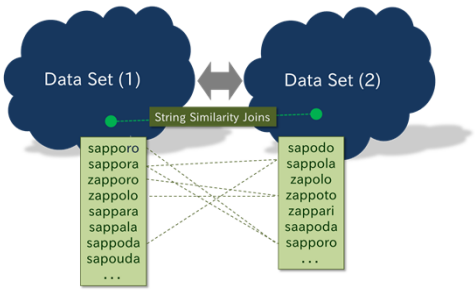
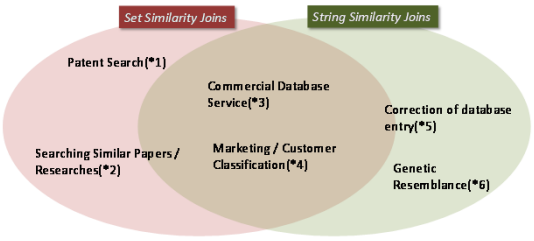
文字列類似結合におけるハイブリッド探索手法

JST-ERATO 湊離散構造処理系プロジェクト
白井康之, 高嶋宏之

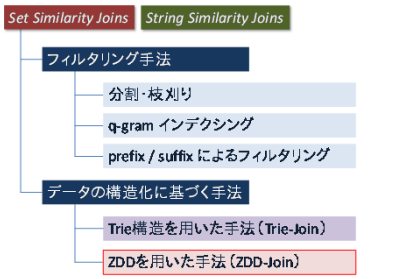
文字列類似結合

一定の条件下（編集距離等）でマッチする類似ペアをすべて列挙する.





(*1) Find similar patents in previous years, by other organizations
(*2) Find similar research papers in previous years, in other countries, or in other fields
(*3) Find similar records for cleaning and integration
(*4) Find potentially good customers, fraud data (person / transaction)
(*5) Find the misspelling of the person name, city name, ...
(*6) Find similar parts of gene sequences



	フィルタリング手法	Trie-Join (ZDD-Join)
レコード長	長い（ほうが得意）	短い（ほうが得意）
アイテム数	多い	少ない
制約（編集距離）	大きい	小さい

問題定義（文字列類似結合）

Input

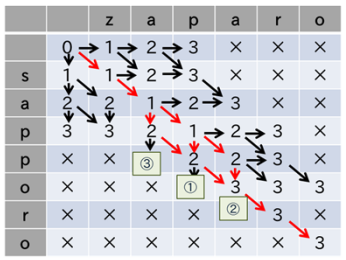
- シーケンスデータ集合 : R, S
  ex.) R={ “abc”, “bc” “def” }, S={ “ab”, “cd”, “efg” }
- r ∈ R, s ∈ S に対する類似尺度: sim(r,s)（編集距離）
  ex.) r1=“abc”, s1=“adc”, sim(r1,s1)=1
  r2=“abc”, s2=“acd”, sim(r2,s2)=2
- 閾値 : N（編集距離の最大値）

