Polynomial-Time Approximation Algorithms for Weighted LCS Problem

Marek Cygan[1], Marcin Kubica[1], Jakub Radoszewski[1], Wojciech Rytter[1,2] and **Tomasz Waleń**[1]

[1]University of Warsaw, Poland

[2]Copernicus University, Toruń, Poland

CPM 2011, 2011–06–29

Definitions

Definition of a weighted sequence

A weighted sequence $X = x_1x_2 \ldots x_n$ of length $|X| = n$ over an alphabet $\Sigma = \{\sigma_1, \sigma_2, \ldots, \sigma_K\}$ is a sequence of sets of pairs of the form:

$$x_i = \{(\sigma_j,\ p_i^{(X)}(\sigma_j))\ :\ j = 1, 2, \ldots, K\}.$$

Here $p_i(\sigma_j)$ is the occurrence probability of the character σ_j at the position i, these values are non-negative and sum up to 1 for a given i.
$\mathcal{WS}(\Sigma)$ is the set of all weighted sequences over the alphabet Σ.
We assume that $|\Sigma| = O(1)$.

Definitions

Example

x_1	x_2	x_3	x_4
$p_1(\mathtt{a}) = 1/3$	$p_2(\mathtt{a}) = 1$	$p_3(\mathtt{a}) = 0$	$p_4(\mathtt{a}) = 1/2$
$p_1(\mathtt{b}) = 1/3$	$p_2(\mathtt{b}) = 0$	$p_3(\mathtt{b}) = 1/2$	$p_4(\mathtt{b}) = 1/4$
$p_1(\mathtt{c}) = 1/3$	$p_2(\mathtt{c}) = 0$	$p_3(\mathtt{c}) = 1/2$	$p_4(\mathtt{c}) = 1/4$

A weighted sequence $X = x_1x_2x_3x_4$ over the alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$

Background

- Weighted sequences are also referred to in the literature as p-weighted sequences or Position Weighted Matrices (PWM) [Amir et al. 2010, Thompson et al. 1994].
- The notion of a weighted sequence was introduced as a tool for motif discovery and local alignment, and is extensively used in computational molecular biology.
- Multiple algorithmic results related to combinatorics of weighted sequences, i.e., repetitions, regularities and pattern matching, have already been presented.

Background

- Weighted sequences are also referred to in the literature as p-weighted sequences or Position Weighted Matrices (PWM) [Amir et al. 2010, Thompson et al. 1994].
- The notion of a weighted sequence was introduced as a tool for motif discovery and local alignment, and is extensively used in computational molecular biology.
- Multiple algorithmic results related to combinatorics of weighted sequences, i.e., repetitions, regularities and pattern matching, have already been presented.

- Weighted sequences are also referred to in the literature as p-weighted sequences or Position Weighted Matrices (PWM) [Amir et al. 2010, Thompson et al. 1994].
- The notion of a weighted sequence was introduced as a tool for motif discovery and local alignment, and is extensively used in computational molecular biology.
- Multiple algorithmic results related to combinatorics of weighted sequences, i.e., repetitions, regularities and pattern matching, have already been presented.

Definitions

Definition (Occurence of subsequence s in weighted sequence X)

$|s| = d$, $\pi = (i_1, i_2, \ldots, i_d)$, $1 \leq i_1 < i_2 < \ldots < i_d \leq |X|$,

$$\mathcal{P}_X(\pi, s) \;=\; \prod_{k=1}^{d} \, p_{i_k}^{(X)}(s_k).$$

$$SUBS(X, \alpha) \;=\; \left\{ s \in \Sigma^* \;:\; \exists \, \left(\pi \in \mathrm{Seq}_{|s|}^{|X|} \right) \;\; \mathcal{P}_X(\pi, s) \geq \alpha \right\}.$$

In other words $SUBS(X, \alpha)$ is the set of deterministic strings which match a subsequence of X with probability at least α.

Problems

α-LCWS problem

Input: Two weighted sequences $X, Y \in \mathcal{WS}(\Sigma)$ and a cut-off probability α.
Output: The longest string $s \in \Sigma^*$ such that

$$\exists \left(\pi \in \mathrm{Seq}_{|s|}^{|X|},\ \pi' \in \mathrm{Seq}_{|s|}^{|Y|}\right) \quad \mathcal{P}_X(\pi, s) \cdot \mathcal{P}_Y(\pi', s) \geq \alpha.$$

Equivalently, s is the longest string in $SUBS(X, \alpha_1) \cap SUBS(Y, \alpha_2)$ for some $\alpha_1 \cdot \alpha_2 \geq \alpha$.

