Frames: Data-driven Windows to Improve Task-Based Performance

Windows segment data streams. By their nature, windows are fixed – defined either by a time interval or a number of tuples – the definition unvarying over the course of a stream. However, streams are highly variable, just like their natural counterparts. It seems intuitive that a fixed, unvarying segmentation does not fit the naturally variable nature of data streams. In this work, we propose frames, or data-dependent windows, which divide data streams into pieces based on the stream content.

One purpose of windows is to reduce stream volume for high-speed / high-volume streams. For example, stream systems may calculate aggregates over (possibly-tumbling) windows – these window aggregates provide a type of summary of the data stream, similar in some ways to what one might get from a sample of the data stream.

Another purpose of windows is to smooth out fluctuations in stream content. Data streams often come from sensors or other sources with known variance in the data readings. A (possibly-sliding) window may serve to smooth those variances and provide a smoothed data stream.

As stated above, windows typically segment streams based on time or number of tuples. Given a stream of traffic speed data from a sensor on a freeway, we might use a five-minute window to smooth out variations and to reduce the volume of speed readings to a rate at which they could be monitored by a human. The choice of a five-minute window is based on traffic engineers’ perception that three-to-five minute windows effectively smooth out the natural variations in traffic speed and traffic sensor inaccuracy. However, the choice of the 5-minute window is somewhat arbitrary and feels restrictive. The goal of the window is to smooth variations, reduce volume and give an estimate of the current traffic speed at the sensor location. A more dynamic windowing scheme might provide a better approximation. In this work, we propose frames, or data-dependent windows, which divide data streams into pieces based on stream content.

With regular windows, we are not including a way to include human judgment. Can choose a size of windows, but may not be a way that the human thinks of the interesting parts of the stream. Interesting parts may be where traffic speed is changing significantly – or periods of time where velocities are under (or over) a certain number. Two parts – standard windows do not reflect all the ways that a user may think are important or reasonable ways of partitioning or condensing the stream. But the second half of that is that a user might not have a good sense about what is needed to perform the particular task they have – how would you assess whether you are actually reducing noise levels without losing too much of the underlying signal. I’m not sure a person would know what is a good way of segmenting the stream to get a good scatter plot, or may not realize that any fixed windowing is going to give them a really bad plot (or even worse a mediocre part). Need to specify segmentation in a way that a human thinks in that domain, also a need to give some guidance as to whether a given framing and/or segmentation is doing the job. We are working on both of those things. We are working on more flexible ways to segment and ? the stream. Developing metrics to see how a given framing is working against a given task.

Traffic case – may want to smooth out noise and capture rapid changes.

Windows do not capture the need for varying resolutions based on data – i.e. higher resolution in ‘more interesting’ regions – delta windows give us higher resolution in regions of change, lower resolution in regions of no change.

In case of tow data, if also was collecting ph (dissolved oxygen). Interesting, but unlikely to influence dye - less likely to capture interesting parts of data set like flsp or depth is.

Windows were initially a response to implementation issues – I have blocking operators or I need to get rid of state. Motivation was to get something that wasn’t going to blow up without a lot of consideration about what windows are semantically meaningful, which ones are useful, and/or do a reasonable job on the data.

Windows do not vary from domain to domain.

Analog in stored data domain – is stuff on histograms, where initial attempts were very split into even intervals. Can answer certain queries on histograms.

Purpose of a scatter plot is to look at the relationship (such as correlation) between two variables. In the dye experiments, the scientists want to know at what depths of water is there dye. Due to the nature of the experiments, there are many near-zero values at all depths. The dye is released in the water at a specific physical location and spreads over time – a boat is driven through the water with a tow thing in a pattern to detect dye. Due to the nature of this experiment, we expect there to be many near-zero values at all depths. Near-zero/zero values occur when the boat is not in the current region of dye.