3N-5

確率的ストリームにおけるグループを用いたパターン問合せ

杉浦 健人[†]　早矢仕 新[†]　石川 佳治[†§]
[†] 名古屋大学大学院情報科学研究科　[§] 国立情報学研究所

1 はじめに

近年，確率的データストリームに対してパターンマッチングを行う研究が増えている [2, 3, 4]．確率的データストリームとは，複数のイベントの生起確率を各時刻毎に保持するストリームである．例えば図 1 の時刻 t_1 は，あるユーザが横になっている（lie）確率が 0.8，立っている（stand）確率が 0.1，歩いている（walk）確率が 0.1 だということを示している．しかし，確率的データストリームではその性質上，複数のマッチが同じ時間帯に存在するという問題がある．例えば，人間の行動モニタリングを行う場合を考える．ユーザが「横になっていた人が起き上がった」という行動を発見しようと，lie stand というパターンを記述したとする．図 1 のような確率的データストリームが得られた場合，$\mathtt{lie}_1$ $\mathtt{stand}_2$, $\mathtt{lie}_2$ $\mathtt{stand}_3$, $\mathtt{lie}_3$ $\mathtt{stand}_4$, $\mathtt{lie}_4$ $\mathtt{stand}_5$ のように，時刻 t_1 から t_5 にかけて四つのマッチが発見される．なお，$\mathtt{lie}_1$ は時刻 t_1 にイベントが lie であったことを示す．

時刻		t_1	t_2	t_3	t_4	t_5
イベント	lie	0.8	0.7	0.8	0.2	0.3
	stand	0.1	0.2	0.1	0.7	0.6
	walk	0.1	0.1	0.1	0.1	0.1

図 1: 確率的データストリームの例

既存の研究では，マッチの確率に閾値を設けてマッチ選別する手法 [2, 4] や確率の高い Top-k のマッチを選択する手法 [3] が提案されている．しかしそのように一部のマッチのみを選択するよりも，同じ時間帯に存在するマッチをひとまとめにして，統合された情報を提供した方が有益であると考えられる．そこで，本研究では時間帯の重なるマッチを**オーバラップ**するマッチとし，一つの**グループ**にまとめる手法を提案する．提案手法により，ユーザは発見したいパターンを記述することで，マッチをひとまとめにしたグループを受け取ることができる．なお，本研究ではパターンは正規表現で記述されるものとし，またパターンマッチを行う際はマッチ中の各イベントが必ず連続でなければならないとする．つまり，図 1 からパターン lie stand を発見しようとした場合，$\mathtt{lie}_1$ $\mathtt{stand}_3$ などは生成されず，上述した四つのマッチのみが生成される．

2 オーバラップとグループの定義

マッチをグループにまとめるための方針として，二つのオーバラップを定義する．あるマッチの集合 M が式 (1) または (2) を満たすとき，集合 M はそれぞれ**完全オーバラップ**または**部分オーバラップ**の性質を持つと定義する．なお，$ts\_overlap(m, m')$ は二つのマッチ m と m' の時区間に交わりがあるときに真となる述語記号である．

$$\forall m, m' \in M \ such \ that \ ts\_overlap(m, m') \quad (1)$$

$$\forall m \in M, \exists m' \in M \ such \ that \ m \neq m' \wedge ts\_overlap(m, m') \quad (2)$$

式 (1) と (2) が示す通り，完全オーバラップの性質を満たす集合は部分オーバラップの性質も満たす．つまり，部分オーバラップは完全オーバラップよりもマッチを統合する条件を緩くした方針である．ユーザはパターンマッチングを行う際に，その要求や扱うストリームデータに応じて，完全オーバラップと部分オーバラップのどちらを使用するか決定する．