(α_1, α_2)-LCWS2 problem

Input: Two weighted sequences X, Y and two cut-off probabilities α_1, α_2.
Output: The longest string $s \in SUBS(X, \alpha_1) \cap SUBS(Y, \alpha_2)$.

Problems

α-LCWS problem

Input: Two weighted sequences $X, Y \in \mathcal{WS}(\Sigma)$ and a cut-off probability α.
Output: The longest string $s \in \Sigma^*$ such that

$$\exists \left(\pi \in \mathrm{Seq}_{|s|}^{|X|},\ \pi' \in \mathrm{Seq}_{|s|}^{|Y|}\right) \quad \mathcal{P}_X(\pi, s) \cdot \mathcal{P}_Y(\pi', s) \geq \alpha.$$

Equivalently, s is the longest string in $SUBS(X, \alpha_1) \cap SUBS(Y, \alpha_2)$ for some $\alpha_1 \cdot \alpha_2 \geq \alpha$.

(α_1, α_2)-LCWS2 problem

Input: Two weighted sequences X, Y and two cut-off probabilities α_1, α_2.
Output: The longest string $s \in SUBS(X, \alpha_1) \cap SUBS(Y, \alpha_2)$.

Example: α-LCWS problem

X	1	2	3	4	5
a	**0.9**	0.2	1.0	0.3	**0.9**
b	0.1	**0.8**	0.0	**0.7**	0.1

Y	1	2	3	4	5
a	**0.9**	0.5	0.1	0.2	**0.8**
b	0.1	0.5	**0.9**	**0.8**	0.2

(s, π, π') is the solution for α-LCWS problem for $\alpha = 0.23$.

$s = abba$
$\pi = (1, 2, 4, 5)$
$\pi' = (1, 3, 4, 5)$
$\mathcal{P}_X(\pi, s) = 0.9 \cdot 0.8 \cdot 0.7 \cdot 0.9 = 0.4536$
$\mathcal{P}_Y(\pi', s) = 0.9 \cdot 0.9 \cdot 0.8 \cdot 0.8 = 0.5184$
$\mathcal{P}_X(\pi, s) \cdot \mathcal{P}_Y(\pi', s) = 0.23514624$

Example: (α_1, α_2)-LCWS2 problem

X	1	2	3	4	5
a	**0.9**	0.2	**1.0**	0.3	0.9
b	0.1	**0.8**	0.0	0.7	0.1

Y	1	2	3	4	5
a	**0.9**	0.5	0.1	0.2	**0.8**
b	0.1	0.5	**0.9**	0.8	0.2

Solution for (α_1, α_2)-LCWS2 for $\alpha_1 = 0.7$, $\alpha_2 = 0.6$.

$s = aba$
$\pi = (1, 2, 3)$
$\pi' = (1, 3, 5)$

$\mathcal{P}_X(\pi, s) = 0.9 \cdot 0.8 \cdot 1.0 = 0.72$
$\mathcal{P}_Y(\pi', s) = 0.9 \cdot 0.9 \cdot 0.8 = 0.648$

Results summary

Previous results for α-LCWS [Amir et al. 2010]

The α-LCWS problem can be solved in $O(n^3)$ time and $O(n^2)$ space. If we are only interested in the length of the output, the problem can be solved in $O(Ln^2)$ time, where L is the length of the solution.

NP-hardness for integer version of (α_1, α_2)-LCWS2

Previous work	Our results
unbounded alphabet	$\lvert\Sigma\rvert = 2$

Approximation results for (α_1, α_2)-LCWS2

Previous work	Our results
$(1/\lvert\Sigma\rvert)$	0.5 ($O(n^5)$ time, $O(n^2)$ space)
	PTAS ($O(n^5)$ space)

Definition (α-LCWS2 problem)

Input: Two weighted sequences $X, Y \in \mathcal{WS}(\Sigma)$ and a cut-off probability α.
Output: The longest string $s \in SUBS(X, \alpha) \cap SUBS(Y, \alpha)$.

The following lemma shows that the (α_1, α_2)-LCWS2 and α-LCWS2 problems are equivalent.

Lemma

The (α_1, α_2)-LCWS2 problem can be reduced in linear time to the α-LCWS2 problem (with $\alpha = \min(\alpha_1, \alpha_2)$).

