

CS-GY 9223 A Prototype for Labeling Text Data

Haifeng Zhang
New York University
Brooklyn, NY

haifeng.zhang@nyu.edu

Abstract

Machine Learning has been a powerful tool in data analytics and been widely used in many areas. To train a machine learning model, especially for supervised learning, large amount of labeled data are demanded. While data labeling is time consuming and labor-intensive, the lack of labeled data has been a big obstacle for the adoption of machine learning techniques. This poses opportunity for computer science researchers to provide algorithms and tools to assist user labeling data efficiently. In this project, we propose a prototype to help users label text data in a more efficient way with active guidance from machine learning and visualization techniques.

1. Introduction

With the development of machine learning, large amount of labeled data are demanded to train machine learning models. The lack of labeled data has been the most significant barriers to adopt machine learning into real world industries.

This issue present opportunities for computer-human interaction and visualization community to design algorithms and tools to support the process of data labeling more efficiently and effectively. Together with unsupervised learning techniques, visualization can help user go through large collection of documents quickly, identify meaningful labels easily and interpret models transparently.

Text data is one of the most common data in real world. Typical text analysis includes topic modeling, semantic analysis and so on. In recent years, online advertising has been an increasingly important tool in political elections. Making these political advertisements more transparent to people will be helpful to keep the election free and fair. Thus online advertisements platforms has allowed third-parties auditors to access these public transparency data and enabled third-parties to detect malicious activities. In this work, we focus on labeling text data from Facebook political online advertisements.

A prototype of labeling text data is proposed in this work. The core idea is using unsupervised learning to recommend similar documents in a way that they can be labeled together and using visualization to check the effectiveness of unsupervised learning by checking the quality of these returned documents. This method is evaluated on a case study.

2. Related Work

2.1. Interactive Text Analytics

Michael et al. [3] presented an interactive visual analytics tool for building new dictionary features(semantically related groups of words) for text classification problems. Steven et al.[5] created a system, with the capability to recommend new items for existing clusters and appropriate clusters for items, for helping users sort large numbers of documents into clusters. Jina et al.[11] presented an interactive visualization tool that facilitated the discovery of prediction errors and previously unseen concepts through human-driven semantic data exploration.

2.2. Data Labeling System

Jurgen et al.[1] presented a visual active learning system that enables physicians to label the well-being state of patient histories suffering prostate cancer. Florian et al.[6] evaluated three methods that incorporate active learning to various degrees in order to reduce the labeling effort as well as to increase effectiveness. Jurgen et al.[2] proposed the “visual interactive labeling” process that unifies both the machine learning and visual interactive perspective. Julia et al.[10] developed labeling interface with semi-automatic approaches that are capable of solving the labeling task by clustering the image data unsupervised and presenting this ordered set to a user for manual labeling. Benjamin et al. [7] presented an inter-active learning methods, which extends active learning by integrating human experts’ background knowledge to greater extent, to facilitate video visual analytics.

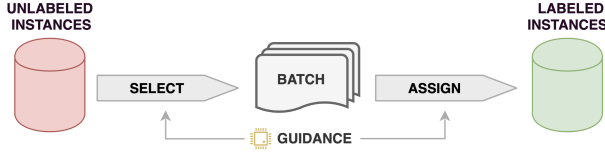


Figure 1: Workflow of Labeling Text data with the guidance of unsupervised learning and visualization

2.3. Political Advertisement Analysis

Laura et al.[8] presented a set of methodologies and performed a security analysis of Facebook’s U.S. Ad Library, which is a advertising transparency product of Facebook.

3. Methods

In order to labeling documents efficiently, the ideal situation is to label a batch of documents with same label together.¹ Starting from one selected document to be labeled, machine will recommend the similar documents. Here, we first train a Doc2Vec[9] model to embedding documents as vectors. Then we can find k nearest neighbors of the selected document in vector space. But the similar documents returned by KNN algorithm may not align well with user’s intention, in other words, some similar documents should not be assigned with the same label as the selected document. In order to interpret the similarity among documents, we designed a visualization of documents vectors in addition to pure text view to help users have a better understand of the documents vectors. After a batch of documents is defined well, next step is to refine the labels. In our system, user can first find a keyword from the documents. Then the machine will return its related words which are represented as a tree view. User can define and refine the label with the guidance of the tree view.

