Information Systems 39 (2014) 290–304

Contents lists available at ScienceDirect

Information Systems

journal homepage: www.elsevier.com/locate/infosys

# Approximation trade-offs in a Markovian stream warehouse: An empirical study ☆

J. Letchner [a,*], M. Balazinska [b], C. Ré [c], M. Philipose [d]

[a] Microsoft Corporation, Bellevue, WA, United States
[b] University of Washington, Seattle, WA, United States
[c] University of Wisconsin, Madison, WI, United States
[d] Intel Research, Seattle, WA, United States

ARTICLE INFO

Available online 24 April 2012

*Keywords:*
Streams
Imprecision
Uncertainty
Approximation
Compression

ABSTRACT

A large amount of the world's data is both *sequential* and *low-level*. Many applications need to query higher-level information (e.g., words and sentences) that is inferred from these low-level sequences (e.g., raw audio signals) using a model (e.g., a hidden Markov model). This inference process is typically statistical, resulting in high-level sequences that are imprecise. Once archived, these imprecise streams are difficult to query efficiently because of their rich semantics and large volumes, forcing applications to sacrifice either performance or accuracy. There exists little work, however, that characterizes this trade-off space and helps applications make an appropriate choice.

In this paper, we study the effects – on both efficiency and accuracy – of various stream approximations such as ignoring correlations, ignoring low-probability states, or retaining only the single most likely sequence of events. Through experiments on a real-world RFID data set, we identify conditions under which various approximations can improve performance by several orders of magnitude, with only minimal effects on query results. We also identify cases when the full rich semantics are necessary. This study is the first to evaluate the cost vs. quality trade-off of imprecise stream models.

We perform this study using Lahar, a prototype Markovian stream warehouse. A secondary contribution of this paper is the development of query semantics and algorithms for processing aggregation queries on the output of pattern queries—we develop these queries in order to more fully understand the effects of approximation on a wider set of imprecise stream queries.



## 1. Introduction

People and computers worldwide generate exabytes of audio, video, text, GPS,[1] RFID,[2] and other types of multimedia and sensor data—and because disk storage is cheap, most of this data is archived for future use [2]. These information-rich archives are poised to revolutionize data-centric applications in diverse areas including patient and asset tracking in hospitals [3], activity monitoring for elder care [4], scientific environment observation [5], e-Learning [6], phone conversation mining, and multimedia search/retrieval.

While some applications can use raw sensor or multimedia streams directly [7,8], most rely on higher-level streams *inferred* from the low-level data. Search engines, for example, can index audio files by content only after these files have been translated into text. Similarly, location

☆ This paper is an extended version of [1].

* Corresponding author. Tel.: +1 206 890 8733.
*E-mail addresses:* letchner@gmail.com (J. Letchner), magda@cs.washington.edu (M. Balazinska), chrisre@cs.wisc.edu (C. Ré), matthai.philipose@intel.com (M. Philipose).

[1] Global positioning system.
[2] Radio frequency identification.

0306-4379/$ - see front matter © 2013 Published by Elsevier Ltd.
http://dx.doi.org/10.1016/j.is.2012.04.003

tracking or activity monitoring applications require that raw sensor streams be transformed into location or activity sequences, respectively, before they are processed. Due to noise in the data or ambiguity in the inference process (or both), these inferred, high-level streams are *imprecise* (e.g., a spoken word might be either "eight" or "ate"; while an RFID reading might only narrow a person's location down to one of several adjacent rooms).

The current state of the art for supporting imprecise sequences is the model-based view [9]. A model-based view allows applications to query data as if it were deterministic; internally, however, the DBMS answers the query on the imprecise sequence and returns results annotated with appropriate confidence scores. Model-based sequence views are most commonly used to represent imprecise location streams, typically inferred from GPS [10] or RFID data [11–14]. They are also used to model environmental statistics (temperature, light levels, *etc.*) inferred from distributed sensor networks [15,5,16], wildlife population counts inferred from sparsely-deployed habitat sensors [17], and structured language information inferred from written text (i.e. information extraction) [14].