Proof.

Solution: just rescale probabilities, and add special symbol # that will sum new probabilities to 1.
Let $\alpha_1 < \alpha_2$, and $\gamma = \log_{\alpha_2} \alpha_1$.

$$p_i^{(X')}(\sigma_j) = p_i^{(X)}(\sigma_j), \quad p_i^{(X')}(\#) = 0$$

$$p_i^{(Y')}(\sigma_j) = p_i^{(Y)}(\sigma_j)^\gamma, \quad p_i^{(Y')}(\#) = 1 - \sum_{j=1}^{k} p_i^{(Y')}(\sigma_j).$$

□

NP-hardness

Definition

Define an *I-weighted sequence* X over the alphabet $\Sigma = \{\sigma_1, \sigma_2, \ldots, \sigma_K\}$ as a sequence of sets of pairs of the form:

$$x_i = \{(\sigma_j,\ w_i^{(X)}(\sigma_j)) \ : \ j = 1, 2, \ldots, K\}, \quad \text{where } w_i^{(X)}(\sigma_j) \in \mathbb{Z}_+.$$

Definition

For an I-weighted sequence X and $s \in \Sigma^d$, define:

$$\mathcal{W}_X(\pi, s) \ = \ \sum_{k=1}^{d} w_{i_k}^{(X)}(s_k) \qquad \text{for } \pi = (i_1, \ldots, i_d) \in \mathrm{Seq}_d^{|X|}.$$

For an I-weighted sequence X and $\alpha \in \mathbb{Z}_+$, denote:

$$SUBS(X, \alpha) \ = \ \left\{ s \in \Sigma^* \ : \ \exists \ \left(\pi \in \mathrm{Seq}_{|s|}^{|X|}\right) \ \mathcal{W}_X(\pi, s) \leq \alpha \right\}.$$

NP-hardness

Definition (α-LCIWS2 problem)

Input: Two l-weighted sequences X, Y and a cut-off value $\alpha \in \mathbb{Z}_+$.
Output: The longest string $s \in SUBS(X, \alpha) \cap SUBS(Y, \alpha)$.

Definition (Partition problem)

Input: A finite set S, $S \subseteq \mathbb{Z}_+$.
Binary output: Is there a subset $S' \subseteq S$ such that $\sum S' = \sum S \setminus S'$.

NP-hardness

Theorem

LCIWS2 problem over a binary alphabet is NP-hard.

Proof.

For instance of Partition Problem, set $S = \{q_1, q_2, \ldots, q_n\}$ we construct l-weighted sequences $X = x_1 x_2 \ldots x_n$ and $Y = y_1 y_2 \ldots y_n$ over the alphabet $\Sigma = \{\mathtt{a}, \mathtt{b}\}$ with the following weights of letters from Σ:

$$w_i^{(X)}(\mathtt{a}) = q_i + c,\ \ w_i^{(X)}(\mathtt{b}) = c,\ \ \ w_i^{(Y)}(\mathtt{a}) = c,\ \ w_i^{(Y)}(\mathtt{b}) = q_i + c.$$

Here $c > 0$ is an arbitrary positive integer. Finally let $\alpha = \frac{1}{2} \sum S + nc$.

The Partition problem for an instance S has a positive answer iff the length of the solution to α-LCIWS2 for X and Y is n.

Approximation results

Theorem (Amir et al. 2010)

The α-LCWS problem can be solved in $O(n^3)$ time and $O(n^2)$ space. If we are only interested in the length of the output, the problem can be solved in $O(Ln^2)$ time, where L is the length of the solution.

Theorem

We can compute a solution to the α-LCWS2 problem for $X, Y \in \mathcal{WS}(\Sigma)$ of length at least $\lfloor \mathrm{OPT}(X, Y, \alpha)/2 \rfloor$ in $O(n^3)$ time and $O(n^2)$ space.

Proof idea

Solve α^2-LCWS in $O(n^3)$ time, and then extract a solution for α-LCWS2 of size $\lfloor \mathrm{OPT}(X, Y, \alpha)/2 \rfloor$.

