Do you know your IQ?
A research agenda for information quality in systems

Kimberly Keeton and Pankaj Mehra
HP Labs
firstname.lastname@hp.com

John Wilkes
Google, Inc.
johnwilkes@google.com

Abstract

Information quality (IQ) is a measure of how fit information is for a purpose. Sometimes called Quality of Information (QoI) by analogy with Quality of Service (QoS), it quantifies whether the correct information is being used to make a decision or take an action. Failure to understand whether information is of adequate quality can lead to bad decisions and catastrophic effects. The results can include system outages, increased costs, lost revenue – and worse. Quantifying information quality can help improve decision making, but the ultimate goal should be to select or construct information sources that have the appropriate balance between information quality and the cost of providing it. In this paper, we provide a brief introduction to the field, argue the case for applying information quality metrics in the systems domain, and propose a research agenda to explore this space.

Categories and Subject Descriptors

H.3.4 [**Information storage and retrieval**]: Systems and Software.

General Terms

Management, Measurement, Performance, Design, Reliability, Experimentation.

Keywords

Information quality, IQ, QoI, data quality, uncertainty, prediction, modeling, information processing pipeline, goal-directed design.

1. INTRODUCTION

Automated earthquake monitoring systems can trigger actions that are designed to mitigate damage if the event is real: closing pipelines, shutting down nuclear reactors, and evacuating schools [Grasso2005]. A false alarm can cost millions of dollars.

A special offer mailed from a pizza chain to the top 20% of its customers missed its revenue target by $0.5M because of bad customer data. An attempt to fix the problem purged 2% of the best customers from their database [Dravis2002].

In 1999, NATO bombed the Chinese embassy in Belgrade, killing three people, because a faulty strike planning process failed to catch the use of inaccurate positioning data [Wikipedia2008].

Half of the reports from a monitoring application on PlanetLab differed from the true state of affairs by more than 30% [Jain2008].

In trying to achieve a guaranteed quality of service for a transaction-processing application, blindly turning on full performance monitoring doubled the CPU load, preventing the performance target from being met [Agarwala2006].

As these examples show, knowing whether information is good enough matters, and when the information is not good enough, bad results occur – or are likely to. The systems community commonly discusses quality of service, but largely ignores information quality. We believe this must change: the goal of this paper is to introduce the field of information quality to the systems community, and suggest ways it can be measured, used, and – finally – designed for.

IQ assesses fitness for use – whether information is good enough for the purpose to which it is put, such as making a decision. The desire to obtain fresh, accurate, complete information has driven a multi-billion dollar business in Enterprise Data Warehousing, which pays for itself by improving the quality of business decisions that can be made. At the same time, the Web has taught us that “good enough” information is often immensely valuable, and that perfection is not necessary for usefulness. Which is the right standard to aim for? It depends on how the information will be used.

Not all data is created equal, and not all attributes of information quality are equally valuable. Just as with QoS, providing high information quality is often costly, and may be unnecessary. And also, just as with QoS, not having sufficient information quality can be costly. Making this tradeoff correctly is a recurring theme in what follows.

1.1 Related work overview

Not surprisingly, most work on information (or data) quality has taken place in the database, decision analysis and business domains. For instance, Trio [Widom2009] and BayesStore [Wang2008b] are extended databases built to support data uncertainty as first-class entities. [Aggarwal2009] and [Dalvi2007] offer surveys of database-related approaches to the use of uncertain data.

Considerable work has been done on decision making in the face of uncertainty, because uncertainty is commonplace in the information sources used in science, economics, and business. See [Kahneman1982] for representative samples.

In the business domain, much work is concerned with models for IQ assessment and processes to increase the IQ of stored data. There is a heavy emphasis on systems and processes that involve people, such as change management and processes for qualifying data as it is captured.

The provenance (or lineage) of a piece of data or information describes the process that produced that piece, including the original data sources and the processing steps used along the way [Beth2005]. Data provenance can be used to determine the data’s IQ, and to build trust or believability in the data, but it is not per se a measure of information quality [Rajbhandari2008]. Indeed, there are times when the provenance of an IQ assessment is itself important. The provenance community is largely concerned with processes and tools for gathering, organizing, and querying the data that will allow deductions about pieces of information to be made. We believe that provenance and information quality complement one another, because information quality is just one of the deductions enabled by provenance, and provenance data is just one input to information quality.

