To be published in IEEE ICASSP 2006, Toulouse, France, May 2006.

SIGNAL PROCESSING CHALLENGES IN DISTRIBUTED STREAM PROCESSING SYSTEMS

Pascal Frossard[*†], *Olivier Verscheure*[‡], *and Chitra Venkatramani*[‡]

[†]Ecole Polytechnique Fédérale de Lausanne (EPFL), Signal Processing Institute, CH-1015 Lausanne
[‡]IBM T.J. Watson Research Center, Yorktown Heights, NY 10598

ABSTRACT

Distributed stream processing represents a novel computing paradigm where data, sensed externally and possibly preprocessed, is pushed asynchronously to various connected computing devices with heterogeneous capabilities for processing. It enables novel applications typically characterized by the need to process high-volume data streams in a timely and responsive fashion. Some example applications include sensor networks, location-tracking services, distributed speech recognition, and network management. Recent work in large-scale distributed stream processing tackle various research challenges in both the application domain as well as in the underlying system. The main focus of this paper is to highlight some of the signal processing challenges such a novel computing framework brings. We first briefly introduce the main concepts behind distributed stream processing. Then we define the notion of relevant information from two related information-theoretic approaches. Finally, we browse existing techniques for sensing and quantizing the information given the set of classification, detection and estimation tasks, which we refer to as task-driven signal processing. We also address some of the related unexplored research challenges.

1. INTRODUCTION

With the widespread use of digital systems, there is a large set of emerging applications that perform operations such as filtering, aggregation and correlation over high-volume, unbounded, continuous data such as documents, email, instant messages, transactional data, digital audio, video and image data, network packet traces, and sensor data. These applications process the streaming data in the context of a larger information mining system that tracks information relevant to a large body of long-running, *continuous* queries on these streaming data sources. Each of these applications can be viewed as a processing pipeline which analyzes data from a set of raw data sources to extract relevant information.

Such applications provide challenges both in the application domain in terms of effective application composition for distributed processing, and in the underlying system in terms of a wide range of resource requirements. For instance, on the data-ingest end, the system has to deal with very high-throughput data containing a lot of irrelevant information which can be eliminated with limited processing. Deeper into the processing pipeline however, data volume is possibly reduced, but processing complexity is significantly increased as deeper classification and detection operations are performed on data aggregated over one or more streams.

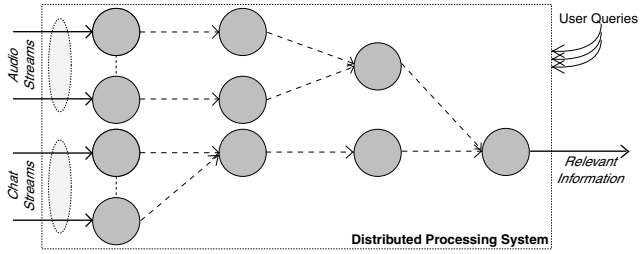

Fig. 1. Sample stream processing graph: The nodes are the application operators and the edges represent the streams. The application operators are running on various, possibly remote, processing devices.

Consider an application that is interested in analyzing and correlating information from a large set of audio and chat streams. In this case, early stages of processing may involve eliminating non-speech data from large volumes of audio data via fast classification techniques applied to the audio energy extracted directly from the compressed domain, and later stages would involve converting the audio data into text (i.e. speech recognition) and performing deeper analysis on the transcribed text (Natural Language Processing, NLP) [1]. This streaming application can be represented as a graph as shown in Figure 1, where the nodes are the application operators and the edges represent the streams. Note that the application operators are instantiated on various, possibly remote, processing devices.

The new paradigm of distributed stream processing addresses these application characteristics and provides an elegant framework for enabling such challenging applications. Section 2 highlights the key features of such systems and introduces related work in the area. It also defines the notion of relevant information from two related information-theoretic approaches. Relevant information plays a key role in task-driven signal processing. Section 3 browses some of the existing sensing and quantization strategies that factor in the set of classification, detection and/or estimation tasks processing the sensed, possibly compressed, information. It addresses a few unexplored research challenges from a signal processing perspective, as generated by such a novel computing paradigm. Concluding remarks are given in Section 4.

2. INFORMATION PROCESSING

Large-scale streaming applications are enabled by an underlying distributed stream processing system, which is the topic of Section 2.1.

