Universität Ulm
The Faculty of Mathematics
and Economics

Determining Information Quality in Social Networks using Message Classification Techniques

Master Thesis

in Management and Economics

Leonid Edelmann

July 18, 2017

Supervision

Prof. Dr. Mathias Klier
Prof. Dr. Leo Brecht

Table of Contents

List of Figures

1 Introduction

1.1 Motivation

placeholder

1.2 Current State of Research

Literary Overview

2 Basic Concepts

2.1 Machine Learning

A sub-branch of computer science that rose to prominence and started evolving during the 1950s as part of research in the field of artificial intelligence. Machine learning refers the development of algorithms, which allow computers to learn from presented examples. The computer is thereafter supposed to learn from its collected experience and automate the process of solving similar tasks. This process is referred to by term *training*.

One official definition as coined by Tom M. Mitchell [4] is "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E."

The most common use of machine learning algorithms is the analysis of real-world data for certain tasks, when a concrete programmer-written application would be ineffective in solving. Such is the case for example with problems, which a human would be able to solve, but would not be able to determine the rules for solving explicitly. Or alternatively, where the rules are not constant, but rather evolving as time progresses. The purpose of teaching a machine to solve such tasks, is modeling, prediction or detection of details or certainties about the real world.

Vivid examples of real world uses are speech recognition as used in cellphones or in call routing system as well as visual recognition, where and algorithm is trained to recognize graphic patterns and used in medicine or in hand written text recognition.

2.2 Text Mining

Text mining refers to the practice of extracting facts out of raw metadata, which comes in the form of text corpora or other unstructured data. Initially the data must be structured into a form compatible for its statistical analysis. This includes but not limited to, segmenting the text to more basic building blocks such as paragraphs or sentences. This practice is known as **stemming**. Cleaning up the data by removing non-informative words, which serve a grammatical role in human language, but tend to be of no use for language processing done by machines. In the next phase the words are normalized to their **stem** by removing inflection which modifies the word's tense, case number and other grammatical properties. In the professional nomenclature, this is referred to as **stemming** or **lemmatizing**.

Methods used in text mining involve statistical pattern recognition,tagging-annotation, information extraction and frequency analysis. The end goal of text mining is namely, the production of qualitative information out of raw text, often automatically using machine learning.

Most of the usage of text mining methods in this work will be with the use of the NLTK module for the Python programming language. NLTK is a suite of libraries developed in the Department of Computer and Information Science at the University of Pennsylvania for Natural Language Processing [1].

2.3 Unstructured data

Data which is derived from the Internet, namely social networks is oftentimes unstructured. Unstructured data refers to information which is not organized according to a unison manner or in a data model, that fits the structure in which the data is to be utilized. This information tend to be heterogeneous in its data types and may contain texts, number, dates as well as media. Additionally, the data can vary highly in its integrity, recurrently having missing or only partial data.

In our case, the data comes in the form of Tweets originating from the Twitter servers. Tweets arrive packaged in the form JSON objects. These objects have no standard morphology, and many of the Tweets' fields are subject to variation. The most common approaches to dealing with such data are either restructuring it to a new data-model or conducting a textual analysis aimed at recognizing patterns in the information. Text Mining and Natural Language Processing are two prevailing mechanisms employed for this purpose. More on those in Chapter 3, *Collecting the Data*.

2.4 Classification

The correct assignment of (often unstructured) data to a category out of a set of predetermined categories, in the case of supervised learning or creation of new categories which best segment the data, in case of unsupervised learning. Supervised and Unsupervised Learning, are different approaches in Machine Learning, where the former requires continues input from the user whereas the latter is more autonomous.

Supervised Learning

Unsupervised Learning In the case of Unsupervised Learning, the task of classification is also commonly referred to as *Clustering*. during the process of *Clustering*, the Machine does not require the input (training-) data to be **labeled**, or classified anteriorly to predetermined categories. Instead, the *Clustering* algorithm is usually given the number of categories, and the algorithm has to fracture the data according to what it observes to be distinguishing features of the each category. Two very common models are Hierarchical Cluster Analysis (HCA), K-Means and Mean Shift. elaborate ?