3.1. Numeric Representation of Documents

In natural language processing, building numeric representation is the first step. Here we adopt Doc2Vec method to embedding documents as vectors. This model is trained with 90,000 documents, including 40,832,130 raw words and 35,241 vocabularies, from the Facebook Advertisement Library. It takes 11916.9s to train the model for 10 epochs on a laptop with 2.9 GHz Quad-Core Intel Core i7 processor and 16GB memory. After training, every document is represented as a 100 dimension vector which can be used for further analysis.

3.2. Label Bootstrap

At the beginning of labeling data, few data are labeled, which means there are not enough samples to train a classifier that can classify documents into several batches. In

this stage, we design a label bootstrap mechanism. First, a single document is randomly selected from the whole corpus. Then, several similar documents is returned by machine based on cosine similarity. User can check these documents returned by machine and assign labels to them. In this way, several documents can be labeled together which is more efficient than labeling them one by one. After one round, users can select one document by themselves or machine will recommend one document to start another round. Repeat former process until there is enough samples to build the training set.

3.3. Label Propagation on Unlabeled Data

If there has been enough labeled data, classifier can be built to facilitate the process of labeling. Here we proposed an new classifier by adopting a label propagation algorithm[12] supposing that documents with the same label belong to the same class. First we define a distance matrix D ,

$$d_{ij} = \sqrt{(\vec{v}_i - \vec{v}_j)^2} \quad (1)$$

where \vec{v}_i is the numeric representation of i th document. Then we define a weight matrix W ,

$$w_{ij} = \frac{\exp(-\frac{d_{ij}^2}{\sigma^2})}{\sum_{k=1}^{l+u} \exp(-\frac{d_{ik}^2}{\sigma^2})} \quad (2)$$

where σ is a parameter to control the weights, l is the number of labeled data and u is the number of unlabeled data. We also define a $(l+u) * c$ label matrix B , where c is the number of classes. If the i th data is labeled, then assign $b_{ij} = 1$, which means it belongs to j th class, and all the other dimensions to 0. If the i th data is unlabeled, then randomly assign a probability distribution to b_i . Then we iteratively update matrix B as follows:

1. $B = WB$
2. Normalize every row of B
3. Change the rows of labeled data to original value

Repeat the upper procedure until B converges.

After B is converged, every unlabeled document get a predicted label. Based on the predicted label, previous unlabeled documents can be classified into different classes which can be treated as batches of documents to be labeled together.

3.4. Visualization

The label predicted by machine is not convincing without human confirmation. Reading these documents one by one is the most naive method to check the correctness of

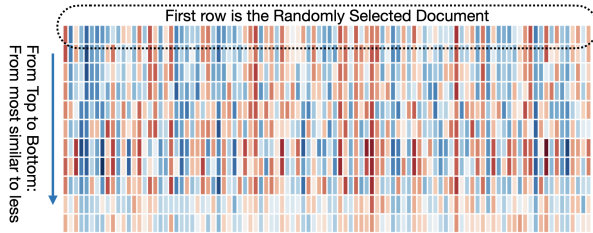


Figure 2: Vector View

the predicted label. However, reading one-by-one is time-consuming because its intrinsically a serial way of processing information. Comparing with reading in serial, visualization can make people capable of checking these documents in parallel. A vector view Fig.2 is designed to visualize the numeric representation of documents. In this view, every single row is one document and every single column is one dimension of document vectors. The first row is the selected document. From top to bottom is encoded with similarity, from the most similar document to less similar document. A diverging color scale is chosen to encode value from -1 to 1 as blue to red.

