (NaBIC), 2014 Sixth World Congress on, On page(s): 281 - 286 http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6921892

□ Weiss, G. M., & Provost, F. (2003). Learning when training data are costly: the effect of class distribution on tree induction. Journal of Artificial Intelligence Research, 315-354.
https://www.jair.org/media/1199/live-1199-2209-jair.pdf

Other Related Readings

Konidaris, Thomas, et al. "Keyword-guided word spotting in historical printed documents using synthetic data and user feedback." International Journal of Document Analysis and Recognition (IJDAR) 9.2-4 (2007): 167-177.
http://rd.springer.com/article/10.1007/s10032-007-0042-4#/page-1

Maaten, L., Chen, M., Tyree, S., & Weinberger, K. Q. (2013). Learning with marginalized corrupted features. In Proceedings of the 30th International Conference on Machine Learning (ICML-13) (pp. 410-418). http://jmlr.csail.mit.edu/proceedings/papers/v28/vandermaaten13.pdf

Smith M. R. and T. Martinez, "Improving classification accuracy by identifying and removing instances that should be misclassified. Proc. of the 2011 International Joint Conference on Neural Networks (UCNN 2011), IEEE, July 2011, pp. 2690-2697, doi: 10. 11 09/IJCNN. 20 11. 6033571. http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6033571

Smith, M.R. and T. Martinez "Reducing the Effects of Detrimental Instances", Machine Learning and Applications (ICMLA), 2014 13th International Conference on, On page(s): 183 - 188 http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7033112

Wei Q, Dunbrack RL Jr (2013) The Role of Balanced Training and Testing Data Sets for Binary Classifiers in Bioinformatics. PLoS ONE 8(7): e67863. doi:10.1371/journal.pone.0067863 http://www.plosone.org/article/fetchObject.action?uri=info:doi/10.1371/journal.pone.0067863&representation=PDF