2.5 Information Quality

Social Networks incorporate the interactions of millions of individuals and organizations, thus allowing such an information flow to be classified as Big Data. Such data could be confined to the the widely acknowledged 5 characteristics of Big Data, also knowns as the 5 **V**'s[2]. Of interest here is the data's *Veracity*, referring to the trustworthiness and statistical reliability of said facts, originating from a plethora of sources and presenting little to no accountability for its correctness.

The questionable quality of such information makes basing critical business decisions on it, risky at best. One solution proposed to overcome these shortcomings would be to attach alongside the information quality metrics, which would describe its correctness, completeness and topicality as proposed by Klier and Heinrich(2016)[3]

3 Methodology

The approach I undertake in this study is similar to previous works in the field. Initially, data is gathered on a given theme. In the case of my research, the main query I wish to study would regard Tweets relating to the subject of E-Commerce platforms.

On of the motives of this research is to observe whether information originating from social media can be evaluated in real-time. With this consideration in mind, the data should bear as much resemblance to a live information torrent on Tweeter as possible. Acknowledging this consideration, the raw data in gathered in the form of Tweets originating from the Twitter databases. I collect the data synchronously, as it is intercepted by the servers. The *Streaming* Application Programming Interface (API)[7] is usually implemented in such use-cases. Using the *Streaming* API, a connection is established to the servers, which captures a narrow stream (about 15%) of all Tweets relevant to a given search term.

The captured data, must then be restructured to fit the data-model of this study, namely appropriate input for Machine Learning algorithms. This includes a preliminary analysis of the data and removal of incomplete observations. The data structure in which Tweets are stored allow for a dynamic non-standard structure, which in turn means they vary in properties. Therefore some Tweets must be adapted to conform to a unitary pattern. It is also quite common for Tweets to retrospectively be removed, either by themselves or by Twitter moderators. Such Tweets cannot be analyzed, since they are no longer available on-line. The process of gathering data and cleansing it is discussed in Part [4.1].

The next stage entails representing the data in a form which is suitable for Machine Learning Algorithms. For this purpose, I build *features*[1] for each observation. *Features* are unique properties of the data, which are used to describe it. Different types of *Features* are used in accordance with the Learning approach undertaken. These approaches we discuss in Part [4.2].

These features are then analyzed for consistency and correctness. Subsequently, the features are passed into different Machine Learning algorithms, in order to *train*[1] them. *Training* is the process of deducing the decision rules for classifying the data into one of the categories. This deduction is based on the information the algorithm draws from the input data. Afterwards, the empirical success of these different algorithms will be statistically measured the summarized. Additionally, implications are to be drawn about other use-cases, which are out of the scope of this research.

[1]Machine Learning nomenclature

4 Application

4.1 Information Procurement

The main theme of the collected data would concentrate around the topic of Internet sales platforms also known as E-Commerce. Several aspects make the topic beneficial for this research. Firstly, the very nature of E-Commerce. By nature of E-Commerce Platforms, I refer to the fact, that such companies are almost exclusively web-based. It is therefore probable that most of the company's marketing efforts as well as overall news relating to such a company, would circulate first and foremost in the Internet. Secondly, having all company-relevant news attainable foremost from the web, means that such news would seep to social media faster in comparison to news, which are usually covered initially by traditional media such as television and Newspapers.

