Machine Learning - CS582

Group 2:
986191 - Boldkhuu Batbaatar
986264 - Batjargal Bayarsaikhan (Alex)
986159 - Unubold Tumenbayar

Sentiment Analysis

May 04, 2018

Project idea

We are going to develop a mobile app that tells which of one’s Facebook friends are pessimistic or optimistic using Sentiment analysis on their posts history. A user must log in to the app with their Facebook account and allow required access permissions. Then the app generates a list of pessimistic and optimistic friends of the user. Our app considers the friends who usually post positive things as optimistic, otherwise as pessimistic.

We are going to use various Machine learning algorithms from naive to advanced and are going to compare their results. The best algorithm (or combined) is going to be used for our mobile app.

Data set

We will be using "Sentiment Labelled Sentences Data Set" from UCI machine learning repository that was created for the Paper "From Group to Individual Labels using Deep Features", Kotzias et. al, KDD 2015.

It contains sentences labeled with positive (1) or negative sentiment (0). There are 500 positive and 500 negative sentences that come from IMDB, Amazon, and Yelp about movies, products, and restaurants.

Softwares

Machine learning algorithms:

- Gradient descent
- Naive Bayes
- Random Forest
- Recurrent Neural Networks
- Long Short Term Memory
- Supporting Vector Machine
- Deep learning
- Word2Vec

Programming language: Python, Javascript

Frameworks: Tensorflow, NodeJs, Ionic

Additional libraries: NumPy, Jupyter, matplotlib, Facebook API

Papers

1. Word2Vec

   Word2vec is a two-layer neural net that processes text. Its input is a text corpus and its output is a set of vectors: feature vectors for words in that corpus. It was introduced in 2013 by team of researchers led by Tomas Mikolov at Google - Read the paper from https://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality.pdf

   Additional reading from https://www.analyticsvidhya.com/blog/2017/06/word-embeddings-count-word2veec/.

2. Visualizing and Understanding Recurrent Networks

   Andrej Karpathy, Justin Johnson, Li Fei-Fei
   (Submitted on 5 Jun 2015 (v1), last revised 17 Nov 2015 (this version, v2))

   Recurrent Neural Networks (RNNs), and specifically a variant with Long Short-Term Memory (LSTM), are enjoying renewed interest as a result of successful applications in a wide range of machine learning problems that involve sequential data. However, while LSTMs provide exceptional results in practice, the source of their performance and their limitations remain rather poorly understood. Using character-level language models as an interpretable testbed, we aim to bridge this gap by providing an analysis of their representations, predictions and error types. In particular, our experiments reveal the existence of interpretable cells that keep track of long-range dependencies such as line lengths, quotes and brackets.

   https://arxiv.org/pdf/1506.02078.pdf

3. Long Short Term Memory

Long Short Term Memory networks – usually just called “LSTMs” – are a special kind of RNN, capable of learning long-term dependencies. They were introduced by Sepp Hochreiter and Jürgen Schmidhuber in 1997.

http://www.bioinf.jku.at/publications/older/2604.pdf

Additional reading from http://colah.github.io/posts/2015-08-Understanding-LSTMs/.

4. Automated Hyperparameter Tuning for Effective Machine Learning

Patrick Koch, Brett Wujek, Oleg Golovidov, and Steven Gardner
SAS Institute Inc, 2017

Machine learning predictive modeling algorithms are governed by “hyperparameters” that have no clear defaults agreeable to a wide range of applications. The depth of a decision tree, number of trees in a forest, number of hidden layers and neurons in each layer in a neural network, and degree of regularization to prevent overfitting are a few examples of quantities that must be prescribed for these algorithms. Not only do ideal settings for the hyperparameters dictate the performance of the training process, but more importantly they govern the quality of the resulting predictive models. Recent efforts to move from a manual or random adjustment of these parameters include rough grid search and intelligent numerical optimization strategies.

https://pdfs.semanticscholar.org/ce60/a2a90be9ba9088f91b1bd51a2c1cabade1ae.pdf

Additional reading from
https://papers.nips.cc/paper/4443-algorithms-for-hyper-parameter-optimization.pdf.

