The q-gram sequence comparison model

- Like the maximal matches model, the q-gram model considers common substrings of the strings to be compared.
- Here $q > 0$ and a *q-gram* of a sequence s is a substring of s of length q.
- A q-gram is sometimes called q-mer or q-tuple.
- Technically, in this model, one counts the number of occurrences of different q-grams in the two sequences to be compared.
- Thus, sequences with many common q-grams have a small distance, independent of where they occur.
- The q-gram model was first described in [Ukkonen, 1992].

- While the maximal matches model considers substrings of possibly different length, the q-gram model restricts to substrings of a fixed length q.
- In this section, let q be a positive integer.
- Recall that u and v are sequences of length m and n, respectively.

Definition 1

The *q-gram profile* of u is the function $P_{u,q} : \mathcal{A}^q \to \mathbb{N}$, such that for any $w \in \mathcal{A}^q$, $P_{u,q}(w)$ is the number of different positions in u where w occurs as substring. $\square$

Example 1

Let $\mathcal{A} = \{\mathrm{a}, \mathrm{c}\}$ and $q = 2$. The q-gram profile of $u = \mathrm{aaca}$ is

$$\mathrm{aa} \mapsto 1, \mathrm{ac} \mapsto 1, \mathrm{ca} \mapsto 1, \mathrm{cc} \mapsto 0$$

The q-gram profile of $v = \mathrm{acacaacc}$ is

$$\mathrm{aa} \mapsto 1, \mathrm{ac} \mapsto 3, \mathrm{ca} \mapsto 2, \mathrm{cc} \mapsto 1$$

The q-gram sequence comparison model

- The size of the alphabet $\mathcal{A}$ and the choice of q determine the q-gram profile.
- For example, if $q = 3$ and $|\mathcal{A}| = 4$, then $|\mathcal{A}|^q = 64$.
- That is, we can assume that in a short string, all q-grams occur, i.e. all values in the profile are > 0.
- If $q = 4$ and $|\mathcal{A}| = 20$, then $|\mathcal{A}|^q = 160\,000$ and the string has to be very long to contain all q-grams.
- In general, one chooses $q \ll n$, e.g. $8 \leq q \leq 11$ when comparing, e.g. entire bacterial genomes.

– As for given $\mathcal{A}$ and q the q-grams are uniquely determined, one often writes the q-gram profile as an ordered list

$$[P_{u,q}(w_0), P_{u,q}(w_1), \dots, P_{u,q}(w_{r^q-1})]$$

with the q-grams w_0, w_1, $\dots w_{r^q-1}$ in lexicographic order, where $r = |\mathcal{A}|$.

Example 2

Let $\mathcal{A} = \{\mathsf{a}, \mathsf{c}\}$ and $q = 2$. The two q-gram profile $\mathsf{aa} \mapsto 1, \mathsf{ac} \mapsto 1, \mathsf{ca} \mapsto 1, \mathsf{cc} \mapsto 0$ is written as $[1, 1, 1, 0]$. The q-gram profile $\mathsf{aa} \mapsto 1, \mathsf{ac} \mapsto 3, \mathsf{ca} \mapsto 2, \mathsf{cc} \mapsto 1$ is written as $[1, 3, 2, 1]$.

The q-gram distance is just the sum of the absolute differences of the profile-lists.

Definition 2

The *q-gram distance* $qgdist(u, v)$ of u and v is defined by

$$qgdist(u, v) = \sum_{w \in \mathcal{A}^q} |P_{u,q}(w) - P_{v,q}(w)|. \quad \square$$

- One can show that the symmetry and the triangle inequality hold for *qgdist* (cf. [Ukkonen, 1992]).
- The zero property does not hold as shown by the following example.

Example 3

Let $\mathcal{A} = \{\mathrm{a}, \mathrm{c}\}$ and $q = 2$. Then $u = \mathrm{aaca}$ and $v = \mathrm{acaa}$ have the same q-gram profile

$$\mathrm{aa} \mapsto 1, \mathrm{ac} \mapsto 1, \mathrm{ca} \mapsto 1, \mathrm{cc} \mapsto 0$$

Hence, the q-gram distance of u and v is 0. As $u \neq v$, this contradicts the zero-property ($f(x, y) = 0 \iff x = y$). So *qgdist* is not a metric.