*This work has been partly supported by the Swiss National Science Foundation, under grant PP-002-68737. Contact author: pascal.frossard@epfl.ch

These applications define the relevant information that the sensing and quantization algorithms should factor in. The notion of relevant information is described in Section 2.2.

2.1. Distributed Stream Processing Systems

A distributed stream processing system provides the architectural substrate and services to enable the requirements of stream processing applications, where users submit *continuous* queries that are evaluated over streams of data. The key challenges these systems address include highly varying processing and throughput requirements across the processing pipeline, low response times when relevant information is detected, scalability to support millions of simultaneous queries, and widely distributed computing.

There has been a lot of recent work on distributed stream processing systems, such as Aurora [2], Borealis [3] and Telegraph CQ [4]. Distributed stream processing systems provide the framework that deploys and runs stream-processing applications on various resource topologies and delivers results to the users. Most of the current systems take a database-centric approach where relational operators are applied to streaming data. Since relational operators are well-understood, the system can statically determine the optimal ordering and placement of operators over the available resource topology. Although some systems like Telegraph CQ provide dynamic adaptivity to available resources, they do not factor in application knowledge to achieve the best resource-to-accuracy tradeoffs. System S [5] makes an attempt in this direction by additionally providing hooks to the applications to determine current resource usage so that they can use it to adapt suitably to achieve the best performance with the available resources. As the body of stream processing applications is growing rapidly, these key issues must be addressed more systematically.

In this paper we focus on identifying and extracting relevant information via sensing and quantization given downstream classification, detection and estimation tasks. This is referred to as *task-driven signal processing*. Other important issues briefly discussed include optimal composition of the processing graph and adapting processing during runtime so as to achieve the best resource-accuracy tradeoffs (see Section 3).

2.2. Relevant Information

Identifying the information that is relevant to the set of classification and estimation operations performed downstream is of crucial importance so as to overcome the scalability challenge that distributed processing systems are facing. Indeed tremendous savings in network bandwidth, disk I/O and computing resources can be achieved by filtering out as much of the useless information as possible; thereby possibly increasing the overall accuracy of the system, which would otherwise blindly drop data to accommodate the available resources.

The notion of *relevant information* can be defined from two related information-theoretic approaches depending on whether or not a distortion measure is available.

Relevance through distortion: Rate distortion theory is the branch of information theory addressing the problem of determining the minimal amount of entropy (or information) R that should be communicated over a channel such that the source X can be reconstructed at the receiver $\hat{X}$ with given distortion $D(X, \hat{X})$. As such, rate distortion theory gives theoretical bounds for how much compression can be achieved using lossy data compression methods. Rate distortion theory determines the level of inevitable expected distortion given the desired information rate in terms of the rate distortion function. The most common distortion measure is the mean square error (MSE), which represents the distortion incurred in reconstructing the source from its compact representation. However, it has long been recognized that MSE is not the most pertinent distortion measure when one is interested in the effect of compression on decision making performance, rather than signal reconstruction. Various researchers have proposed different distortion measures for assessing compression algorithms relative to detection, classification and other decision objectives in the rate distortion theory framework (see following subsections). Others have taken a different approach by alleviating the need to *explicitly* specify a distortion function.

Relevance through another variable: Tishby *et al.* [6] proposed a principled information-theoretic approach by introducing an additional variable Y that determines what is relevant. The *information bottleneck* (*IB*) method. The *IB* method is a general non-parametric clustering framework. Given a joint distribution $p(x, y)$ of two (random) variables X and Y, it attempts to extract the relevant information $\hat{X}$ that X contains about Y. In [6] it is argued that both the compactness of the representation and the preserved relevant information are naturally measured by the symmetric *mutual information* $I(.;.)$, hence the above principle can be formulated as a trade-off between these quantities. The *IB* problem can be stated as finding a (stochastic) mapping $p(\hat{x}|x)$ such that the IB-functional $\mathcal{L} = I(\hat{X}; X) - \beta I(\hat{X}; Y)$ is minimized, where β is a positive Lagrange multiplier that determines the trade-off between compactness of the representation and its precision. It was shown that this problem has an exact optimal (formal) solution without any assumption about the origin of the joint distribution $p(x, y)$ (see [7] for further information) but may suffer for multiple minima (non-global convergence) [8]. This method has been successfully employed in several applications such as document categorization [9, 10], phoneme and speaker recognition [11], and image clustering [12].

