Big Data or Right Data?

Ricardo Baeza-Yates

Yahoo! Labs Barcelona &
Web Research Group, DTIC, Univ. Pompeu Fabra
Barcelona, Spain
rbaeza@acm.org

Abstract. Big data nowadays is a fashionable topic, independently of what people mean when they use this term. The challenges include how to capture, transfer, store, clean, analyze, filter, search, share, and visualize such data. But being big is just a matter of volume, although there is no clear agreement in the size threshold where *big* starts. Indeed, it is easy to capture large amounts of data using a brute force approach. So the real goal should not be big data but to ask ourselves, for a given problem, what is the right data and how much of it is needed.[1] For some problems this would imply big data, but for the majority of the problems much less data is necessary. In this position paper we explore the trade-offs involved and the main problems that come with big data: scalability, redundancy, bias, noise, spam, and privacy.

Keywords: Scalability, redundancy, bias, sparsity, noise, spam, privacy.

1 Introduction

According to Wikipedia, "*big data* is a collection of data sets so large and complex that it becomes difficult to process using on-hand database management tools or traditional data processing applications". However, what really means on-hand database management tools or traditional data processing applications? Are we talking about terabytes or petabytes? In fact, a definition of volume threshold based on current storing and processing capacities might be more reasonable. This definition then may depend on the device. For example, *big* in the mobile world will be smaller than *big* in the desktop world.

Big data can be used in many applications. In the context of the Web, it can be used to do Web search, to extract information, and many other data mining problems. Clearly for Web search, big data is needed as we need to search over the whole content of the Web. Hence, in the sequel, we will focus on data mining problems, using the Web as main example. This is now called *wisdom of crowds* when the data comes from people [18]. The crucial difference between Web search and Web data mining, is that in the first case we know what we are looking for, while in the second we try to find something unusual that will be the answer to a (yet) unknown question.

[1] We use data as singular, although using it as a plural is more formal.

Recently we see a lot of data mining for the sake of it. This has been triggered by the availability of big data. Sometimes is valid to ask ourselves what is really interesting in a new data set. However, when people hammer a data set over and over again just because it is available, newer results become usually less meaningful. In some cases the results also belong to other disciplines (e.g. social insights) and there is no contribution to computer science (CS), but still people try to publish it in CS venues.

Good data mining is usually problem driven. For this we need to answer questions such as: What data we need? How much data we need? How the data can be collected? Today collecting data might be cheap and hence big data can be just an artifact of this step. After we have the data we need to worry about transferring and storing the data. In fact, transferring one petabyte even over a fast Internet link (say 100 Mbps) needs more than two years, too much time for most applications. On the other hand, today several companies store hundreds of petabytes and process dozens of terabytes per day.

When we have the data in place, another set of questions appears: Is all the data unique or we need to filter duplicates? Is the data trustworthy or there is spam? How much noise there is? Is the data distribution valid or there is a hidden bias that needs to be corrected? Which privacy issues must be taken care of? Do we need to anonymize the data?

After answering these questions we focus on the specific mining task: Can we process all our data? How well our algorithm scale? The ultimate question will be related to the results and the usefulness of them. This last step is clearly application dependent.

Another subtle issue is that most of the time when we need to use big data, the problem is to find the *right data* inside our large data. Many times this golden subset is hard to determine, as we need to discard huge amounts of data, where we have to deal again with bias, noise or spam. Hence, another relevant question is: How we process and filter our data to obtain the right data?

Hence, handling large amounts of data poses several challenges related to the questions and issues above. The first obvious one is scalability, our last step. Privacy is also very relevant as it deals with legal and ethical restrictions. Other challenges come with the data content and its intrinsic quality, such as redundancy, bias, sparsity, noise, or spam. We briefly cover all these issues in the following sections.

There are many other aspects of big data that we do not cover, such as the complexity and heterogeneity of data, as they are outside the scope of this paper.

2 Scalability

We can always collect and use more data thinking that more data will give improved results. In many cases that is true, but then transferring, storing, and processing larger amounts of data may not be feasible, as we challenge the bandwidth of the communication channels, the disk space of the computer infrastructure, and the performance of the algorithms used. As Internet bandwidth and storage has become cheaper, scaling the communication and hardware does not imply a proportional increase in cost. More data can also imply more noise as we discuss later.

On the other hand, the algorithms used to analyze the data may not scale. If the algorithm is linear, doubling the data, without modifying the system architecture, implies doubling the time. This might still be feasible, but for super linear algorithms most surely it will not. In this case, typical solutions are to parallelize and/or distribute the processing. As all big data solutions already run on distributed platforms, increasing the amount of data requires increasing the number of machines, which is clearly costly and proportional to the increase needed.

How else we can deal with more data? Well, another solution consists of developing faster (sometimes approximate) algorithms, at the cost of probably decreasing the quality of the solution. This becomes a clear option when the loss in quality is inferior to the improvements obtained with more data. That is, the time performance improvements should be larger than the loss in the solution quality. This opens a new interesting trade-off challenge in algorithm design and analysis for data mining problems.