A model-based view decouples the queried view from the model used to represent the underlying imprecise sequences, giving a DBMS designer considerable flexibility in choosing an appropriate model. The simplest model, called MAP in the AI literature [18], represents the imprecise stream using only a single deterministic sequence (e.g. the most likely path a person took through a building) [19,20]. A slightly richer model might represent uncertainty within each individual sequence element (e.g. distributions over a person's uncertain location at each timestep) [21,18], while an even richer model might additionally represent correlations between these uncertain values (e.g. distributions over entire paths through a building) [12,11,17]. An orthogonal design question involves the level of detail at which the chosen model is expressed. In the location domain, for instance, imprecision might reflect uncertain values at the level of every room, or might instead model many rooms as the same entity, using a single label (e.g. "Office" or "Lab").

In general, model choice has a significant impact on DBMS quality and performance: increased complexity yields higher accuracy, but incurs additional computational and I/O costs. The appropriate model choice depends on the application. As an example, consider an RFID-based tracking infrastructure deployed in a hospital. An application that detects equipment theft requires high performance but does not need high precision. False positives are tolerable. In contrast, an application that identifies all the locations visited by a potentially infected piece of equipment or person can tolerate slower performance but requires high precision.

The appropriate model choice is complicated by the fact that applications generally require support for complex queries, including *event queries* [19,20] (e.g. *Find all times in May when Bob entered the coffee room.*'), and *aggregated event queries* (e.g. *How many people entered the coffee room each day in May?*) [12]. Such queries are expensive to compute on sequential, imprecise data, where the utility of scalability techniques like indexing and compression is limited. These high processing costs naturally raise the question of whether rich imprecise sequence models are worthwhile. Would applications notice a difference in result quality if rich, imprecise streams were approximated using simple, deterministic ones? What performance benefits could be gained from such an approximation, which would allow any of several high-performance, deterministic stream processing engines [19,20] to be leveraged? How might a system achieve a flexible trade-off between the accuracy and efficiency of imprecise sequence processing?

In this paper, we address these questions using an empirical study of several common Markovian stream approximations. We show using examples and a brief theoretical analysis that worst-case error bounds on these approximations are too large to be of practical consequence; however, we demonstrate that in practice, errors are often orders of magnitude smaller than these bounds would indicate. We report both performance and accuracy results on real-world location sequences inferred from an office-building RFID deployment. We provide heuristics to guide model choice for various types of query, and finally we generalize our results beyond our application domain by identifying the underlying properties of our data that are responsible for the effects that we measure.

We perform our study using the Lahar Markovian stream warehouse prototype [12,11]. Lahar natively supports Markovian model-based sequence views, which are the richest of the models used in practice [22–24, 12,11,17]. With simple modifications, Lahar also supports the approximations that we study here. As part of this study, we additionally augment Lahar to support novel aggregated-event queries. The semantics and processing algorithms for these queries are a secondary contribution of this paper, developed to support our primary goal of studying the effects of approximation on a wide variety of imprecise stream queries.

Through our study, we find that the accuracy/performance trade-off space is rich. A sample of the trade-offs achieved using approximation are shown in Fig. 1. As the figure illustrates, in our study, we identify two models – called independence and MAP – that accelerate performance by 1 and 2 orders of magnitude respectively. This performance gain, however, comes at the cost of precision. While these approximations lead to typically low errors ($10^{-3}$ or lower in the figure), occasionally the errors can be high (0.1 or higher in the figure). We return to this figure in Section 5.4. These models are thus best suited for performance-critical applications. We identify two additional models – thresholding and rollups – that are best suited for applications that prioritize accuracy. These models accelerate performance only moderately, but consistently return errors that are either zero or so close as to be negligible. Interestingly, we find that richer stream models do not always produce better accuracy than simple models. In particular, the accuracy of the two high-performance models varies significantly based on query characteristics: the simpler (and faster) of the two models consistently outperforms the more complex model on some types of aggregate-event query, while on other query types the more complex approximation achieves better accuracy, as one would expect.

The remainder of this paper is structured as follows: In Section 2, we introduce the Markovian stream model and