3. TASK-DRIVEN SIGNAL PROCESSING

The set of processing operations in a distributed stream processing system is often not explicitly considered. This usually leads to inefficient processing of the information captured by the distributed sensors, or even an inefficient acquisition of the information. From a signal processing perspective, this question should clearly be the driver for the algorithms used to best sense the environment and optimally quantize the sensed information. We refer to such a framework as *task-driven signal processing*.

This section examines the problem of extracting a relevant summary of the information in the context of sensing and quantization strategies given downstream classification, detection and estimation processing operators. The 'Discussions' section addresses some of the related unexplored research challenges.

3.1. Sensing

Data acquisition plays a central role in capturing the relevant information. Indeed the sensed information may be useless (or not as useful) if the characteristics of the task that processes it are not factored in the sensing process itself.

A sensor network consists of a collection of sensor nodes distributed over a geographic area [13]. Each node has one or more sensing devices (e.g., microphone, thermometer, camera), a wireless communication device, simple processing capability, and a limited energy supply. Though each node is an independent hardware device, they must coordinate their sensing strategy in order to acquire

the essential information about their environment, which is relevant to a set of downstream tasks processing the sensed data.

Sensors typically provide the capability to self-organize their sensing and processing networks, possibly in a hierarchical way. Self-organizing sensor networks may be built from sensor nodes that may spontaneously create impromptu network, assemble the network themselves, dynamically adapt to device failure and degradation, and react to changes in task and network requirements. Some sensors are even capable of spatially organizing themselves (mobility). The gateway node aggregates the (possibly pre-processed) data from the various sensors (or cluster heads in a hierarchical network) and streams it to distributed stream processing systems.

Task-driven sensing [14] has been the subject of recent research work. For example, the problem of counting the number of people in the workspace to monitor unusual activities via information gathered by visual sensors has been addressed in [15]. Extracting the relevant information from visual sensors is critical because of the amount of data each video camera captures. The authors solve a clustering problem resulting from a well-defined distortion measure. In a more recent example [16], the authors proposed a distributed detection algorithm applicable to highly decentralized architectures and showed that its performance were greatly improved by introducing a small degree of randomness in the underlying connectivity, with which the network attains desirable small-world characteristics. Finally, researchers have also addressed enhancing the teleconference experience using readily available microphone arrays [17]. The authors tackle time synchronization, localization, and distributed cascaded beamforming in order to enhance audio perception (relevant information) in a self-organizing distributed system consisting of N mobile PC platforms each equipped with built-in microphone array for recording acoustic signals and the capacity to support 5.1/7.1 channel audio outputs. Here, the notion of relevant information is directly related to human perception, which makes it difficult to formally define a distortion measure. It must be noted that the information bottleneck method has not yet been applied to sensing strategies.

3.2. Quantization

The use of quantization is nearly always motivated by the need to reduce the amount of data needed to represent (fidelity-based criterion such as MSE) or process (distortion measure given the task at hand) a signal. Quantization refers to the process of approximating a continuous range of values (or a very large set of possible discrete values) by a relatively small set of discrete symbols or integer values. It may also be used to greatly reduce the complexity of the algorithms processing the quantized data (e.g., via simple table lookups).

Various researchers have shown that a task-driven quantization strategy usually leads to better decision-making performance than the ordinary MSE Lloyd quantizer design [18] for signal processing tasks including statistical classification, estimation and modeling. Generic approaches for incorporating such tasks into the quantization strategy by carefully choosing the distortion measure are surveyed in [19].

Task-driven quantization has also been successfully applied to specific applications. For example, the authors of [20, 21] recently proposed an algorithm to jointly quantize acoustic sensor readings and perform the task of source localization on those quantized observations. In another example [22], the authors tackle quantizer design for speaker verification systems based on adapted Gaussian mixture models. Their quantization strategy minimizes the squared loss in log-likelihood ratio, and was shown to trivialize to a conventional weighted MSE quantizer.