マッチの集合 G が完全オーバラップまたは部分オーバラップの性質を満たす極大な集合であるとき，集合 G を**グループ**と呼ぶ．なお，極大な集合とは，任意のマッチ m を集合 G へ加えたときオーバラップの性質が満たされなくなる集合を指す．1 節で使用した例を用いて，グループの例を示す．四つのマッチをそれぞれ m_1, m_2, m_3, m_4 としたとき，完全オーバラップでは $\{m_1, m_2\}$, $\{m_2, m_3\}$, $\{m_3, m_4\}$ の三つのグループが，部分オーバラップでは $\{m_1, m_2, m_3, m_4\}$ という一つのグループが生成される．

3 グループ確率の計算方法

グループの確率を計算するために，まずマッチの確率について考える．マッチ $m = \mathtt{a}_1\ \mathtt{a}_2\ ...\ \mathtt{a}_n$ が得られたとき，マッチ確率 $\Pr(m)$ は式 (3) で計算される．なお，$\Pr(\mathtt{a}_i)$ はイベント $\mathtt{a}_i$ の生起確率を表す．

$$\Pr(m) = \prod_{i=1}^{n} \Pr(\mathtt{a}_i) \quad (3)$$

次に，グループの確率について考える．グループはマッチをひとまとめにしたものであるため，グループ中のマッチのいずれかが存在する確率がグループの確率として妥当である．具体的に，2 節で完全オーバラップの例として使用したグループ $\{m_1, m_2\}$ の確率を考えてみる．m_1 と m_2 は時刻 t_2 において異なるイベント $(\mathtt{stand}_2, \mathtt{lie}_2)$ を選択しているため，この二つのマッチが同時に存在することはない．従って，$\Pr(m_1) = 0.16$ と $\Pr(m_2) = 0.07$ を足しあわせた 0.23 がグループ $\{m_1, m_2\}$ の確率となる．

上記の例では，グループ中に同時に存在するマッチが無いため簡単にグループの確率を計算できた．しかし実際には，m_1 と m_3 のように二つのマッチが同時に存在する可能性も考慮する必要があるため，単純に

Pattern Queries Using Groups over Probabilistic Streams
Kento Sugiura[†], Arata Hayashi[†], Yoshiharu Ishikawa[† §]
[†] Graduate School of Information Science, Nagoya University
[§] National Institute of Informatics

マッチ確率の総和をグループの確率とすることはできない．そもそも，確率的ストリームでは大量のマッチが生成されうるため，すべてのマッチ確率を計算すると計算量が非常に大きくなってしまう．そこで，与えられたパターンをオートマトンで表し，オートマトンの各状態にいる確率を計算することでグループの確率を求める手法を提案する．なお，本稿の残りでは部分オーバラップを使用した場合の例についてのみ考えるが，完全オーバラップでも同様の方法で処理できる．

図 2 にパターン `lie stand` をオートマトンで表した例を示す．なお，$\overline{\texttt{a}}$ は a 以外のイベント，`a | b` は a または b どちらかのイベント，*は任意のイベントを表す．単にパターンを発見するだけであればエッジは状態 0 から 1 及び 1 から 2 へ遷移する二本のみであるが，図 2 ではグループ確率を計算するためにいくつかのエッジを加えている．状態 2 から 2 へ遷移するエッジは受理状態に達したマッチを保持するための，状態 1 から 1 へ遷移するエッジはパターンの最初のイベントが連続する場合に対応するためのものである．また，状態 0 から 0 及び 1 から 0 のエッジは，パターンに一致しないとき初期状態へ戻ることを示す．

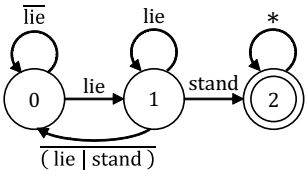

図 2: グループ確率計算用オートマトンの例

実際に，図 2 のオートマトン及び図 1 のストリームを使用して，部分オーバラップによって生成されたグループ $\{m_1, m_2, m_3, m_4\}$ の確率を計算した例を図 3 に示す．なお，t_0 はパターンマッチが行われる前に，初期確率がどのように分布しているかを表す．図 3 の各列は，ある時刻に各状態まで到達した確率を示す．例えば，時刻 t_3 に状態 1 へ到達する確率を考える．時刻 t_3 に状態 1 へ到達するには，時刻 t_2 に状態 0 または 1 に到達し，かつ時刻 t_3 にイベント `lie` を選択しなければならない．従って，その確率は $(0.14+0.70)*0.80$ つまり 0.672 となる．このような計算をすべての時刻及び状態に対して行い，最終的に時刻 t_5 に受理状態（状態 2）に到達している確率がグループの確率となる．つまりこの例の場合，グループ $\{m_1, m_2, m_3, m_4\}$ の確率は約 0.79 となる．