One interesting example of the trade-off mentioned above is in lexical tagging. That is, recognize all named entities in a text. The best algorithms are super-linear but in [6], Ciaramita and Altun, present the first of a family of linear time taggers that have high quality (comparable to the state of the art).[2] To understand the trade-off we can do a back-of-the-envelope analysis. Let us assume that we can achieve a higher quality result with an algorithm that runs in time $O(n \log n)$ where n is the size of the text. Let us define Δq as the extra quality and as Q the quality of the linear algorithm. Without doubts, the number of correctly detected entities per unit of time should be larger for the linear algorithm. In fact, if we consider the case when both algorithms use the same running time, we can show that for enough text, that is $n = O(\beta^{\Delta q/Q})$, where $\beta > 1$ is a constant, the number of correct entities found by the faster algorithm will be larger. For some cases this will imply big data, but for many other cases it will not (for example, if the better quality algorithm has quadratic time performance).

Another important aspect of scalability is the processing paradigm that we can use to speed-up our algorithms. This is application dependent, as the degree of parallelization depends on the problem being solved. For example, not all problems are suitable for the popular map-reduce paradigm [8]. Hence, more research is needed to devise more powerful paradigms, in particular for the analysis of large graphs. In some cases we need to consider the dynamic aspect of big data, as in this case we may need to do online data processing that makes scalability even more difficult. Map-reduce is also not suitable for this case and one on-going initiative for scalable stream data processing is SAMOA [5].

[2] http://sourceforge.net/projects/supersensetag/.

3 Redundancy and Bias

Data can be redundant. Worse, usually it is. For example, in any sensor network that tracks mobile objects, there will be redundant data for all sensors that are nearby. In the case of the Web this is even worse, as we have lexical redundancy (plagiarism) estimated in 25% [2, 16] and semantic redundancy (e.g. same meaning written with different words or in different languages), which makes up an even a larger percentage of the content.

In many cases when we use data samples, the sample can have some specific bias. Sometimes this bias can be very hard to notice or to correct. One of the most well-known examples is click data in search engines, where the data is biased by the ranking and the user interface (see for example [10, 7]). In [2] we show evidence that Web publishers actually perform queries in order to find some content and republish. Thus, the conclusion is that part of the Web content is biased by the ranking function of search engines. Hence, this affects the quality of search engines.

Another interesting example of algorithm bias is tag recommendation. Imagine that we can recommend tags to new objects contributed by people (e.g. images). If we do so, in the long run, the recommendation algorithm will generate most tags, not the people. Hence, the resulting tag space is not really a folksonomy; it is a combination of a folksonomy and a machine-sonomy. This is a problem not only because we do not have a folksonomy anymore, but also because the algorithm will not be able to learn if there is not enough new data coming from users.

4 Sparsity, Noise and Spam

Many measures in the Web and other types of data follow a power law; so mining big data works very well for the head of the power law distribution without needing much data. This stops being true when the long tail is considered, because the data is sparser. Serving long tail needs is critical to all users, as all users have their shares of head and long tails needs as demonstrated in [12]. Yet, it often happens that not enough data covering the long tail is available when aggregated at the user level. Also, there will always be cases where the main part of the data distribution will bury the tail (for example, a secondary meaning of a query in Web search). We explored the sparsity trade-offs regarding personalization and privacy in [3].

We can always try to improve results by adding more data, if available. Doing so, however, might not always be beneficial. For example, if the added data increases the noise level, results get worse. We could also reach a saturation point without seeing any improvements, so in this case more data is worthless.

Worsen results can also be due to Web spam. That is, actions done by users in the form of content, links or usage, that are targeted to manipulate some measurement in the Web. The main example nowadays is Web spam to improve the ranking of a given website in a Web search engine [1] and there are a myriad of techniques to deal with it [17]. However this manipulation can happen at all levels, from hotel ratings to even Google Scholar citation counts [9]. Filtering spam is a non trivial problem and is one of the possible bias sources of any data[3].

5 Privacy

Currently, most institutions that deal with personal data guarantee that this data is not shared with third parties. They also employ as much secure communication and storage as possible to promise their clients or users that personal information cannot be stolen away. In some cases, such as in Web search engines, they have devised data retention policies to reassure regulators, the media and, naturally, their users, that they comply with all legal regulations. For example, they anonymize usage logs after six months and they de-identify them after 18 months (that is, for example queries cannot be mapped to anonymized users). One of the problematic twists of data, even more for big data, is that in many cases a specific user would prefer to forget past facts, especially in the context of the Web. This privacy concern keeps rising, especially with the advent of social networks.