However a distortion measure is not always readily available, in which case the problem of task-driven quantization has to be addressed directly, by preserving the relevant information about another variable. The information bottleneck principle has been successfully applied to various practical quantization problems. For example, the *simple* agglomerative information bottleneck [23] method has recently been applied to phoneme recognition and speaker identification [11]. In particular, the authors first performed a standard vector quantization of the cepstral feature set; then applied the method on the clusters resulting from vector quantization such as to extract the relevant information for the task at hand (phoneme or speaker recognition).

3.3. Discussions

Most of the above work derive strategies for a single processing task. Distributed stream processing systems typically perform multiple processing operations either simultaneously on the same input data or sequentially on the data flowing through multiple processing stages (directed acyclic graph). Therefore one must extend existing theories and methods to accommodate for such complex environments. The theoretical foundation along with several quantization algorithms for the general simultaneous estimation and/or detection problem(s) have recently been addressed in [24, 25]. The authors also proposed and explored a compression approach to support sequential inference tasks, referred to as *task-embedded compression*. However, many problems remain open. For example, consider a cascade of detection tasks in which a task is triggered for processing only if the upstream task has detected an event (e.g., speech recognition for select subset of speakers). A possible approach is to extract the information relevant to a weighted function of the various targets (i.e., '*Jack of All Trades*' solution). However such an approach may increase the false alarm rate [1] of some detection tasks, resulting in an increased demand for computing resources of the downstream processing operators, thereby potentially diverting system resources away from processing tasks residing on the affected computing platforms. Another approach consists in providing each processing task with the most relevant information in order to maximize its classification accuracy. One potential solution to this problem has been addressed in [26] in the context of scalable speech recognition. The authors considered two sequential speech recognition systems with very different resource requirements. They combine scalable recognition with scalable compression in a distributed speech recognition application to reduce both the computational load and the bandwidth requirements at the stream processing system.

Also, relevant information is a function of the processing tasks instantiated within the stream processing system. Those processing tasks depend on the queries submitted to the system at any point in time. Clearly, user queries come and go. One possible strategy to track relevant information is by resorting to a feedback loop. Feedback comes at a cost (e.g., in terms of architectural complexity and stability of the overall system). To the best of our knowledge, such problems have not yet been addressed.

Finally, distributed stream processing systems introduce additional problems in terms of data losses and resource constraints, which processing operators have to cope with. Some recent stream processing systems [5] can provide information to processing operators about the available resources like CPU, memory usage and I/O utilization, which could be used by the algorithms to achieve the best resource-to-accuracy tradeoffs.

[1] A false alarm occurs where a non-target event exceeds the detection criterion and is identified as a target.

4. CONCLUSIONS

Task-driven signal processing is an excitingly rich research field. The primary objective of this position paper was to highlight some of the existing research and unexplored challenges associated with task-driven signal processing in distributed stream processing systems. This paper merely skims the surface of wealth of literature in this area.

We first presented the notion of *relevant information* from two different information-theoretic approaches – the Rate Distortion (*RD*) theory and the Information Bottleneck (*IB*) method. Armed with this definition, we reviewed existing sensing and quantization strategies designed for some specific classification, detection and/or estimation tasks. Finally, we briefly described a few unexplored challenges stream processing systems bring, including some beyond the scope of (yet related to) signal processing.

5. ACKNOWLEDGEMENTS

The authors of the present paper are grateful to the authors of [19, 20, 24, 16, 17] for their contribution to the ICASSP 2006 Special Session on Signal Processing Challenges in Distributed Stream Processing Systems. The authors would also like to thank Deepak Turaga from the IBM T.J. Watson Research Center for useful comments and fruitful discussions.

6. REFERENCES

[1] T. Gotz and O. Suhre, "Design and implementation of the UIMA common analysis system," *IBM Systems Journal*, vol. 43, no. 3, 2004.

[2] D. Abadi, D. Carney, U. Cetintemel, M. Cherniack, C. Convey, S. Lee, M. Stonebraker, N. Tatbul, and S. Zdonik, "Aurora: A new model and architecture for data stream management," *VLDB Journal*, vol. 12, no. 2, pp. 120–139, August 2003.

[3] M. Cherniack, H. Balakrishnan, M. Balazinska, D. Carney, U. Cetintemel, Y. Xing, and S. Zdonik, "Scalable distributed stream processing," in *Proceedings of the CIDR*, 2003.