Other communities have also recognized the value of explicitly tracking information quality. Members of the eScience community share experimental datasets, and must explicitly describe their contents and quality, to match information producers and consumers [Preece2008]. Additionally, the visualization community is beginning to provide support for visualizing the quality of large datasets [Wang2008a].

On the other hand, there is little recognition of the value of information quality in the systems domain: observe the lamentable lack of statistical properties for measurements such as repeatability, standard deviation, confidence limits, and significance in systems papers. “Everybody knows” that information quality is important, but few of us do much about it!

There has been some recent progress: a recent OSDI paper discussed the value of measuring information quality for a network-monitoring system [Jain2008]. Bartlet-Ros, et al., describe a network monitoring system that sheds excess load under extreme traffic conditions, while maintaining acceptable traffic query accuracy [Bartlet-Ros2007]. Murty and Welsh advocate using the IQ (e.g., harvest and freshness) of information sources to drive the development of fault tolerance mechanisms in Internet-scale sensing environments [Murty2006]. In the area of modeling IQ, Cohen, et al., describe a framework for calculating confidence intervals for arbitrary combinations of aggregation operations with sampling operations [Cohen2008]. But those studies are just the first steps.

1.2 Paper outline

The main contributions of this paper are to present a framework on which to hang systems research in IQ; to explain a few of the noteworthy research problems; and (hopefully) to encourage others to work in this space.

We believe that making IQ a first-class property like QoS will benefit the users of the systems we construct, and open up a range of interesting research. The remainder of this paper discusses three parts of a research agenda for Information Quality:

- Metrics for measuring information quality.
- Predicting the effects of analyses such as aggregation, averaging, “data cleansing”, and correlations between multiple sources, on IQ.
- Automatically constructing an information processing flow that meets the needs of a decision-making process.

2. MOTIVATING EXAMPLES

In this section we present two examples to illustrate the role of information quality.

2.1 System monitoring

Imagine a large internet service provider that runs many user-facing applications in several data centers across tens of thousands of machines. Each service provides instrumentation points, many of which are capable of generating voluminous data – so much so that it is not cost-effective to enable them all, all the time.

Calculating and using IQ is made harder by the scale, asynchrony, and partial failures induced by the distributed nature of the target. These issues apply to the monitoring system, as well.

People monitor the system to look for opportunities to tune it; to decide where to bring up new services; to see if it is meeting its customers’ needs; and – when things go wrong – to determine the problem’s cause, so it can be fixed. Each of these purposes can manage with a different level of information quality: long-term trend analysis doesn’t typically need the most up-to-date data, but diagnosing a problem is often best done with the most recent status information that is available – even if it is too expensive to gather in the normal state.

2.2 Information service

Consider a large enterprise seeking to achieve a single-view-of-customer (SVC) information system and, from this integrated view, drive various on-line analytics and decision support workloads. To do this, information needs to be gathered from hundreds of operational and organizational systems, each of which may have its own processes and standards.

A number of IQ criteria arise quite naturally in this world. Freshness is important: new information from the sources must be reflected in the integrated SVC as quickly as possible, suggesting perhaps that trickle updates or other incremental updating schemes should be used. But the data cleansing queries used in information integration operate best when applied to multiple updates simultaneously, in order to avoid duplicate or inconsistent information, which suggests batch updates. IQ-driven optimization can help people decide how to make the tradeoff between freshness and consistency by providing quantitative data about the alternatives.

Tracking IQ for data sources through the system can provide users with information about whether the query results they see are to be believed. Many questions arise, including how one should obtain an appropriate level of IQ for an important decision. This topic is the subject of the framework we present below.

3. MEASURING YOUR IQ

Information quality is an assessment of whether information is suited for the purposes to which it is put. IQ metrics provide quantitative data to make this assessment.

Standalone IQ metrics are independent of the use the information is put to, and can be directly measured by the information producer. They include: how recent is the data? how complete is it? how accurate is it? how representative is it (if sampled)?

For example, in a distributed monitoring or sensor system, producers can evaluate the quality of the analyses performed by measuring coverage (e.g., fraction of nodes represented in an aggregate); the variance of the aggregates; and the freshness of the aggregates, due to the aggregation interval.

Context-dependent IQ metrics can only be calculated relative to the context and needs of the information consumer. They cannot be evaluated by looking solely at a single information source.