The q-gram sequence comparison model

The simplest method to compute the q-gram distance is to encode each q-gram into a number, and to use these numbers as indices into tables holding the counts for the corresponding q-gram.

Definition 3

- Let $\mathcal{A} = \{a_1, \ldots, a_r\}$ be an ordered alphabet such that $a_1 < a_2 < \cdots < a_r$.
- Then

$$\overline{a_\ell} = \ell - 1$$

is the code of a_ℓ and

$$\overline{w} = \sum_{i=1}^{q} \underbrace{\overline{w[i]}}_{\substack{\text{char}\\\text{code}}} \cdot \underbrace{r^{q-i}}_{\text{weight}}$$

is the code of $w \in \mathcal{A}^q$.

Example 4

- Let $\mathcal{A} = \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$ be the DNA-alphabet and define $\overline{\mathtt{A}} = 0$, $\overline{\mathtt{C}} = 1$, $\overline{\mathtt{G}} = 2$, and $\overline{\mathtt{T}} = 3$.
- For $q = 3$, there are $r^q = 4^3 = 64$ q-grams.
- Here are some examples of how the codes are computed:

$$\overline{\mathtt{AAA}} = \overline{\mathtt{A}} \cdot 4^2 + \overline{\mathtt{A}} \cdot 4^1 + \overline{\mathtt{A}} \cdot 4^0 = 0 \cdot 16 + 0 \cdot 4 + 0 \cdot 1 = 0$$

$$\overline{\mathtt{ATA}} = \overline{\mathtt{A}} \cdot 4^2 + \overline{\mathtt{T}} \cdot 4^1 + \overline{\mathtt{A}} \cdot 4^0 = 0 \cdot 16 + 3 \cdot 4 + 0 \cdot 1 = 12$$

$$\overline{\mathtt{CGT}} = \overline{\mathtt{C}} \cdot 4^2 + \overline{\mathtt{G}} \cdot 4^1 + \overline{\mathtt{T}} \cdot 4^0 = 1 \cdot 16 + 2 \cdot 4 + 3 \cdot 1 = 16 + 8 + 3 = 27$$

$$\overline{\mathtt{TAA}} = \overline{\mathtt{T}} \cdot 4^2 + \overline{\mathtt{A}} \cdot 4^1 + \overline{\mathtt{A}} \cdot 4^0 = 3 \cdot 16 + 0 \cdot 4 + 0 \cdot 1 = 48$$

$$\overline{\mathtt{TTT}} = \overline{\mathtt{T}} \cdot 4^2 + \overline{\mathtt{T}} \cdot 4^1 + \overline{\mathtt{T}} \cdot 4^0 = 3 \cdot 16 + 3 \cdot 4 + 3 \cdot 1 = 48 + 12 + 3 = 63$$

The q-gram sequence comparison model

Let us generalize on the previous example. The first character in w is weighted by r^{q-1}, the second character by r^{q-2}, etc. If all characters in $w \in \mathcal{A}^q$ have a minimum code 0, then

$$\begin{aligned}\overline{w} &= \sum_{i=1}^{q} \overline{w[i]} \cdot r^{q-i} \\ &= \sum_{i=1}^{q} 0 \cdot r^{q-i} \\ &= \sum_{i=1}^{q} 0 \\ &= 0\end{aligned}$$

That is, the minimum code of any $w \in \mathcal{A}^q$ is 0.

The q-gram sequence comparison model

The maximum code is obtained when all characters in w have a maximum code $r-1$. Then

$$
\begin{aligned}
\overline{w} &= \sum_{i=1}^{q} \overline{w[i]} \cdot r^{q-i} \\
&= \sum_{i=1}^{q} (r-1) \cdot r^{q-i} \\
&= (r-1) \cdot \sum_{i=1}^{q} r^{q-i} \\
&= (r-1) \cdot (r^0 + r^1 + \cdots + r^{q-2} + r^{q-1}) \\
&= r \cdot (r^0 + r^1 + \cdots + r^{q-2} + r^{q-1}) - (r^0 + r^1 + \cdots + r^{q-2} + r^{q-1}) \\
&= r^1 + r^2 + \cdots + r^{q-1} + r^q - (r^0 + r^1 + \cdots + r^{q-2} + r^{q-1}) \\
&= r^1 + r^2 + \cdots + r^{q-1} - (r^1 + \cdots + r^{q-1}) + r^q - r^0 = r^q - 1
\end{aligned}
$$

That is, the maximum code of any $w \in \mathcal{A}^m$ is $r^q - 1$.