Search Terms

The data was collected in the form of relevant Tweets from the Twitter Stream API. A Tweet would be considered relevant if it contained a search parameter related contextually to E-Commerce. The initial efforts were concentrated around the web-store Amazon. Amazon appears to the most fruitful search parameter in terms of the quantity of Tweets relating to it. Additional search words that were tested were **Alibaba, Zalando** and **Groupon**. The widespread mention of Amazon in Tweets is however somewhat over-inflated due to the extensive use of Amazon gift cards. Amazon gift-cards have become prominent due their variety of uses. A few examples of common practices involving Amazon gift cards are rewarding users for services, such as polls and questionnaires, enticing people to take part in events or groups, and being offered as general rewards in competitions and games. The plethora of uses, facilitates Amazon gift cards to be viewed as a sort of pseudo-currency in the Internet. In turn this means, that Amazon could be mentioned in a Tweet, despite the context only indicating the Gift-card and being completely unassociated to the E-Commerce platform whatsoever.

Collecting the Data

The gathering of Tweets was executed using a program written by me in the Python programming language. The main module being used in the program was a Twitter Streaming API called Tweepy. Tweepy is an open source interface, which allows communicating with the Twitters servers and sending queries requesting specific information from Twitter's databases. The interface allows for two main type of queries, Rest and Streaming. The former allows looking up information posted on Twitter in the past whereas the latter, as the name implies connects to an active data stream containing a narrowed down flow of Tweets being actively

published by Twitter users. Both types of API's are being offered for free to a certain extent, whereas almost unrestricted versions of the same API are offered as a proprietary fee-based product of Twitter. The free version of the REST API is restricted to only looking up Tweets posted in the last two to three weeks. And the gratis version of the Streaming API is restricted to a fire-hose narrowed down to about 15% of the total Bandwidth of all current Tweets. The Tweets from the Twitter servers come in form of JSON strings, which allows for embedding other JSON objects in them, which allows for multi-level storage of Tweet properties. For example, one of the JSON objects integrated in each Tweet JSON object is the USER object for the Tweet-poster. The USER object in turn contains all data publicly available in Twitter about a Twitter account such as, location, date of registration, homepage etc. An additional object of interest is the ENTITIES JSON object, which contains all outside references from the Tweet's text such as, URLs, Multimedia, References to other Tweets or other users. This structure greatly eases the construction an analysis of a Tweet and its features, since most of the necessary data is available from the Tweet self and no further queries about the Tweets and its posting-user are necessary.

elaborate about stages of data collection:

1. Initial data: full of duplicates
2. Filtered data - fewer duplicates and Spam, but still rather very one-subject-centered. Leads to overfitting when classifying
3. still untried - collect up to x (400) Tweets per day, for a duration of 2 weeks

Data cleansing

It was observed that numerous Tweets were being posted more than once and in several occurrences even hundreds of times. These duplicates were being primarily posted by bots, as was evident from a short observation of user profiles belonging the Tweets original posters. Evidently, additional effort was being made by the programmers of the bots to try and mask them by slightly altering the content of the Tweets, or the user account. This was usually done by changing or adding characters to the text, which carry no lingual significance in themselves. Moreover, in furtherance of increasing the bots' credibility as an actual people (Fig. 1), often times entire nets of such bot could be observed, wherein the bots would maintain friendship and following connections among themselves. This in turn, further adding to each of them having a multiplicity of friends and followers contributing to their veil of disguise as real human users of Tweeter. Upon closer observation such accounts reveal their true essence, since most of the content propagated by them is commercial in nature and is repeated verbatim time and again across many of the related accounts *followers* and *friends*, it would be safe to assume that no actual people are behind them.

Figure 1: Twitter accounts, which present themselves as actual people

Several precautions were undertaken to try and filter out such bots. A passive precaution which was implemented, was blacklisting users, which had priorly been observed posting Tweets, which were verbatim copies of other Tweets within the same query. The suspected users were added to a suspect database and content originating from them was ignored in future queries. Another action made with the same purpose in mind was a retrospective cleanup of the collected Tweets, based on their content similarity. After closing a collection query, a maximum similarity measure for each Tweet in relation to all other recorded Tweets was calculated using a simple . Following, Tweets which were found to have a maximal similarity score to other previously captured Tweets higher than a predetermined threshold were classified as non-unique copies and were disregarded. As a measure of similarity, the Sørensen-Dice coefficient[6] (Fig. 2) was implemented using the *SequenceMatcher.ratio()* function from the difflib Python module also try Levenshtein . A round of cleanup using this procedure would usually reduce a data set by from one quarter and up to one half of its original size.