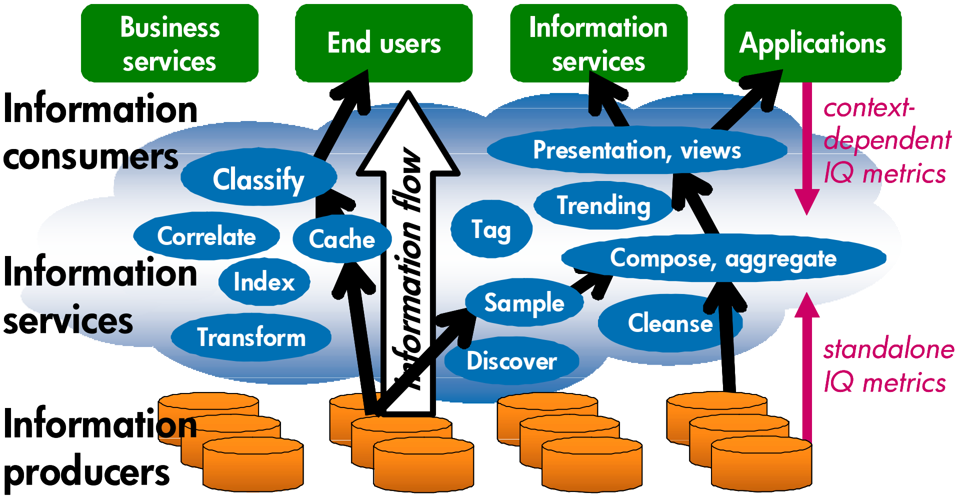

Figure 1: an information flow view

For example, a consumer that is trying to diagnose problems will evaluate IQ using metrics like the false positive and negative rates, and time delays for analyses used for detection and diagnosis. In the area of search, end users want to understand the relevance of their search results – typically measured using precision and recall. Precision is a measure of the accuracy of the results (fraction of results that are correctly identified), and recall is a measure of the completeness of the results (fraction of true matches that are identified).

Some context-dependent IQ metrics are difficult to quantify, such as whether information is actionable, trustworthy, or privacy-preserving (even when combined with other data). These are often the most useful metrics, and will benefit from further investigation into how they can be provided.

Composite IQ metrics are measures taken across multiple sources. They can be context dependent or independent. For example: is this data source unique, or is there a duplicate copy obtainable elsewhere? Do these two sources agree (e.g., the strength of correlations or duplicate coverage between them)? Do we know the information's provenance? Is it auditable? Which source should be trusted more for the desired purpose?

3.1 Research challenges

Our basic observation is that unless systems explicitly track their information quality, consumers of the information they provide cannot make judgments and decisions with high confidence. Information providers don't have to provide perfect IQ, but they need to be explicit about what IQ they do provide. Thus, a first research challenge is in *providing lightweight, scalable mechanisms for determining IQ metrics*.

In addition, consumers need to determine which IQ metrics (and what values) are appropriate for their purposes (e.g., decision-making, taking action) and resist the urge to use ill-suited metrics just because they are easy to measure. A resulting research challenge is *mapping between meaningful consumer-oriented IQ metrics and easy-to-measure producer-oriented IQ metrics*.

For example, provisioning decisions for peak usage might rely on a monitoring system that drops measurements under heavy load; not knowing this is likely to lead to end-user dissatisfaction. Availability metrics that have poor coverage are likely to omit precisely the systems experiencing the most difficulties, leading to inappropriate system-management responses.

It is often more straightforward to measure (or deduce) IQ than to predict it *a priori*. This approach has the advantage of adapting to changes in the underlying source's behavior. But which metrics should be generated? By analogy with performance monitoring for diagnostics [Cohen2004], machine learning techniques could potentially allow the choice of IQ metrics to be determined dynamically, with the goal of reducing the amount of duplicate IQ information reported or maximizing its predictive value.

IQ-driven tools that build models of data sources can produce a much higher fidelity description by automatically dividing the description into different time periods [Kiernan2009]. And to address the question of what IQ is "good enough", consumers might combine machine learning and information retrieval techniques to calculate IQ signatures, keeping track of acceptable and unacceptable values, so that they can easily be identified when observed in the future, as in [Cohen2005].

