センサデータに対する問合せ高速化のための索引の実装

中島 沙季　(指導教員：渡辺 知恵美)

1　はじめに

近年，ハードディスクの低価格化が進み，大容量のハードディスクが安価で手に入るようになり，大量のデータを保存することが可能になっている．そこで，センサデータのような，常に新しい情報が到着し続けるストリームデータを全て格納することが容易になってきている．

しかし，センサから取得した情報を基に,「いつドアが開いたか」,「人が部屋のどの位置にいるか」などを知りたい場合には，不便だと考えられる．短時間で大量に生成されるストリームデータに対して問合せを行うことは時間がかかりすぎてしまい，困難であるからである．

そこで本研究では， Ocha House [1] 向けに，大量のセンサデータに対して問合せを行った際に，高速な問合せが可能となるような，索引の作成を行う．

2　センサデータの格納

実験住宅 Ocha House には，現在約 40 個のセンサが設置されている．天井やキッチン，ドアなどに赤外線センサ，位置センサなどを設置し，モニタリングを行っている．そして，設置しているそれぞれのセンサから取得したデータが，実験住宅内にある PC に無線で送信され，送られてきたセンサデータを全て DB に格納している．

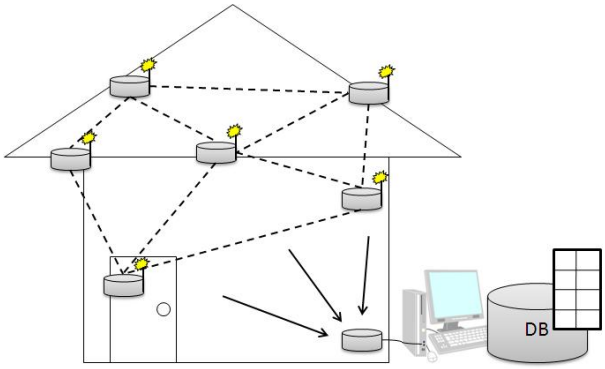

図 1: センサデータを DB に格納

短時間で大量に到着するセンサデータを全て DB に格納する利点としては，格納したデータを後で使いたいときに便利であるという点である．例えば，後々データマイニングを行うときに便利だと言える．また,「最近 1ヵ月間のドアの開閉を行った時刻を知りたい」といった，過去のデータに対して問合せを行う要求があったときにも，データを全て格納しておくことは有効であると考えられる．

問題点としては，約 40 個ものセンサから常にデータが送られてくるので，データ量が多くなってしまうという点である．センサ 1 個につき，1 日で約 40MB，1 年で約 15GB ものデータ量になり，センサ 40 個では 1 日で約 1.6GB，1 年で約 600GB ものデータ量になる．よって，このテーブルに対して問合せを行おうとすると，タプル数がとても多く，問合せに時間がかかってしまう．

3　索引の作成

問合せを高速に行えるようにするために，ストリームデータに対する索引テーブルを作成し，問合せを行う際は，索引テーブルに対して行うことにした．

3.1　SAX によるセンサデータの文字列化

時系列データの圧縮方法としては， FFT を用いた圧縮，ウェーブレット変換を用いた圧縮など数多くの手法が知られているが，本研究では， SAX : Symbolic Aggregate approXimation [2][3] を用いたデータ圧縮を行うことにした．SAX とは， Lin らが提案した時系列データ表現手法の 1 つであり，データを文字列で表現するという特徴を持っている．それによって時系列データに対し自然言語処理のアルゴリズムを適用できるというメリットがある．

時系列データの文字列化の手順としては，

(1) 時間軸を等間隔に区分.
(2) 区間ごとに平均値を算出.
(3) 正規分布に従って，図 2 のように a，b，c…とアルファベットを割り振り，分割線を定める．そして，(2) で求めた平均値を文字列に変換（図 2）.

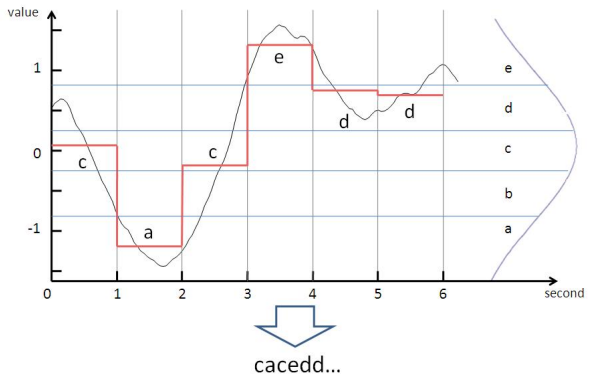

図 2: SAX

このように SAX の手法を使うことで，データを文字列に変換し圧縮することが可能になる．

3.2　ランレングス圧縮による文字列の圧縮

時系列データを SAX を用いて文字列化を行うと，非常に長い文字列になる．そこで， SAX を用いて変換した文字列に対し，さらにランレングス圧縮を用いて圧縮し，短い文字列に変換することにした．

ランレングス圧縮とは，データ圧縮方式のことで，データの中に同じ符号が連続して並んでいる場合に，その符号と個数によって表現することで圧縮する手法である．例えば,「AAABBBBCCCCC」というデータでは，最初の「A」は 3 個連続して並んでいるためこれを「3A」と表す．次に「B」が 4 個並んでいるのでこれを

「4B」と表す．この操作を繰り返し行うと,「3A4B5C」という 6 文字のデータに圧縮することができる．このようにランレングス法を使うことで，文字列化したデータをさらに圧縮することが可能になる．

3.3 索引テーブルの作成

索引テーブルの作成の手順としては,

(1) 全てのデータを格納したテーブルから 1 日分のデータを抽出.
(2) 抽出したセンサデータを SAX によって文字列化.
(3) ランレングス圧縮によって文字列の圧縮.
(4) 索引テーブルに格納.