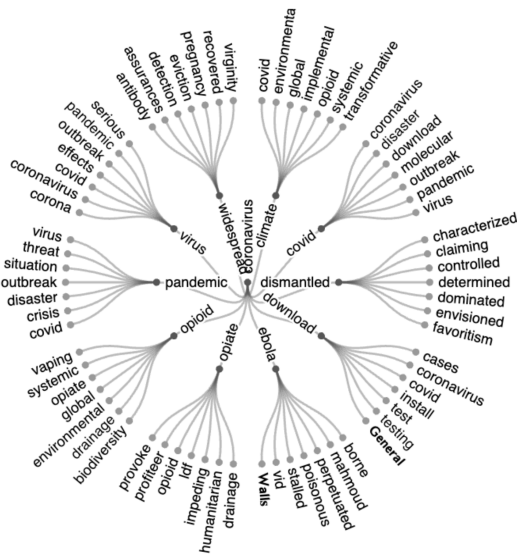



Figure 3: Tree view of Coronavirus


In the tree view, the root word is the keyword defined by user. The parent-children relation in the tree is defined as that the children words are the semantic similar words to the parent word. Following the tree structure, user can be reminded of the most suitable words to define labels. A circular form is adopted for saving display space.

4. Evaluation

4.1. Case Study

Start from one selected advertisement:


Hey There! We know you really wanted this beautiful Trump 2020 Law and Order Premium Sherpa Blanket and thought we would help you out! 

Click on shop now and get a matching 3x5" Single reverse flag with it. 

⌚ Limited Time Offer Only! Get Yours Today! ⌚

Shop Now => doperage.com/trump-law-and-order-blanket-with-fla


System will automatically return its similar documents:

Hey There! We know you really wanted this beautiful Trump 2020 Law and Order Premium Sherpa Blanket and thought we would help you out! 

Click on shop now and get a matching 3x5" Single reverse flag with it.

⌚ Limited Time Offer Only! Get Yours Today! ⌚

Shop Now => doperage.com/trump-law-and-order-blanket-with-flag

Hey There! We know you really wanted this beautiful Trump 2020 Law and Order Premium Sherpa Blanket and thought we would help you out! 

Click on shop now and get a matching 3x5" Single reverse flag with it. 🇺🇸

Limited Time Offer Only! Get Yours Today!

Shop Now => doperage.com/trump-law-and-order-blanket-with-flag

Hey There! We know you really wanted this beautiful Trump 2020 Law and Order Premium Sherpa Blanket and thought we would help you out! 🛒

Click on shop now and get a matching 3x5" Single reverse flag with it. 🇺🇸

Limited Time Offer Only! Get Yours Today!

Shop Now => doperage.com/trump-law-and-order-blanket-with-flag

Hey There! We know you really wanted this beautiful Betsy Ross Premium Sherpa Blanket and thought we would help you out! 🛒

Click on shop now and get a matching 3x5" Single reverse flag with it.

⌚ Limited Time Offer Only! Get Yours Today! ⌚

Shop Now => doperage.com/betsyross-blanket-with-flag

Bee Kind 🐝 Help us spread this message ❤️ New yellow sweatshirt available in the store. Grab yours before it's gone. Shop now!

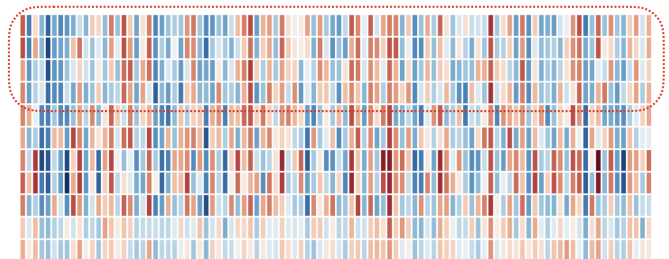
🐾 10% of our profits go to animal rescue and environmental organizations

🌱 Made from organic cotton and recycled materials

- 🌿 Made from Orga.
- ✌️ Ethically Made

👯 Order with your best friend and save on shipping!

We can see from the first 4 rows of vector view that they are almost the same, which means that they can be assigned with the same label.



Next step is define label for these documents, in order to distinguish them from others, we need to find the most specific word in the documents, here we select "doperage".