- Moreover, for any $u, w \in \mathcal{A}^q$, $\overline{u} = \overline{w}$ implies $u = v$ (proof will be an exercise).
- As $|\mathcal{A}^q| = r^q$ and there are r^q numbers in the range from 0 to $r^q - 1$, we conclude: for each i, $0 \leq i \leq r^q - 1$, there is some $w \in \mathcal{A}^q$ such that $\overline{w} = i$.
- To put it into mathematical terms, the mapping from q-grams to integer codes is bijective.

Example 5

- Let $\mathcal{A} = \{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$ be the DNA-alphabet and define $\overline{\mathtt{A}} = 0$, $\overline{\mathtt{C}} = 1$, $\overline{\mathtt{G}} = 2$, and $\overline{\mathtt{T}} = 3$.
- For $q = 3$, there are $r^q = 4^3 = 64$ q-grams.
- The smallest q-gram AAA in the lexicographic order of all q-grams has integer code 0, the second smallest AAC has integer code 1, etc.
- In general, for any q and any ordered alphabet, the ith q-gram in the lexicographic order of all q-grams gets code i, see Figure 1 for an example.

Figure 1: All 3-grams over the alphabet $\{\mathtt{A}, \mathtt{C}, \mathtt{G}, \mathtt{T}\}$ with $\overline{\mathtt{A}} = 0$, $\overline{\mathtt{C}} = 1$, $\overline{\mathtt{G}} = 2$, and $\overline{\mathtt{T}} = 3$.

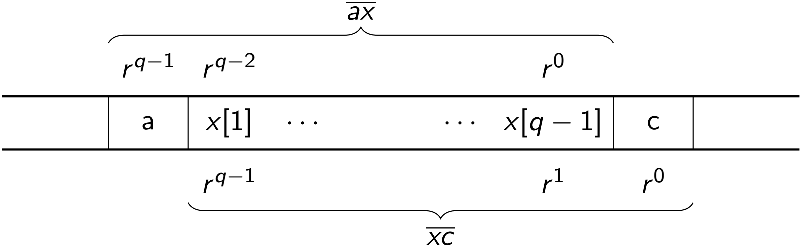
AAA	0	ACA	4	AGA	8	ATA	12
AAC	1	ACC	5	AGC	9	ATC	13
AAG	2	ACG	6	AGG	10	ATG	14
AAT	3	ACT	7	AGT	11	ATT	15
CAA	16	CCA	20	CGA	24	CTA	28
CAC	17	CCC	21	CGC	25	CTC	29
CAG	18	CCG	22	CGG	26	CTG	30
CAT	19	CCT	23	CGT	27	CTT	31
GAA	32	GCA	36	GGA	40	GTA	44
GAC	33	GCC	37	GGC	41	GTC	45
GAG	34	GCG	38	GGG	42	GTG	46
GAT	35	GCT	39	GGT	43	GTT	47
TAA	48	TCA	52	TGA	56	TTA	60
TAC	49	TCC	53	TGC	57	TTC	61
TAG	50	TCG	54	TGG	58	TTG	62
TAT	51	TCT	55	TGT	59	TTT	63

The q-gram sequence comparison model

An important property is that the code of each q-gram in a sequence can be computed incrementally in constant time, due to the fact that

$$\overline{xc} = (\overline{ax} - \overline{a} \cdot r^{q-1}) \cdot r + \overline{c}$$

for any $x \in \mathcal{A}^{q-1}$ and any $a, c \in \mathcal{A}$, see the following illustration:



The q-gram sequence comparison model

The algorithm to compute the q-gram distance (see Algorithm 1) follows the following strategy:

1. Accumulate the q-gram profiles of u and v in two arrays τ_u and τ_v such that

   $$\tau_u[\overline{w}] = P_{u,q}(w) \text{ and } \tau_v[\overline{w}] = P_{v,q}(w)$$

   for all $w \in \mathcal{A}^q$.
2. Compute the set $C = \{\overline{w} \mid w \text{ is } q\text{-gram of } u \text{ or } v\}$, i.e. the set of codes of all q-grams in u and w.
3. Compute $qgdist(u, v) = \sum_{c \in C} |\tau_u[c] - \tau_v[c]|$.