4. PREDICTING YOUR IQ

It's not enough to measure information quality at a data source, if that data will be transformed before it is used – e.g., by averaging, sampling, aggregation, cleansing, merging, indexing, caching, correlating against other sources, and so on. It's also necessary to understand the IQ of the transformed data.

A good way to think about this problem is to consider the IQ of different stages of an information processing pipeline, or directed graph (DAG). See Figure 1 for a small sample of the kinds of components or building blocks that might be found in such a graph. These building blocks can be implemented in many ways, including modules within a single (potentially distributed) application or as services in a service-oriented architecture.

Each processing step transforms one or more inputs into a new data source, with a new set of IQ metrics. For example, averaging elements in a time series across non-overlapping time windows may increase predictive quality, but lower freshness; smoothing a noisy source can improve usability at the expense of eliminating potentially significant outliers; and correlations can improve

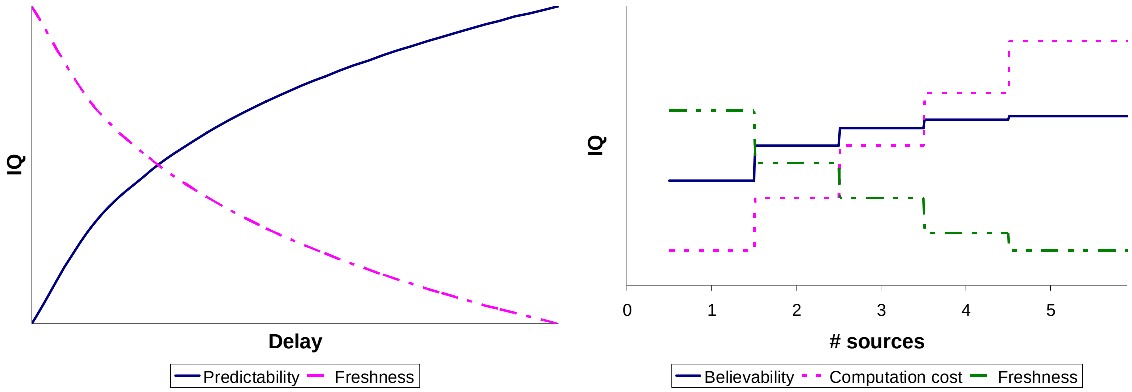

Figure 2: some typical IQ tradeoffs

believability at the expense of filtering out potentially useful material. Different algorithms or parameter settings may have different costs (e.g., resources used) and produce different results (e.g., over different averaging intervals or different fractions of nodes contained in a spatial aggregate).

For example, distributed diagnostic tools monitor the performance of applications and infrastructure devices by collecting a variety of time series observations, including low-level CPU, disk and network performance metrics; energy consumed; system logs and application logs. Unfortunately, collecting large quantities of data is expensive, so there is pressure on administrators to gather as little data as possible, or to subset it as quickly as possible. To limit collection costs, administrators carefully configure various parameters, such as the frequency of gathering and reporting metrics, the choice of which nodes are instrumented, or the rate of sampling employed (e.g., 5 minutes out of every hour or roughly every Nth request). Experience has shown that system monitoring data is often noisy, so administrators often apply data scrubbing to remove missing, duplicate and out-of-bounds observations [Arlitt2005]. Once this has been done, it is possible to do trend analyses, aggregate multiple data sources together (e.g., all machines in a rack or site), and correlate information across multiple sources (e.g., low-level infrastructure observations and application-level logs) to classify anomalous behavior [Cohen2004]. The cleaning, aggregation, and analyses performed on those data streams often dictate how successful the diagnostic tools are going to be. Knowing just how IQ will vary as these analyses are applied allows administrators to maximize diagnostic abilities while minimizing data collection costs.

4.1 Research challenges

We need to be able to predict the effects of data analysis on IQ if we are going to understand how to use the transformed data. Doing so requires the ability to *model the IQ effects of each of the components in a processing DAG*. Additionally, because we are trying to predict the effect of a complete processing pipeline, we need the ability to *compose these IQ models.*

The modeling and measurement community provides techniques that have been used to address some related challenges – it remains to be seen whether they can provide the breadth of coverage that's needed for a general IQ solution. For instance, active probing and fitness models (e.g., [Mesnier2007]) may prove useful for measuring the IQ of a single DAG component. Work in the systems community on end-to-end tracing of requests in distributed environments may provide insights into methods for effectively tracking end-to-end IQ. If the system components are well understood (e.g., because access to source code is available), then white box techniques (e.g., [Barham2004, Thereska2006]) may be effective for directly tracking IQ. However, if components must be treated as a black box, then IQ behavior must be observed and/or inferred, as in [Aguilera2003].

