Analytics for Noisy Unstructured Text Data I

Shourya Roy
IBM Research, India Research Lab, India

L. Venkata Subramaniam
IBM Research, India Research Lab, India

INTRODUCTION

Accdrnig to rscheearch at Cmabrigde Uinervtisy, it deosn't mttaer in what oredr the ltteers in a wrod are, the olny iprmoetnt tihng is that the frist and lsat ltteer be at the rghit pclae. Tihs is bcuseae the human mnid deos not raed ervey lteter by istlef, but the wrod as a wlohe.[1]

Unfortunately computing systems are not yet as smart as the human mind. Over the last couple of years a significant number of researchers have been focussing on noisy text analytics. Noisy text data is found in informal settings (online chat, SMS, e-mails, message boards, among others) and in text produced through automated speech recognition or optical character recognition systems. Noise can possibly degrade the performance of other information processing algorithms such as classification, clustering, summarization and information extraction. We will identify some of the key research areas for noisy text and give a brief overview of the state of the art. These areas will be, (i) classification of noisy text, (ii) correcting noisy text, (iii) information extraction from noisy text. We will cover the first one in this chapter and the later two in the next chapter.

We define *noise* in text as any kind of difference in the surface form of an electronic text from the intended, correct or original text. We see such *noisy text* everyday in various forms. Each of them has unique characteristics and hence requires special handling. We introduce some such forms of noisy textual data in this section.

Online Noisy Documents: E-mails, chat logs, scrapbook entries, newsgroup postings, threads in discussion fora, blogs, etc., fall under this category. People are typically less careful about the sanity of written content in such informal modes of communication. These are characterized by frequent misspellings, commonly and not so commonly used abbreviations, incomplete sentences, missing punctuations and so on. Almost always noisy documents are human interpretable, if not by everyone, at least by intended readers.

SMS: Short Message Services are becoming more and more common. Language usage over SMS text significantly differs from the standard form of the language. An urge towards shorter message length facilitating faster typing and the need for semantic clarity, shape the structure of this non-standard form known as the *texting language* (Choudhury et. al., 2007).

Text Generated by ASR Devices: ASR is the process of converting a speech signal to a sequence of words. An ASR system takes speech signal such as monologs, discussions between people, telephonic conversations, etc. as input and produces a string a words, typically not demarcated by punctuations as *transcripts*. An ASR system consists of an acoustic model, a language model and a decoding algorithm. The acoustic model is trained on speech data and their corresponding manual transcripts. The language model is trained on a large monolingual corpus. ASR convert audio into text by searching the acoustic model and language model space using the decoding algorithm. Most conversations at contact centers today between agents and customers are recorded. To do any processing of this data to obtain customer intelligence it is necessary to convert the audio into text.

Text Generated by OCR Devices: Optical character recognition, or 'OCR', is a technology that allows digital images of typed or handwritten text to be transferred into an editable text document. It takes the picture of text and translates the text into Unicode or ASCII. . For handwritten optical character recognition, the rate of recognition is 80% to 90% with clean handwriting.

Call Logs in Contact Centers: Today's contact centers (also known as call centers, BPOs, KPOs) produce huge amounts of unstructured data in the form of call logs apart from emails, call transcriptions, SMS, chat

transcripts etc. Agents are expected to summarize an interaction as soon as they are done with it and before picking up the next one. As the agents work under immense time pressure hence the summary logs are very poorly written and sometimes even difficult for human interpretation. Analysis of such call logs are important to identify problem areas, agent performance, evolving problems etc.

In this chapter we will be focussing on automatic classification of noisy text. Automatic text classification refers to segregating documents into different topics depending on content. For example, categorizing customer emails according to topics such as billing problem, address change, product enquiry etc. It has important applications in the field of email categorization, building and maintaining web directories e.g. DMoz, spam filter, automatic call and email routing in contact center, pornographic material filter and so on.

NOISY TEXT CATEGORIZATION

The text classification task is one of the learning models for a given set of classes and applying these models to new unseen documents for class assignment. This is an important component in many knowledge extraction tasks; real time sorting of email or files into folder hierarchies, topic identification to support topic-specific processing operations, structured search and/or browsing, or finding documents corresponding to long-term standing interests or more dynamic task-based interests. Two types of classifiers are generally commonly found viz. *statistical classifiers* and *rule based classifiers*.

In statistical techniques a *model* is typically trained on a corpus of labelled data and once trained the system can be used for automatic assignment of unseen data. A survey of text classification can be found in the work by Aas & Eikvil (Aas & Eikvil, 1999). Given a training document collection $\mathcal{D}=\{d_1, d_2, ..., d_M\}$ with true classes $\{y_1, y_2, ..., y_M\}$ the task is to learn a *model*. This model is used for categorizing a new unlabelled document d_u. Typically words appearing in the text are used as features. Other applications including search rely heavily on taking the markup or link structure of documents into account but classifiers only depend on the content of the documents or the collection of words present in the documents. Once features are extracted from documents, each document is converted into a document vector. Documents are represented in a vector space; each dimension of this space represents a single feature and the importance of that feature in that document gives the exact distance from the origin. The simplest representation of document vectors uses the binary event model, where if a feature $j \in V$ appears in document d_i, then the j^{th} component of d_i is 1 otherwise it is 0. One of the most popular statistical classification techniques is naive Bayes (McCallum, 1998). In the naive Bayes technique the probability of a document d_i belonging to class c is computed as:

$$\Pr(c \mid d) = \frac{\Pr(c,d)}{\Pr(d)}$$

$$= \frac{\Pr(c)\Pr(d \mid c)}{\Pr(d)}$$

$$\propto \Pr(c)\Pr(d \mid c)$$

$$\propto \prod_j P(d_j \mid c)$$

The final approximation of the above equation refers to the naive part of such a model, i.e., the assumption of word independence which means the features are assumed to be conditionally independent, given the class variable.

Rule-based learning systems have been adopted in the document classification problem since it has considerable appeal. They perform well at finding simple axis-parallel frontiers. A typical rule-based classification scheme for a category, say *C*, has the form:

Assign category C if antecedent or
Do no assign category C if antecedent or

The antecedent in the premise of a rule usually involves some kind of feature value comparison. A rule is said to cover a document or a document is said to satisfy a rule if all the feature value comparisons in the antecedent of the rule are true for the document. One of the well known works in the rule based text classification domain is RIPPER. Like a standard separate-and-conquer algorithm, it builds a rule set incrementally. When a rule is found, all documents covered by the rule are discarded including positive