Social Media Monitoring by Using Data Mining

Fuat Basık

Presentation Plan

- Introduction
- Motivation
- Stream Processing
- Data Set
- Turkish Language
- Pre Processing and Stemming
- Term Frequency and Inverse Document Frequency
- Support Vector Machines
- Demo
- Conclusion

Introduction

- Different than the discussion on the class.
  - But includes applications of what we learned so far.
- Business Intelligence applications use old data.
  - It is like using only rearview mirror to drive a car.
- Social media is a bridge that companies can reach to customers directly.
- Statistics about Social Media
  - Facebook reached 1 billion customers recently.
  - By the end of 2010, Twitter gets 400 million tweets per day.
  - Turkey is the 11^{th} country that contributes Twitter most.
- Big Numbers, Big Data, Big Market, Big Problem.

Motivation

- Turkish is an agglutinating language.
- There are only a few numbers of research about Turkish Language.
- Companies need a way to extract knowledge from social media.
- Twitter data fit perfectly stream processing applications.
- Creating an application that is able to process streaming data.

Motivation

- Stream Processing
  - Process the data while it is flowing into the system.
  - Before inserting to database.
  - No I/O cost.
  - Real time analysis.
  - Good for the Business Intelligence applications, Fraud Detection, Image processing applications.

Data Set

- For Learning Phase
  - Data collected and written to a file by using Twitter4J.
  - Labeled as negative or positive.
  - 2250 Instances
    - 1650 Negative Labeled.
    - 600 Positive Labeled.
- For Application Phase
  - Live data flow from Twitter itself.
  - Keyword based search.
  - Using Twitter4J, Twitter API for Java.
  - Example Tweet:
    - avea gibi hiçbir yerde çekmeyen baska bir hat daha yoktur heralde.

Turkish Language

- Turkish Language is one of the Morphologically Rich Languages.
- It is an agglutinating language.
- Harder to process.
- Stemming algorithm should remove suffixes from the root.
- For example:
  - Söyleyemedim (Söy-le-ye-me-dim) vs Söylemedim (Söy-le-me-dim).
  - Not being able to.

Pre Processing and Stemming

- Stop word removal.
  - Words that do not have meaning.
  - Domain specific words.
- Repeating letter removal.
- Smiley replacing.
  - Not done yet.
  - Replace ☺ with Positive and ☹ with negative.
- Zemberek, Natural Language Processing Tool.
  - Official spell checker of Turkish Applications of Open Office.
  - Spell checking, word suggestions, separating the root and suffixes.

Pre Processing and Stemming

- Example:
  - Go over the same tweet:
    - avea gibi hiçbir yerde çekmeyen baska bir hat daha yoktur heralde.
    - After Pre Process:
    - hicbir yer _cek baska hat yok herhalde
- After pre processing, each word will be evaluated separately.
- Then, TF-IDF Transformation will be applied.

Term Frequency

- Term Frequency is the count of occurrences of a word in a given class.
- How often a term occurs.
- Normalized.

$$\mathrm{tf}(t,d) = \frac{\mathrm{f}(t,d)}{\max\{\mathrm{f}(w,d) : w \in d\}}$$

Inverse Document Frequency

- Counts the occurrences of a word in the not given classes.
- For example count of word “Harika” in positive labeled tweets are TF and negative labeled tweets are IDF.
- Formulization

$$\operatorname{idf}(t, D) = \log \frac{|D|}{|\{d \in D : t \in d\}|}$$

TF-IDF

- Helps to find words that are occurring frequently in a class and not frequently in other classes.
- In our case we have two classes: positive and negative.
- We try to find most negative words by applying TF-IDF.
- Each word becomes an id + a frequency value.
- TF * IDF

Support Vector Machines

- A collection of features called an instance.
- Includes a class value, and list of features.
- Each feature is a word but with TF-IDF transform.
- Ex:
  - {0 ,136 2.500744,413 2.137636,427 3.627779,436 3.688091,890 2.208427,897 1.970956}

Support Vector Machines

- Correctly Classified Instances 1772 78.7556 %
- Incorrectly Classified Instances 478 21.2444 %
- Kappa statistic 0.4551
- Mean absolute error 0.2455
- Root mean squared error 0.4089
- Relative absolute error 62.7583 %
- Root relative squared error 92.474 %
- Total Number of Instances 2250

- === Confusion Matrix ===

- a b <-- classified as
- 358 242 | a = positive
- 236 1414 | b = negative

Demo & Conclusion

- Please send a tweet that contains “cs553” in it.

References

- 1. Cnet News, Dan Farber, June 6 2012, September 2012
- 2. Nathan Marz, Storm Project, http://storm-project.net
- 3. Semiocast publications, January 31 2012 Paris, France, September 10 2012.
- 4. Zemberek Natural Language Processing Tool, http://code.google.com/p/zemberek/
- 5. TF-IDF: http://en.wikipedia.org/wiki/Tf%E2%80%93idf