[4] S. Chandrasekaran, O. Cooper, A. Deshpande, M. J. Franklin, J. M. Hellerstein, W. Hong, S. Krishnamurthy, S. R. Madden, V. Raman, F. Reiss, and M. A. Shah, "Telegraphcq: Continuous dataflow processing for an uncertain world," in *Proceedings of the CIDR*, 2003.

[5] L. Amini, H Andrade, F. Eskesen, R. King, Y. Park, P. Selo, and C. Venkatramani, "The Stream Processing Core," Tech. Rep. RSC 23798, IBM T. J. Watson Research Center, Nov. 2005.

[6] N. Tishby, F. Pereira, and W. Bialek, "The information bottleneck method," in *Proceedings of the 37th annual Allerton Conference on Communication, Control, and Computing*, 1999.

[7] N. Slonim, *The Information Bottleneck: Theory and Applications*, Ph.D. thesis, The Hebrew University, 2002.

[8] Y. Dong and L. Carin, "Rate-distortion bound for joint compression and classification," in *Data Compression Conference (DCC)*, Snowbird, Utah, March 2003, IEEE Computer Society.

[9] N. Slonim and N. Tishby, "Document clustering using word clusters via the information bottleneck method," in *Proceedings of the 23rd annual international ACM SIGIR conference*, July 2000.

[10] N. Slonim, N. Friedman, and N. Tishby, "Unsupervised document classification using sequential information maximization," in *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Tampere, Finland, July 2002, pp. 129–136.

[11] R. M. Hecht and N. Tishby, "Extraction of relevant speech features using the information bottleneck method," in *Proceedings of InterSpeech*, June 2005.

[12] J. Goldberger, H. Greenspan, and S. Gordon, "Unsupervised image clustering using the information bottleneck method," in *DAGM*, September 2002.

[13] M. Tubaishat and S. Madria, "Sensor networks: an overview," *IEEE Potentials Magazine*, vol. 22, pp. 20–23, April-May 2003.

[14] F. Zhao and L. Guibas, *Wireless Sensor Networks: An Information Processing Approach*, Morgan Kaufmann Publishers, May 2004.

[15] D. Yang, J. Shin, A. Ercan, and L. Guibas, "Sensor tasking for occupancy reasoning in a camera network," in *IEEE/ICST 1st Workshop on Broadband Advanced Sensor Networks (BASENETS)*, 2004.

[16] S. A. Aldosari and J. M. F. Moura, "Topology of sensor networks in distributed detection," in *Proceedings of IEEE ICASSP*, May 2006.

[17] Y. Jia, Y. Luo, Y. Lin, and I. V. Kozintsev, "Distributed microphone arrays for digital home and office," in *Proceedings of IEEE ICASSP*, May 2006.

[18] S. P. Lloyd, "Least squares quantization in pcm," *Reprinted in IEEE Transactions on Information Theory*, vol. 28, pp. 127–135, March 1982.

[19] R. M. Gray, "Quantization in task-driven sensing and distributed processing," in *Proceedings of IEEE ICASSP*, May 2006.

[20] Y.-H. Kim and A. Ortega, "Maximum a posteriori (map)-based algorithm for distributed source localization using quantized acoustic sensor readings," in *Proceedings of IEEE ICASSP*, May 2006.

[21] Y. H. Kim and A. Ortega, "Quantizer design for source localization in sensor networks," in *Proceedings of IEEE ICASSP*, March 2005.

[22] I. H. Tseng, O. Verscheure, D. S. Turaga, and U. V. Chaudhari, "Quantization for adapted gmm-based speaker verification," in *Proceedings of IEEE ICASSP*, May 2006.

[23] N. Slonim and N. Tishby, "Agglomerative information bottleneck," in *Proceedings of the Neural Information Processing Systems (NIPS) Conference*, 1999.

[24] M. Chen and M. Fowler, "Data compression for simultaneous/sequential inference tasks in sensor networks," in *Proceedings of IEEE ICASSP*, May 2006.

[25] M. Chen, *Data Compression for Inference Tasks in Wireless Sensor Network*, Ph.D. thesis, State University of New-York at Binghamton, 2005.

[26] N. Srinivasamurthy, A. Ortega, and S. Narayanan, "Efficient scalable speech compression for scalable speech recognition," in *Proceedings of Eurospeech*, Aalborg, Denmark, September 2001.