Project Proposal

Will your blog appeal to the world? : An inference and estimation of popularity of a blog post based on stylometric analysis of its content

Project team:
1) Manisha Kamal - 108312148
2) Nilesh Poddar - 108393824
3) Purvesh Sahoo - 108231506

Introduction

A **blog** is a discussion or informational site published on the World Wide Web and consisting of discrete entries ("posts") typically displayed in reverse chronological order (the most recent post appears first) [2]. Many blogs provide commentary on a particular subject; others function as more personal online diaries; others function more as online brand advertising of a particular individual or company. A typical blog combines text, images, and links to other blogs, Web pages, and other media related to its topic. The ability of readers to leave comments in an interactive format is an important contribution to the popularity of many blogs.

Why blogs?

There is nothing on the internet more powerful than being able to *project your thoughts and ideas through blogging*. Blogs have become one of the most stable ways to not only attract people to your business or yourself but also **a channel of valuable information to the world**.

Types:

There are many different types of blogs, differing not only in the type of content, but also in the way that content is delivered or written.

1. Personal blogs
2. Corporate and Organizational blogs
3. By genre
4. By media type
5. By device
6. Reverse blog

Problem Statement

We intend to classify the blog posts as to whether they are popular or not. The domain that we intend to target is Technology blogs.

The motivation lies in the fact that this area has a lot of scope in success prediction but unfortunately, little research has been done in this area. Also, this kind of analysis will help a novice blogger to predict if his blog post will be popular or not. Finally, such analysis can also help in the marketing world, wherein new products are being launched every day, to decide if the product will be a success or not based on the information provided to the users.

Data Collection & Categorization

Data collection can be divided in 2 stages:

1 Getting a list of blog posts (www.reddit.com/r/technology) by sorting it based on score for a particular time range.
2 Get the content of the top "X" and bottom "X" of the blog posts from the above list.
3. We will use the reddit search API:

The scores and other parameters can be fetched in JSON format by using the search API of reddit. We will be fetching the posts within a specific timeframe. The posts are sorted based on the score value. The detailed explanation about the scoring mechanism is given in the section that follows. Since we will get a list of the top posts, we are going to train our model based on the top posts as popular posts. The posts have an associated score, and a time period with it. We will normalize the post's score per day, based on the score and the time period of the particular post. For example, if there was a post made 3 days ago, it has an upvote score of X, and we have a timeframe of a week; we score the post as X / 3. We do that for each post in this time frame and get the number of upvotes per day for each post. We will categorize the posts based on this method.

Experiments/Hypotheses to explore

We plan to conduct the following experiments:

1 Explore the effects of different time periods on the result.
2 Adding customized features like the "Title of the post" to see if the results are better.

Hardware and Software Requirements

- Tools : Weka
- Classifier : Naive Bayes or LibSVM
- Packages: OpenNLP, Stanford Parser

- OS : Windows, Ubuntu
- Language: Java

Implementation

The implementation can be divided into 3 stages:

1 Collection and classification of training data. The blog post needs to be classified as popular or unpopular. This classification is done as follows:

Let, U = no. of upvotes
D = no. of downvotes
T = timestamp (in days, when the blog post was created)

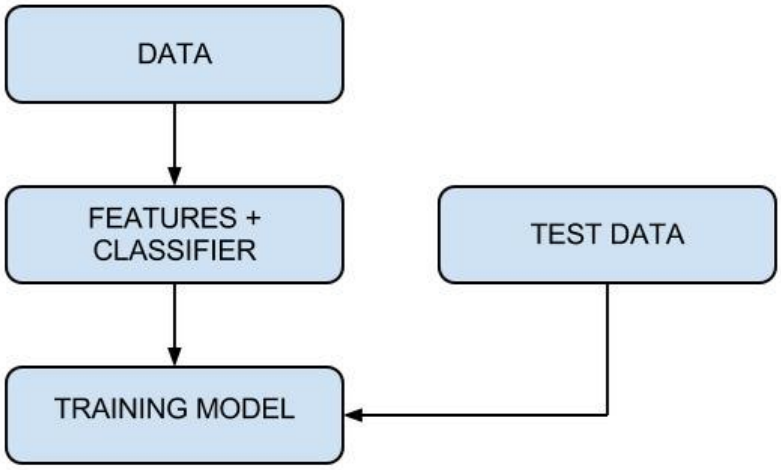
The metric for threshold will be the normalized per day score. It is calculated as follows:

Score/day = (U - D) / T

2 Train the data using a set of features on one or more classifiers after appropriate preprocessing and normalization using a tool such as weka. Some of the features are as follows:

a unigram, bigram, trigram language models.
b POS
c POS +bigram/trigram
d Other customized features

3 Test the data on the training model created above and report the accuracy.



Analysis

We plan to do the analysis in two parts:

1. Get results with high accuracy.
2. Linguistic analysis.

References

1. http://www.cs.cornell.edu/~cristian/memorability_files/memorability.pdf
2. http://en.wikipedia.org/wiki/Blog
3. https://developers.google.com/custom-search/v1/overview
4. https://dl.acm.org/citation.cfm?id=1937087
5. www.fbe.hku.hk/~mchau/papers/Blog_WITS2009.pdf