$$QS_{XY} = \frac{2|X \cap Y|}{|X| + |Y|}, \qquad QS_{XY} \in [0:1]$$

* $|X|$ and $|Y|$ are the numbers of elements in given Tweets X and Y accordingly
** QS ranges from 0 (completely different) to 1 (identical)

Figure 2: Sørensen-Dice coefficient

4.2 Building the Datasets

Once a list of labeled Tweets is obtained, the next stage is constructing a feature-set to be later passed on as input for training an ML Classifier. The features contained if the feature sets describe certain aspects and characteristics of a Tweet and its owner. Two main approaches to feature sets are found in the literature, *Word-Based* also often called *Bag-of-Words* as can be observed in **insert citation** and *Descriptive* **insert citation**.

4.2.1 Word-Based Features

The former approach simply converts the entire text corpus to a frequency charts of all the words contained within. Words are then selected to act as features in incoming data, which is to be classified. The features are hence a variable list (usually of several thousands in length), where each variable is a boolean representation, indicating the presence or absence of a certain word. Usually the words undergo preprocessing as is common in Natural Language Processing prior to being used as features. The corpora are segmented to lists of words, often omitting articles, proposition and punctuation. Such grammatical structures are critical in human speech and writing ind order to convey ones meaning clearly and explicitly, however for the purposes of more ambiguous classification as in our case, such nuances are avoided for the sake of simplicity. Words are then *stemmed* or *lemmatized*, meaning their are reverted to their grammatical stem - dropping all prefixes and suffixes. This eases the enumeration of words, since it is preferable that the same words in different inclinations would be counted as the same. For example, the word pair *eating* and *ate* would be reverted to their stem *eat*, as well as *apple* and *apples* would be considered as one and the same. Finally, words would be assigned their part of speech (noun, verb, adjective ..) and could be either ignored or incorporated into the feature set, according to the conceived importance of a given part of speech.

4.2.2 Descriptive Features

The latter approach mentioned is based of more generalized view of the Tweet, instead of concentrating on the actual textual content. Descriptive features are aimed at describing the Tweet implicit properties, such as attitude, sentiment, seriousness and trustworthiness. These features detect presence of different symbols, their frequency and consecutiveness. Additionally, unlike the Web-Based approach, non-textual objects such as multimedia, links and mentions of other users and Tweets are also taken into account. Furthermore, features of a Tweets owner are included alongside. Since Tweeters API provides a complete user profile incorporated inside the Tweet data object itself, constructing features describing the user is done simultaneously to features describing the content of the Tweet itself. This approach might be viewed as an *indirect*

one, since less obvious properties of the Tweet are used to characterize it.

Descriptive features could be segregated into three distinctive groups. The first will be referred to as text-based features. As the name suggests, the features will mostly denote the presence or lack of specific characters such as emoticons and signs in the Tweets text. Whether a Tweet contains combinations or sequences of certain symbols as well as ratios defining the text also befit this category.

The second tier of features describe any special *Entities* (Tweeter's nomenclature) contained within a Tweet. *Entities* refer to non-textual contents of a Tweet such as media (in form of pictures, sound or videos), URL's linking to external websites, Mentions or ReTweets (Referring to other Tweets or to Tweeter user profiles) and finally Hashtags. A word or phrase preceded by the Hashtag symbol # indicate the association of web content (such as a Tweet or other micro-blogging post) to a specific theme such as an event, news, gossip or any other tidbit [8]. Hashtags are used primarily to simplify looking up Tweets or other social media content by technically associating them with the *hashtagged* topic.