時刻		t_0	t_1	t_2	t_3	t_4	t_5
状態	0	1	0.2	0.14	0.098	0.1456	0.11732
	1	0	0.8	0.70	0.672	0.1540	0.08988
	2	0	0	0.16	0.230	0.7004	0.79280

図 3: グループ確率の計算例

4 グループの肥大化に対する処理

グループをまとめるための方針として部分オーバラップが選択されたとき，入力となるストリームによってはグループ中に含まれるマッチが非常に多くなってしまう．例えば，図 1 のように `lie` と `stand` の確率が 0 より大きい値を持ち続けるとき，ストリームが続く限りグループにマッチが追加されていく．そこで，このような問題に対応するための処理方法を二つ提案する．

まず一つ目は，窓（window）を利用した方法である．データストリーム管理システムでは，ストリームを有限の長さに区切るための仕組みとして滑り窓が使用される [1]．例えば，図 1 において現在時刻が t_5 のときウィンドウサイズが 4 に設定されたとすると，時刻 t_1 のデータは破棄され時刻 t_2 から t_5 のデータが処理の対象となる．パターン `lie, stand` に対して部分オーバラップでグループを生成する際に，滑り窓の処理を適応した場合について考える．現在時刻が t_4 のとき，生成されるグループは $\{m_1, m_2, m_3\}$ である．一方，現在時刻が t_5 になったとき，時刻 t_1 のデータは破棄され，生成されるグループは $\{m_2, m_3, m_4\}$ となる．このように，窓によってストリームを区切ることで，グループを制限することができる．

二つ目は，グループ確率に閾値を設定する方法である．ユーザにグループ確率の閾値，つまり要求する信頼度を指定させ，グループ確率が閾値以上になった時点でグループを確定し出力する．この方法は，ユーザの要求する信頼度を超えてグループにマッチを追加する必要はないという考えに基づいている．例えば，図 3 の例においてグループ確率の閾値を 0.5 に設定した場合，時刻 t_4 の時点で確率が 0.5 を超えるため，時刻 t_5 を待たずにグループ $\{m_1, m_2, m_3\}$ が出力される．

5 まとめ

本稿では，確率的ストリームに対してパターンマッチを行う際に，時間帯の重なるマッチをグループとして統合する手法を提案した．グループとしてまとめるために完全オーバラップと部分オーバラップの二つを定義し，オートマトンを利用してグループの確率を計算する方法を提案した．また，グループ中のマッチが際限なく増えることを防ぐための処理方法を二つ提案した．

謝辞

本研究の経費の一部は科学研究費（23650047, 25280039）および内閣府最先端研究開発プロジェクト（FIRST）による．

参考文献

[1] Brian Babcock, Shivnath Babu, Mayur Datar, Rajeev Motwani, and Jennifer Widom. Models and issues in data stream systems. In *Proc. ACM PODS*, pp. 1–16, 2002.

[2] Zheng Li and Tingjian Ge. Online windowed subsequence matching over probabilistic sequences. In *Proc. ACM SIGMOD*, pp. 277–288, 2012.

[3] Zheng Li, Tingjian Ge, and Cindy X. Chen. ε-matchingg: Event processing over noisy sequences in real time. In *Proc. ACM SIGMOD*, pp. 601–612, 2013.

[4] Christopher Ré, Julie Letchner, Magdalena Balazinksa, and Dan Suciu. Event queries on correlated probabilistic streams. In *Proc. ACM SIGMOD*, pp. 715–728, 2008.