Programming and Classification Summer 2019

Jaccard similarity — 27.03, 2019

dr hab. inż. Marek Klonowski *Scribe: Krzysztof Agieńczuk*

1 Overview

In the last lecture we talked about **metrics** and **distances between words** (Hamming, Levenshtein).

In this lecture we covered: **Jaccard similarity** and

2 Sets and bags

Set is a collection of elements in which order does not matter and elements can appear a few times (and will be simplified). Eg. { a,a,b,c,c,c } = { a,b,c }

Bag is a set with repetitions. Multiple elements will not be simplified. Eg. { a,a,b,c,c,c } $\neq$ { a,b,c }. Same operations are defined as in sets, ie. unions, intersections, differences, etc.

3 Jaccard similarity

Definition 3.1. *Jaccard similarity*
$For A \cup B \neq \emptyset$

$$J_{SIM}(A,B) = \frac{|A \cap B|}{|A \cup B|} \tag{1}$$

Jaccard distance is defined as

$$1 - J_{SIM} \tag{2}$$

This distance is a metric. Proof can be achieved by definition. First two properties are trivial. The third is not easy and proving it was skipped during the lecture.
Jaccard similarity is defined for sets. It can be used for bags, but then the J_{SIM} is in range $[0, 0.5]$. Jaccard distance in this case is not a metric, though still may be useful.

3.1 Applications of Jaccard distance

Possible applications:

- Similarity of documents
- Plagiarism

- Filtering news
- Mirror pages
- Collaborative filtering

Collaborative filtering is "recommendation". Knowing what users like we can suggest them new things basing on Jaccard similarity between (for example) movies they watched and movies their friends watched. However users may not have liked some movies. Taking their ratings into consideration we get better results. Some ideas for using those ratings is eg.

- removing films that are not liked
- using two lists (liked and hated)
- using bags instead of sets

In the last approach we have one movie many times in a bag. The number of occurrences is number of stars given to the title by user (so if movie was rated 5/5 it appears in bag 5 times).

4 Shingle

Shingle usually refers to substring of length k. Set of k-shingles is characteristic for a given document. The problem is what to do with white characters or blank spaces.
K should be picked large enough that the probability of any given shingle in a random text of the typical length (in corpus) is low. Typically $k = 5, ..., 10$. Shingles can be built also using words, not only letters.

Number of shingles in the worst case scenario is

$$numberOfWordsInAText - k + 1 \tag{3}$$

This means that text characteristics (when considering big data examples) can be bigger than the text itself