Proof sketch

Let (s, π, π') be the solution of α^2-LCWS

$$\mathcal{P}_X(\pi, s) \cdot \mathcal{P}_Y(\pi', s) \; \geq \; \alpha^2. \tag{1}$$

We can split this solution to two parts. Let $g = \left\lfloor \frac{d}{2} \right\rfloor$. Obtaining partial probabilities:

$$A = \prod_{j=1}^{g} p_{i_j}^{(X)}(s_j), \quad B = \prod_{j=1}^{g} p_{i'_j}^{(Y)}(s_j),$$

$$C = \prod_{j=g+1}^{d} p_{i_j}^{(X)}(s_j), \quad D = \prod_{j=g+1}^{d} p_{i'_j}^{(Y)}(s_j).$$

Observe that only one of A, B, C, D can be smaller then α. So either (A, B) or (C, D) forms a solution with weight $\geq \alpha$.

Approximation results

Theorem

There exists a $(1/2)$-approximation algorithm for the α-LCWS2 problem which runs in $O(n^5)$ time and $O(n^2)$ space.

Proof.

Basically it is a consequence of previous lemma. To obtain the exact approximation ratio, we have to deal with the odd n case (this causes an $O(n^2)$ increase in the time complexity). $\square$

Approximation results: PTAS

Definition

Let $X, Y \in \mathcal{WS}(\Sigma)$, $n = \max(|X|, |Y|)$, and $\alpha \in (0, 1]$. We say that an instance (X, Y, α) of the α-LCWS2 problem is a (γ, T)-*power* if all the non-zero weights in the sequence X are powers of γ, where $0 < \gamma < 1$ and $\gamma^{T-1} \geq \alpha > \gamma^{T}$.

Lemma

The α-LCWS2 problem for (γ, T)-power instances can be solved in $O(n^3 T)$ time and space.

Proof idea

We can use dynamic programming.

Approximation results: PTAS

Algorithm details

Our approach is a generalisation of the standard LCS algorithm. We have $O(n^3T)$ states, each described by a tuple (a, b, ℓ, t), where:

- a is the position in the sequence X, $1 \leq a \leq n$;
- b is the position in the sequence Y, $1 \leq b \leq m$;
- ℓ is the length of the subsequence already chosen, $0 \leq \ell \leq m$;
- t is a γ-based logarithm of the product of $p_i(\sigma_j)$ values of the chosen subsequence of X; by the definition of the (γ, T)-power, we only consider integral values of t from the interval $[0, T-1]$.

Each state can be handled in $O(1)$ time.

Approximation results: PTAS

Lemma

For any $\epsilon > 0$ we can compute in $O(n^4/\epsilon)$ time and space a string which is an $\alpha^{1+\epsilon}$-subsequence of X and an α-subsequence of Y of length at least $\mathrm{OPT}(X, Y, \alpha)$.

Proof.

Let $T = \frac{n}{\epsilon}$ and $\gamma = \alpha^{1/T}$. For all i, j we set:

$$p'_i(\sigma_j) \;=\; \gamma^{\lfloor \log_\gamma (p_i^{(X)}(\sigma_j)) \rfloor}.$$

Use the algorithm from the previous lemma (note that the new weight p' is not a probability distribution, but the algorithm does not use that assumption). □

Approximation results: PTAS

Lemma

Let (X, Y, α) *be an instance of the LCWS2 problem. In* $O(n^5)$ *time and space one can find a string* s *which is an* $(\alpha, d-1)$*-subsequence of both* X *and* Y *such that no* $(\alpha, d+1)$*-subsequence of both* X *and* Y *exists.*

Proof.

Set $\epsilon = 1/n$ and use the algorithm from the previous lemma. Then remove a single character (which has the smallest value of $p^{(X)}_{i_k}(z_k)$). $\square$

Approximation results: PTAS

Theorem

For any real value $\epsilon \in (0, 1]$ there exists a $(1 - \epsilon)$-approximation algorithm for the LCWS2 problem which runs in polynomial time and uses $O(n^5)$ space. Consequently the LCWS2 problem admits a PTAS.

Proof

Using the algorithm from the previous lemma find a positive integer d and an $(\alpha, d - 1)$-subsequence.

- If $d \geq 1/\epsilon$ then we are done since in that case we have $(d - 1)/d = 1 - 1/d \geq 1 - \epsilon$ which means that we have found a $(1 - \epsilon)$-approximation.
- If $d < 1/\epsilon$ then we search for an (α, d)-subsequence using a brute-force approach, i.e., we try all $\binom{|X|}{d}, \binom{|Y|}{d}$ subsets of positions in each sequence.

Thank you for your attention!