Algorithm 1 (Computation of q-gram distance)

Input: sequences $u = u[1 \ldots m]$, $v = v[1 \ldots n]$ over alphabet $\mathcal{A}$, $q > 0$
Output: $qgdist(u, v)$

$r \leftarrow |\mathcal{A}|$
for $c \leftarrow 0$ **upto** $r^q - 1$ **do**
    $(\tau_u[c], \tau_v[c]) \leftarrow (0, 0)$
end for
$c \leftarrow \sum_{i=1}^{q} \overline{u[i]} \cdot r^{q-i}$
$\tau_u[c] \leftarrow 1$
$C \leftarrow \{c\}$
for $i \leftarrow 1$ **upto** $m - q$ **do**
    $c \leftarrow (c - \overline{u[i]} \cdot r^{q-1}) \cdot r + \overline{u[i+q]}$
    if $\tau_u[c] = 0$ **then**
        $C \leftarrow C \cup \{c\}$
    end if
    $\tau_u[c] \leftarrow \tau_u[c] + 1$
end for

Algorithm 1 continued

$c \leftarrow \sum_{i=1}^{q} \overline{v[i]} \cdot r^{q-i}$

$\tau_v[c] \leftarrow 1$

if $\tau_u[c] = 0$ **then**

  $C \leftarrow C \cup \{c\}$

end if

for $i \leftarrow 1$ **upto** $n - q$ **do**

  $c \leftarrow (c - \overline{v[i]} \cdot r^{q-1}) \cdot r + \overline{v[i+q]}$

  if $\tau_u[c] = 0$ and $\tau_v[c] = 0$ **then**

    $C \leftarrow C \cup \{c\}$

  end if

  $\tau_v[c] \leftarrow \tau_v[c] + 1$

end for

return $\sum_{c \in C} |\tau_u[c] - \tau_v[c]|$

The q-gram sequence comparison model

- Let us consider the efficiency of the algorithm.
- The space for the arrays τ_u and τ_v is $O(r^q)$.
- The space for the set C is $O(m - q + 1 + n - q + 1) = O(m + n)$.
- Hence the total space requirement is $O(m + n + r^q)$.
- We need $O(r^q)$ time to initialize the arrays τ_u and τ_v.
- The computation of the codes requires $O(m + n)$ time.
- Each array lookup and update requires $O(1)$ time.
- Hence the total running time is $O(m + n + r^q)$.
- If $r^q \in O(n + m)$, then this method is optimal.

Like the maximal matches distance, the q-gram distance provides a lower bound for the unit edit distance.

Theorem 4

Let δ be the unit cost function. Then $qgdist(u, v)/(2 \cdot q) \leq edist_\delta(u, v)$.

Proof.

See [Jokinen and Ukkonen, 1991] or [Ukkonen, 1992]. □

- The relation between *qgdist* and $edist_\delta$ suggests to use *qgdist* as a filter in contexts where the unit edit distance is of interest only below some threshold k.
- See the remarks at the end of the section on the maximal matches model.

Remark: [Luczak et al., 2019, Cattaneo et al., 2022] define several variations of distance measure based on the q-gram profile.

Cattaneo, G., Ferraro Petrillo, U., Giancarlo, R., Palini, F., and Romualdi, C. (2022).
The power of word-frequency-based alignment-free functions: a comprehensive large-scale experimental analysis.
Bioinformatics, 38(4):925–932.

Jokinen, P. and Ukkonen, E. (1991).
Two Algorithms for Approximate String Matching in Static Texts.
In *Proceedings of the 16th International Symposium on Mathematical Foundations of Computer Science*, pages 240–248. Lecture Notes in Computer Science **520**, Springer Verlag.

Luczak, B. B., James, B. T., and Girgis, H. Z. (2019).
A survey and evaluations of histogram-based statistics in alignment-free sequence comparison.
Briefings in bioinformatics, 20(4):1222–1237.

Ukkonen, E. (1992).
Approximate String-Matching with q-Grams and Maximal Matches.
TCS, 92(1):191–211.