5. GETTING THE IQ YOU WANT

Our ultimate goal is to provide end users with the information quality that they need. This will require choosing information source(s) that provide it directly, or constructing a DAG to generate it if such sources aren't readily available – or both.

It may be necessary to use multiple sources to increase confidence in the result, or it may simply be possible to select a suitably-trustworthy source. It may be necessary to request different amounts of data: [Agarwala2006] describes a system where the amount of monitoring data being gathered can be increased or decreased, allowing a tradeoff between completeness, freshness, coverage, and collection cost. It may be possible to combine new data with old, for trend analysis and anomaly detection.

For example, a distributed monitoring and control system might contain two disjoint information processing pipelines, which can be combined in different ways to achieve different goals. The monitoring-centric pipeline collects frequent observations, which allows it to identify outliers that may indicate problems. However, because it generates such a high volume of data, it does not retain observations for a long time. A control-centric pipeline collects observations that are aggregated over longer time intervals, and retains them for longer periods of time, to permit trending analysis. These pipelines can be combined in different ways to achieve different goals. If the control system detects a problem, it could use observations from the more intensive monitoring system to permit a more detailed diagnosis of a problem. Similarly, if the monitoring system experiences false positive rates that are too high, it could leverage the smoothing provided by the control system's information processing pipeline to increase its confidence in reporting a problem.

5.1 Research challenges

We believe the third set of research challenges is in *automating the design of DAGs that deliver a target IQ*. Designing a DAG requires working backwards from a target IQ and the IQ metrics of the available sources to (1) pick the topology of the DAG, (2) select the components to use, (3) their sequence, and (4) their configurations, while minimizing costs such as collection, processing, and storage overheads, and conforming to security, privacy, and auditability requirements.

Today, processing pipelines are typically constructed using rules of thumb (e.g., "scrub data before aggregating it"), or a semi-exhaustive search ("let's try *this* combination first"). To automate this process, we must find ways to:

1. discover information sources that provide the necessary IQ, which may include characterizing new sources,
2. explore alternative processing pipelines/DAGs, using the predictive models described in Section 4,
3. select one that produces the desired information quality while satisfying other constraints, and
4. deploy the resulting design

The two key components of this approach are the ability to *model a tentative solution*, and the ability to *explore the design space efficiently*. Both are significant research challenges.

6. SUMMARY

Understanding the quality of the information used to make decisions matters; without it, inappropriate decisions can all too easily be made on poor data, with a range of adverse consequences.

In this paper, we have presented a model for how to think about information quality in the systems context; identified some common IQ metrics; highlighted the importance of predicting and modeling the IQ that an information-processing system or service stack will produce; and suggested a challenging end-goal of automatically constructing information pipelines to meet given IQ goals. We believe that the benefits are real, and the research problems are both challenging and tractable. We hope other researchers will join us in exploring this field.

7. REFERENCES

[Agarwala2006] S. Agarwala, Y. Chen, D. Milojicic, and K. Schwan, "QMON: QoS- and utility-aware monitoring in enterprise systems", *3rd IEEE International Conference on Autonomic Computing (ICAC)*, 2006.

[Aggarwal2009] C. Aggarwal and P. Yu, "A survey of uncertain data algorithms and applications," *IEEE Trans. on Knowledge and Data Engineering*, Vol. 21, No. 5, May 2009, pp. 609-623.

[Aguilera2003] M. Aguilera, J. Mogul, J. Wiener, P. Reynolds, and A. Muthitacharoen, "Performance debugging for distributed systems of black boxes," *Proc. SOSP*, 2003, pp. 74-89.

[Arlitt2005] M. Arlitt, K. Farkas, S. Iyer, S. P. Kumaresan, S. Rafaeli, "Data assurance: a prerequisite for IT automation", HPL-TR-2005-212, November 2005.

[Barham2004] P. Barham, A. Donnelly, R. Isaacs, and R. Mortier, "Using Magpie for request extraction and workload modelling," *Proc. OSDI,* 2004, pp. 259-272.