Output

- Rの部分集合 R’, s.t., ∀r∈R’, ∃s sim(r,s) ≤ N

動的計画法によるストリングマッチング

sapporo
zaparo　（δ=3）



① s a p p o r o
　z a p　 a r o

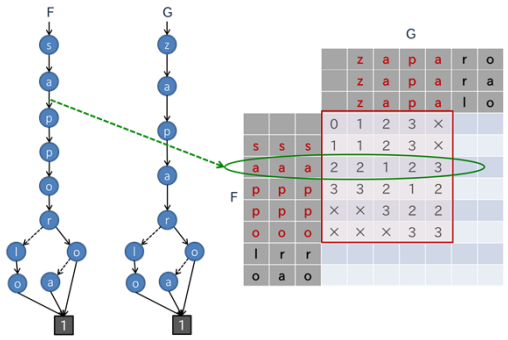
② s a p　p o　r o
　z a p　a　　r o

③ s a p p　o r o
　z a　 p a r o

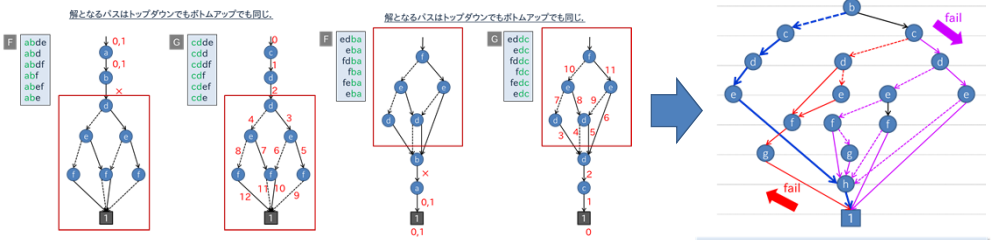
（赤字は replace, 青字は insert/delete）

ZDD（Seq-BDD）によるストリング類似結合

- Sequence BDD [Loekito 2009, Denzumi 2011, ほか]を用いて, ストリング集合間の類似結合を行う.
- f の各ステップでは, 動的計画法に基づき, 対応する g のパスとの編集距離を更新する.



ハイブリッド探索



トップダウンの方が効率的な例（δ=1）　　ボトムアップの方が効率的な例（δ=1）

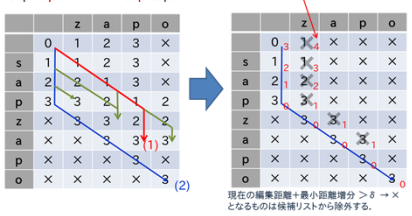
解の総数が非常に少ない場合, fail する部分集合をできるだけ早く見つけて伝搬させる必要がある.

初期状態としてトップとボトムから開始し, 最も候補数の少ない枝を都度選択して探索する.

Length Pruning

ex. 編集距離 = 3, {zapo, sapzapo}　　長さが大きく異なるデータセットでは, 有効

(1) zap o　　(2) zapo
sap zapo　　sap zapo

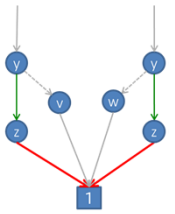


現在の編集距離＋最小距離増分 ＞δ → ×
となるものは候補リストから除外する.

冗長計算の削除

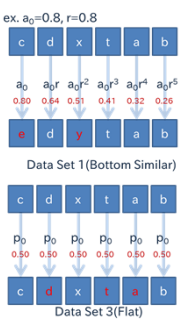
キャッシュを利用して, 冗長計算を排除.

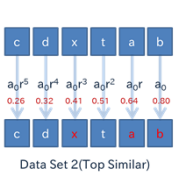
g でパスは異なるが, アイテム集合は同じもの



実験データ

ex. a_0=0.8, r=0.8

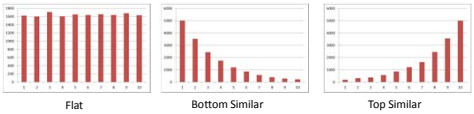




アルファベットのシーケンス
長さ = 5,10,12,15,18

マッチング結果（解の総数）がほぼ同数になるように確率値を調整

F: 10,000件, G:10000件とし, 解は10件程度, 5000件程度のものを対象



実験結果

10000件×10000件（解が少ないケース）

		編集距離	#Results	F側ノード探索数
random	top-down	1	159	33165
	bottom-up	1	159	34242
	hybrid	1	159	34757
Bottom Similar	top-down	1	183	29660
	bottom-up	1	183	45896
	hybrid	1	183	38286
Top Similar	top-down	1	206	45859
	bottom-up	1	206	31267
	hybrid	1	206	40977

10000件×10000件（解が比較的多いケース）

		編集距離	#Results	F側ノード探索数
random	top-down	1	4971	50118
	bottom-up	1	4971	50225
	hybrid	1	4971	51205
Bottom Similar	top-down	1	4863	44594
	bottom-up	1	4863	55289
	hybrid	1	4863	55160
Top Similar	top-down	1	4886	55261
	bottom-up	1	4886	45163
	hybrid	1	4886	47378

Length Pruning の効果

Length Pruning	編集距離	L=10-20		L=12-18		L=15	
		#Results	F側ノード探索数	#Results	F側ノード探索数	#Results	F側ノード探索数
無	1	3	305048	3	308074	3	311024
有	1	3	229610	3	243563	3	288917

今後の予定

- ボトムアップ手法とトップダウン手法の効率的な融合方法の検討（問題の類型化と制御方法の検討）
- 高速化