TextCompare

Prof. Tony Eng, Antonio Rivera, Vahid Fazel-Rezai

1 Introduction

Learning a new language can be daunting because it can be hard to know where to start. There are lots of words in a given language, and one has to somehow organize them into some order for learning, basic words first and slowly become more advanced. One way to acquire new words is to read books or watch movies in the language. We propose a system for deciding, for example, which books to read and in which order. Given a set of text blobs (e.g. book or movie transcripts), we construct a graph with edges indicating jump in skill level or perhaps other metrics. Using the graph, we can find a path from the current skill level (represented as a blob of words or as starting nodes in the graph) to the desired set of words in which each step has a limit on the number of new words. An additional benefit of the system is that it can potentially even discern which words are more basic or more advanced. The key problems to solve in order to create such a system is to (1) find a good metric for the edge weights and (2) easily make apply the graph to its use case.

2 Data

We have data of several types:

- Control documents with specific properites (e.g. single repeated word).
- Children's books.
- Nursery rhymes.

3 Similarity and Distance Metrics

Types of words

$$\begin{aligned} a &= \text{Number of unique words only in A} \\ b &= \text{Number of unique words only in B} \\ c &= \text{Number of unique words in common} \end{aligned}$$

The metrics used to compare two texts can be split into three groups:

1. Metrics that use only c (i.e. overlap comparison). These include Jaccard index, Canberra, Sorenson, and Minkowski2.
2. Metrics that use a, b, and c. Examples of these are TF, TF-IDF, and their variants, such as sublinear TF-IDF and TF-ITTF.
3. Asymmetric metrics. The main one used is Tversky Index, but one could also consider simpler metrics such as number of new words or number of new occurrences.

TF-IDF and Variants:

These metrics measure the angle between word vector representations of texts. The difference between the variants is how those word vectors are determined. The simplest form is TF (Term Frequency). This simply constructs word vectors where the ith value in the vector denotes the frequency of term i for that particular text. Given two texts A and B with word vectors X and Y respectively, we then simply calculate the angle between the texts:

$$\arccos\left(\frac{X \cdot Y}{\|X\|\|Y\|}\right)$$

Asymmetric Metrics:
Tversky index is defined, for sets of words X and Y, as:

$$\frac{|X \cap Y|}{|X \cap Y| + \alpha|X - Y| + \beta|Y - X|}$$

with $\alpha, \beta \geq 0$. It should be noted that if $\alpha = \beta$, then the metric is symmetric. For our purposes, I think that setting $\alpha = 0$ will give us the metric we want. When considering whether a child can read book Y after reading book X, what we care about is how many new words there are in Y. Thus, letting $\alpha = 0$ and $\beta \neq 0$ reflects this. This intuition has been reflected when attempting to maximize/minimize particular distances in a test set of books.

Distance:

- Vocabulary (number of new words)
-

Similarity:

- Vocabulary (number of overlapping words)

4 Path Finding

Clustering
Finding scalar score for each node that minimizes error