5. An overview of gradient descent optimization algorithms

Sebastian Ruder

(Submitted on 15 Sep 2016 (v1), last revised 15 Jun 2017 (this version, v2))

Gradient descent optimization algorithms, while increasingly popular, are often used as black-box optimizers, as practical explanations of their strengths and weaknesses are hard to come by. This article aims to provide the reader with intuitions with regard to the behaviour of different algorithms that will allow her to put them to use. In the course of this overview, we look at different variants of gradient descent, summarize challenges, introduce the most common optimization algorithms, review architectures in a parallel and distributed setting, and investigate additional strategies for optimizing gradient descent.

https://arxiv.org/pdf/1609.04747.pdf

Additional reading from http://ruder.io/optimizing-gradient-descent/.

Expected result

Various machine learning algorithms will be used for Sentiment Analysis (Positive or Negative). The best resulting algorithm will be chosen for a mobile app that generates the list of the optimistic and pessimistic friends of the user on Facebook.

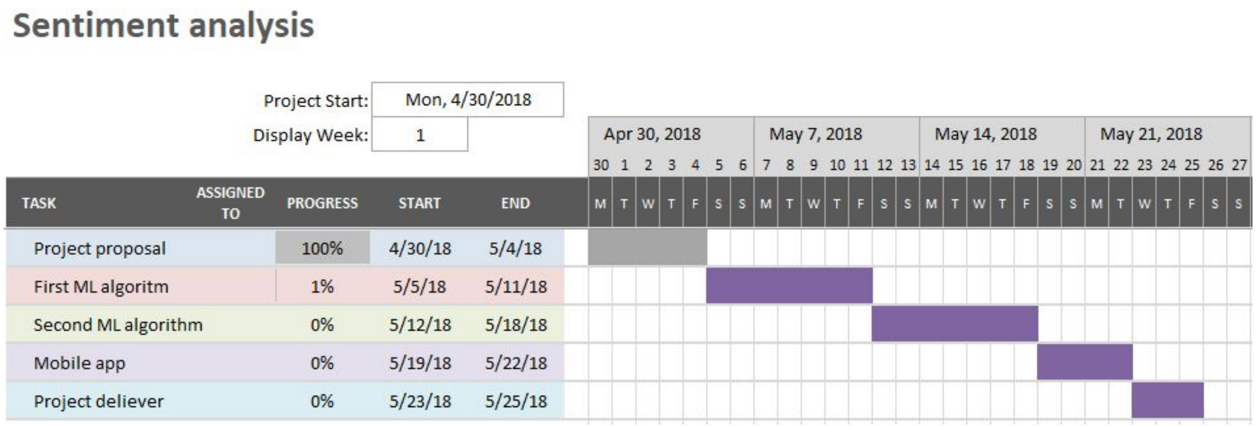
- Machine Learning algorithms comparisons
- A Mobile application that uses the best resulting algorithm

Milestone

By 11th of May, we will have done some ML algorithms that we did not cover or did not cover in depth in class such as Recurrent Neural Networks, Swarm Intelligence, graphical models, Hidden Markov Model, and Deep learning for our project.

By 18th of May, we will have implemented Machine Learning techniques that we learned in class and compared the 2 algorithms.

By 22nd of May, we will have finished developing our simple mobile phone app. Then we will start preparing presentation and preparing code, snippets to submit.



Sentiment analysis

Project Start: Mon, 4/30/2018

Display Week: 1

TASK	ASSIGNED TO	PROGRESS	START	END
Project proposal		100%	4/30/18	5/4/18
First ML algoritm		1%	5/5/18	5/11/18
Second ML algorithm		0%	5/12/18	5/18/18
Mobile app		0%	5/19/18	5/22/18
Project deliever		0%	5/23/18	5/25/18