Consistency scores in text data *

Rohan Alexander *University of Toronto*
Ke-Li Chiu *University of Toronto*

...

Keywords: ...

Introduction

When we think about a data science project, we may like to think that our job is to 'let the data speak'. But this is rarely the case in practise. Datasets can have errors, be biased, incomplete, or messy. In any case, it is the underlying statistical process of which the dataset is an artifact that is typically of interest. In order to use statistical models to understand that process we often need to prepare the dataset in some way. This is particularly the case when working with text. However, this process requires many decisions. Should we correct obvious errors? What about slightly-less-obvious errors? To what extent have we introduced new errors? Have we made decisions that have affected, or even driven, our results?

In this paper we introduce the concept of consistency for a text corpus. Consistency refers to the proportion of words that are able to be predicted by a trained model based on the previous words and surrounding context. Further, we define internal consistency as when the model is trained on the corpus itself, and external consistency as when the model is trained on a more general corpus. Together, these concepts provide a guide to the cleanliness and consistency of a text dataset. This can be important when deciding whether a dataset is fit for purpose; when carrying out data cleaning and preparation tasks; and as a comparison between datasets.

To provide an example, consider the sentence, 'the cat in the...'. A child who has read this book could tell you that the next word should be 'hat'. Hence if the sentence was actually 'the cat in the bat', then that child would know something was likely wrong. A consistency score would likely be lower than if the sentence were 'the cat in the hat'. After the researcher corrects this error, a consistency score would likely increase. By following how the consistency scores change during the data preparation and cleaning stages the researcher can better understand the effect of the changes. Including consistency scores when corpuses are shared allows researchers to be more transparent about their corpus. And finally, the use of consistency scores allows for automation in the cleaning process.

We apply our approach to X (one option is a Hansard, but it's really big, so yeah, I don't really want to do that). Additionally, we construct a Shiny app that computes internal and external consistency scores for smaller corpuses and allows the researcher to make changes and see how it updates.

The remainder of our paper is structured as follows...

*Our code and datasets are available at: X. Comments on the 08 July 2020 version of this paper are welcome at: rohan.alexander@utoronto.ca.

Background

A typical data science workflow involves

Internal and external consistency

/// Start of OpenAI GPT-3 section ///

In recent years, pre-trained language models have tremendous contributions in advancing natural language processing (NLP) tasks such as reading comprehension, text generation, and many others (citation). As the pre-trained models remove the need for researchers and technologists to train models from scratch, they promote faster growth and advancement int the field of NLP. OpenAI GPT-3 is the latest generative pre-trained language model released by OpenAI in 2020. With 175 billion parameters, GPT-3 is claimed to have the capacity to generate text that ressembles human creation (citation). GPT-2, the predecessor of GPT-3, employs a semi-supervised approach that combines unsupervised pre-training and supervised fine-tuning where task-specific data set and task-specific fine-tuning were still needed in achieving these tasks until the birth of GPT-3 (citation). GPT-3 removes such needs, and its task-agnostic nature enables the accessible creation of cutting-edge NLP applications.

GPT-3 also learns like humans do — with brief directives and simple instructions…

The task-agnostic feature also permits it to adapt to the style of the data being fed to it (citation).

In our application, we deploy OpenAI GPT-3 to generate text and …

/// End of OpenAI GPT-3 section ///

Article: Improving Language Understanding by Generative Pre-Training (2018)

Introduction of a semi-supervised approach for language understanding machine learning tasks. The approach is the combinaiton of unsupervised pre-training and supervised fine-tuning.

Article: Language Models are Few-Shot Learners (2020)

Official paper from OpenAI to introduce OpenAI GPT-3. Emphasizes on the "fewer-shots" and "task-agnostic" aspects of the latest langual model compared to its predecessor GPT-2.

Stylized example

As a stylized example, let's consider the following actual paragraph from Jane Eyre, by Charlotte Bronte:

> There was no possibility of taking a walk that day. We had been wandering, indeed, in the leafless shrubbery an hour in the morning; but since dinner (Mrs. Reed, when there was no company, dined early) the cold winter wind had brought with it clouds so sombre, and a rain so penetrating, that further out-door exercise was now out of the question.

Let's pretend that this text had been created from optical character recognition and that it had the following errors: some 'h' were replaced with 'b'; and some 'd' have been replaced with 'al':

> There was no possibility of taking a walk that day. We had been wandering, indeed, in tbe leafless shrubbery an hour in tbe morning; but since dinner (Mrs. Reeal, when tbere was no company, dined early) the cold winter wind had brought with it clouds so sombre, and a rain so penetrating, tbat further out-door exercise was now out of the question.

Assume a model that is trained to perfectly forecast the next word in Jane Eyre. For this fragment there are 62 words, comprising 5 errors and 57 correct words. So the internal consistency score of this fragment would be: 57/62 = 0.919. When we recognise and correct the errors, this consistency score would increase to 1.

Similarly, assume a model that is trained on an external data source. This means that it will recognise the the cases where some 'h' were replaced with 'b', but not recognise that 'Reed' has become 'Reeal'. Hence, the external consistency score would be 58/62 = 0.935.

Models

Various models can be used....

ngrams

Traditionally, word-correction techniques evaluate errors one by one without considering the context of the surrounding words. This was no longer the case in modern correction techniques as statistical language models (SLMs) and feature-based methods have been used for context-sensitive correction (citation). Without exception, all human languages have some words that co-occur more frequently with others. Under this assumption, we can regard the production of English text as a set of conditional probabilities, written as $\Pr(w_k \mid w_1^{k-1})$, where k is the number of words in a sentence, w_k is the predicted word, and w_1^{k-1} is the history of the word occurring in a sequence (citation). In other word, the generation of prediction w_k is based on the history w_1^{k-1}. This conditional probability is the foundation of an n-gram language model.

An n-gram model is a probabilistic language model that predicts the next word in a sequence of words. The n in n-gram represents the number of words in a sequence. Taking an incomplete sentence from the Jane Eyre excerpt as an example:

> "We had been wandering,"

"We" is a unigram, "We had" is a bigram, "We had been" is a trigram, and "We had been wandering" is a 4-gram. On the other hand, the trigrams of the excerpt are "We had been," and "had been wandering." The overlapping chaining of words reveals the statistical behaviour in human language. Moreover, N-gram then enables us to assign a

probability to the occurrence of a sequence of words or the likelihood of a word occurring next. Consider the two sentences: "We had been wandering," and "We had been wangling." The former is likely to be more frequently encountered in a training corpus. Thus, the n-gram would assign a higher probability to "wandering" than to "wangling."

By including the n into the previous conditional probability to be $\Pr(w_k \mid w_1^{k-1})$

The application of n-gram is versatile. N-gram is widely applied in text prediction, spelling-correction, machine translation...

Article: Class-Based n-gram Models of Natural Language (1992)

Syntax-based (grammartical) and semantic-based (sensible) word classifications in n-gram models.

Article: A Statistical Approach to Automatic OCR Error Correction in Context (1996)

Context-sensitive correction system to both non-word and real-word errors based on n-gram models applied to OCR postprocessing.

Article: Unsupervised Learning of Sentence Embeddings using Compositional n-Gram Features (2017)

Embedding of sentences (sequence of words with semantic representations) compositional n-gram features.

Something

Soemthing else

Data

Various data can be used....

Application

Discussion

Internal validity vs external- one is their own words the other is a general set of words.

In the same way that precision and recall provide important measures...