[Bartlet-Ros2007] P. Bartlet-Ros, G. Iannaccone, J. Sanjuas-Cuxart, D. Amores-Lopez and J. Sole-Pareta, "Load shedding in network monitoring applications," *Proc. USENIX Annual Technical Conf.*, 2007, pp. 59-72.

[Beth2005] Y. Beth, B. Plale and D. Gannon,"A survey of data provenance in e-Science," *SIGMOD Record*, Vol. 34, 2005, pp. 31-36.

[Cohen2004] I. Cohen, M. Goldszmidt, T. Kelly, J. Symons, and J. Chase, "Correlating instrumentation data to system states: a building block for automated diagnosis and control," *Proc. OSDI*, 2004, pp. 231-244.

[Cohen2005] I. Cohen,S. Zhang, M. Goldszmidt, J. Symons, T. Kelly, and A. Fox,"Capturing, indexing, clustering, and retrieving system history, *Proc. SOSP*, 2005, pp. 105-118.

[Cohen2008] E. Cohen, N. Duffield, C. Lund, M. Thorup, "Confident estimation for multistage measurement sampling and aggregation,", *Proc. SIGMETRICS*, 2008, pp. 109-120.

[Dalvi2007] N. Dalvi and D. Suciu, "Management of probabilistic data: foundations and challenges," *Proc. PODS*, 2007, pp. 1-12.

[Dravis2002] Frank Dravos. "Information quality: the quest for justification", *Business Intelligence Journal* 7(2), Spring 2002.

[Grasso2005] V.F. Grasso, J.L. Beck,. and G. Manfredi, "Seismic early warning systems: procedure for automated decision making," Technical report EERL-2005-02, Caltech, Pasadena, CA, November 2005.

[Jain2008] N. Jain, P. Mahajan, D. Kit, P. Yalagandula, M. Dahlin, and Y. Zhang, "Network imprecision: a new consistency metric for scalable monitoring," *Proc. OSDI'08*, December 2008.

[Kahneman1982] D. Kahneman, P. Slovic and A. Tversky, *Judgment under Uncertainty : Heuristics and Biases*, Cambridge University Press, April 1982.

[Kiernan2009] J. Kiernan and E. Terze, "EventSummarizer: a tool for summarizing large event sequences," *Proc. 12th Intl. Conf. on Extending Database Technnology (EDBT'09)*, March 2009.

[Mesnier2007] M. Mesnier, M. Wachs, R. Sambasivan, A. Zheng, and G. Ganger, "Modeling the relative fitness of storage," *Proc. SIGMETRICS*, 2007, pp. 37-48.

[Murty2006] R. Murty and M. Welsh, "Towards a dependable architecture for Internet-scale sensing," *Proc. 2nd Workshop on Hot Topics in Dependability (HotDep '06)*, November 2006.

[Preece2008] A. Preece, P. Missier, S. Embury, B. Jin and M. Greenwood, "An ontology-based approach to handling information quality in e-Science", *Concurrency and Computation: Practice and Experience* 20:253–264, 2008.

[Rajbhandari2008] S. Rajbhandari, O. Rana and I. Wootten, "A fuzzy model for calculating workflow trust using provenance data," *Proc. of 15th ACM Mardi Gras Conf.*, 2008, pp. 1-8.

[Thereska2006] E. Thereska, B. Salmon, J. Strunk, M. Wachs, M. Abd-El-Malik, J. Lopez and G. Ganger, "Stardust: tracking activity in a distributed storage system," *Proc. SIGMETRICS*, June 2006, pp. 3-14.

[Wang2008a] C. Wang, K.-L. Ma, "A statistical approach to volume data quality assessment," *IEEE Trans.s on Visualization and Computer Graphics*, Vol. 14, No. 3, May/June 2008, pp. 590-602.

[Wang2008b] D. Wang, E. Michelakis, M. Garofalakis, and J. Hellerstein, "BayesStore: Managing Large, Uncertain Data Repositories with Probabilistic Graphical Models," *Proc. VLDB*, 2008, pp. 340-351.

[Widom2009] J. Widom, "Trio: a system for data, uncertainty, and lineage," In C. Aggarwal, editor, *Managing and Mining Uncertain Data*, Springer, 2009, pp. 113-148.

[Wikipedia2008] "NATO bombing of the Chinese embassy in Belgrade", Wikipedia, Dec. 2008.