センサデータは, 1 秒に 15 回送られてきており，Sensordata テーブルに格納されている．Sensordata テーブルは， ID ，センサ値，日付・時刻の属性を持っている．そして， Sensordata テーブルのデータを圧縮して作成する索引テーブルは，ID ，日付，センサ値の文字列の属性を持っている（図 3）.

Sensordata

id	value	time
1	-0.13	2010-01-01 10:10:10
1	0.01	2010-01-01 10:10:11
1	-0.57	2010-01-01 10:10:12
...	...	...

索引

id	date	indexvalue
1	2010-01-01	50abc6de...
1	2010-01-02	ad275eda...
1	2010-01-03	b2d56e8b...

図 3: テーブル例

毎日 1 回， DB に格納してある Sensordata テーブルのデータを SAX を用いて圧縮し，索引テーブルに 1 タプル分の情報を追加していく.

3.4 問合せ

索引テーブルへの問合せの方法を示す．例として，玄関のドアに加速度センサを設置し，ドアの開閉データを取得している場合を想定し,「1 月 1 日に，ドアの開閉を行った時刻を知りたい」といった問合せについて考える．手順としては,

(1) ドアの開閉 1 回分の動きを調べて，SAX を用いて文字列に変換する.
(2) 索引テーブルの 1 月 1 日のタプルを参照し，格納している文字列が，(1) で求めた文字列と同じになっている箇所がないか探す．この際，ランレングス法を用いて圧縮した文字列を復号化せずに検索する.
(3) 見つかったら，前から何文字目にあったか調べ，そのときの時刻を計算する．すると，1 月 1 日に，ドアの開閉を行った時刻を知ることができる.

4 実験

センサデバイスとして， Sun Microsystems 社の研究開発組織であるサン・ラボで開発された Sun SPOT [4] を使用し，実験を行った．[5]

研究室のドアに加速度センサを 1 個設置し，ドアの開閉データを取得した.

4.1 データ量の計測

Sensordata テーブルと，作成した索引テーブルのデータ量の計測を行った.

Sensordata テーブルは，センサ 1 個につき 1 日で約 40MB であった．そこから算出すると，1 年で約 15GB，Ocha House に設置している 40 個のセンサのデータを全て格納すると，1 年で約 600GB になる.

作成した索引テーブルは，センサ 1 個につき 1 日で約 5KB であった．そこから算出すると，1 年で約 2MB ，Ocha House に設置している 40 個のセンサのデータを全て格納しても，1 年で約 80MB ほどである.

4.2 問合せ時間の計測

Sensordata テーブルと，作成した索引テーブルの問合せ時間の計測を行った.

Sensordata テーブルに 1 週間分のデータを格納し，問合せを行ったところ， 24.2 秒ほどの時間がかかった．そして，データを人工的に増やし， 1 か月分のデータに対して問合せを行ったところ，1 分 52 秒かかった.

しかし，作成した索引テーブルに 1 週間分のデータを格納し，問合せを行ったところ，0.02 秒しかかからず，1 年間分の人工データに対して問合せを行った際も，かかった時間は 0.05 秒ほどであった.

Sensordata

1週間分	1ヶ月分
24.2 sec	1min 52 sec

索引

1週間分	1ヶ月分	1年分
0.00 sec	0.02 sec	0.05 sec

図 4: 問合せ時間の比較

5 まとめ

本稿では，センサデータを収集・格納し，格納したデータへの問合せ高速化のための索引の作成について述べた．また，実際に研究室に設置した加速度センサのデータを格納し，索引の作成を行い，データ量・問合せ時間の比較を行った.

今後は，加速度センサのドアの開閉データへの問合せの際に，人によって加速度が変わってしまうので，検索方法を工夫していく.

参考文献

[1] Ocha House : http://ochahouse.com/

[2] Jessica Lin , Eamonn Keogh , Stefano Lonardi , Bill Chiu : ” A Symbolic Representation of Time Series, with Implications for Streaming Algorithms”

[3] Sorabh Gandhi , Suman Nath , Subhash Suri , Jie Liu : ” GAMPS : Compressing Multi Sensor Data by Grouping and Amplitude Scaling ”, SIGMOD2009

[4] Sun SPOT : http://jp.sun.com/products/software/sunspot/

[5] 川口菜々，小口正人 :”センサネットワークを用いたストリームデータ処理実験環境の構築”, DEIM2009