The third tier - subject

4.3 Training Classifier

The purpose of *training* is to create Classifiers, which automatically recognize and *label* new observations. The intrinsic decision algorithm, through which the Classifier will decide how to classify a novel observation, usually remain hidden and operate as a black-box of sorts. Since the decision rules could be numerous and far from intuitive for human readers. This is especially the case, when said algorithms are convolutional. Convolutional machine learning schemes, such as Deep Neural Networks, may incorporate numerous stages of parameter construction which renders them practically incomprehensible to human users. The actual implementation of all the algorithms would be programmed in the Python programming language and will be primarily making use of the scikit-learn[5] module.

The data used for the purpose of this study consolidates 12.520 unique Tweets. For the purpose of reaching robust results, the *Hold-out Method* is used as follows. For each instance of *Algorithm training* the entire data corpus is split into two parts - a training set and testing set. The training set would usually be allocated the larger portion of the data (about 70%) and would be used, as the name suggests for training the Classifiers. The rest of the data (about 30%) would be used for testing the Classifiers accuracy. During each such training-testing session, the data corpus is shuffled. This in turn means, that the training and testing sets constantly differ. This process of splitting, training and testing using different data in each iteration should produce statistically significant results. This method of constantly splitting the data randomly ensures

the results robustness.

The following paragraphs expand on the different Machine Learning schemes. Figure 3 illustrates these schemes according to their types.

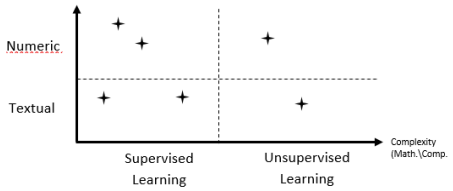

Figure 3: Clustering of Different Classifiers

Regressions

One approach and probably the most basic one is simply running a regression with all the features as the independent variables and the a numeric representation of the classes as the dependent variable. Some threshold value for the classes must be determined since the estimate for the explained variable, will have a non-deterministic value.

Linear Regression This method builds a linear dependency system between the explained variable (the *class* in my case) and the explaining variables (*features*). With the *Bag-of-Words* feature-approach, each variable x is a dummy variable indicating whether a given word m is present in Tweet i. The estimations parameters, which are denoted with β_j are derived using Ordinary Least Squares. In itself, the linear regression estimation is a weak predictor for the purpose of classification, but it allows for calculating other useful statistics such as the coefficient of determination, commonly known as R^2. This static demonstrates the part of the variance that is explained using the provided variables. As a rule of thumb, I want the largest possible chunk of variance to be explained when building a regression. Equation 1 shows the basic scheme for the use of linear regressions as classification tools, where n is the number of observations, m is the number of features.

$$y_i = \beta_1 x_{1i} + \beta_2 x_{2i} + ... + \beta_m x_{mi} + \epsilon_i \qquad \forall\ i \in [1, n]. \tag{1}$$

4.4 Classifier Types

placeholder

ython

5 Genaral Use Cases

placeholder

References

[1] Bird, S., Klein, E., and Loper, E. *Natural language processing with Python: analyzing text with the natural language toolkit*. O'Reilly Media, Inc., 2009.

[2] Demchenko, Y., Grosso, P., De Laat, C., and Membrey, P. Addressing big data issues in scientific data infrastructure. In *Collaboration Technologies and Systems (CTS), 2013 International Conference on* (2013), IEEE, pp. pp. 48–55.

[3] Klier, M., and Heinrich, B. Datenqualität als erfolgsfaktor im business analytics. *Controlling 28*, 8-9 (2016), pp. 488–494.

[4] Mitchell, T. *Machine Learning*. McGraw-Hill, 1997.

[5] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research 12* (2011), 2825–2830.

[6] Sørensen, T. A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on danish commons. *Biol. Skr. 5* (1948), pp. 1–34.

[7] Streaming APIs. Twitter Developer Documentation. `https://dev.twitter.com/streaming/overview`. Accessed: July 13. 2017.

[8] "tweet". *https://www.merriam-webster.com*. Merriam-Webster, Accessed: June 20. 2017.