Companies that use any kind of data are accountable to regulators such as the Federal Trade Commission (FTC) in the United States or should comply with the Data Protection Directive legislated in 1995 by the European Union Parliament. Indeed, the FTC has defined several frameworks for protecting consumer privacy, especially online [11]. Recently, the FTC commissioner even threatened to go to congress if privacy policies do not "address the collection of data itself, not just how the data is used" [15]. For similar reasons, the European Union is currently working on a new data protection directive that will supersede the old one.

Numerous research efforts have been dedicated to data anonymization. A favored one in large data sets is *k-anonymity*, introduced by Sweeney [19], which proposes to suppress or generalize attributes until each entry in the repository is identical to at least $k - 1$ other entries. To motivate this concept Sweeney shows that a few attributes are sufficient to identify many characteristics of most people. For example, a triple such as {ZIP code, date of birth, gender} allows to identify 87% of citizens in the USA by using publicly available databases. Today, in most problems where we need to derive insights from big data, k-anonymity is a *de facto* standard protection technique.

However, sometimes anonymizing data is not enough. One of the main examples appears in the context of Web search engines. In this case, users are concerned that their queries expose certain facets of their life, interests, personality, etc. that they might not want to share. This includes sexual preferences, health issues or even some seemingly minor details such as hobbies or taste in movies that they might not be comfortable sharing with everybody. Queries and

[3] We distinguish between noise that comes from the data itself, e.g., due to the measurement mechanism, from spam, which can be considered as artificial noise added by humans.

clicks on specific pages indeed provide so much information that the entire business of computational advertising rely on these. Query logs and click data reveal so much about users that most search engines stopped releasing logs to the research community due to the infamous AOL incident. Indeed, a privacy breach in query logs was exposed by two New York Times journalists who managed to identify one specific user in the anonymized log [4]. They spotted several queries, originating from a same user, referring to the same last name or specific locations. Eventually, they could link these queries to a senior woman who confirmed having issued not only these queries, but also more private ones. While not all users could necessarily be as easily identified, it revealed what many researchers had realized a while back, namely that simply replacing a user name by a number is not a guarantee of anonymity. Moreover, this incident exposed how difficult was the problem, as it is very hard to guarantee any privacy disclosure when you can cross your private data with a large number of public data sets. Subsequent research has shown that several attributes, such as gender or age, can be predicted approximately quite well [13].

6 Epilogue

Today *big data* is certainly a trendy keyword. For this reason we have explored many fundamental questions that we need to address when handling large volumes of data. For the same reason this year the first Big Data conferences are being held, in particular the first IEEE Big Data conference[4]. What is not clear is the real impact that these conferences will have, and which researchers will be attracted to them. As [14] states, could be a matter of size, efficiency, community, or logistics. Time will tell.

Acknowledgements

We thank Paolo Boldi for his helpful comments. This research was partially supported by MICINN (Spain) through Grant TIN2009-14560-C03-01.

References

1. R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval: The Concepts and Technology behind Search.* Second Edition. Addison-Wesley, 2011.
2. R. Baeza-Yates, Á. Pereira Jr., and N. Ziviani. Genealogical trees on the Web: a search engine user perspective. In *WWW 2008*, Beijing, China, Apr 2008, 367-376.
3. R. Baeza-Yates and Y. Maarek. Usage Data in Web Search: Benefits and Limitations. In *Scientific and Statistical Database Management: 24th SSDBM*, A. Ailamaki and S. Bowers (eds). LNCS 7338, Springer, Chania, Crete, 495–506, June 2012.
4. M. Barbaro and T. Z. Jr. A face is exposed for AOL searcher no. 4417749. *The New York Times*, Aug 9 2006.

[4] www.ischool.drexel.edu/bigdata/bigdata2013.

5. A. Bifet. SAMOA: Scalable Advanced Massive Online Analysis. `http://samoa-project.net/`, 2013.
6. M. Ciaramita and Y. Altun. Broad-Coverage Sense Disambiguation and Information Extraction with a Supersense Sequence Tagger In *Proceedings of Empirical Methods in Natural Language Processing (EMNLP)*, 2006.
7. O. Chapelle and Y. Zhang. A dynamic bayesian network click model for web search ranking In *Proceedings of the 18th international conference on World wide web (WWW'09).* pp. 1–10, 2009.
8. J. Dean and S, Ghemawat. MapReduce: Simplified Data Processing on Large Clusters. In *Proceedings of 6th Symposium on Operating Systems Design and Implementation (OSDI'04).* pp. 137-149, 2004.
9. E. Delgado López-Cózar, N. Robinson-García, and D. Torres-Salinas. Manipulating Google Scholar Citations and Google Scholar Metrics: simple, easy and tempting. Arxiv: `http://arxiv.org/abs/1212.0638`, 2012.
10. G. Dupret and B. Piwowarski. A user browsing model to predict search engine click data from past observations. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval.* pp. 331–338, 2008.
11. Federal Trade Commission. Protecting Consumer Privacy in an Era of Rapid Change. A Proposed Framework for Business and Policymakers. Preliminary FTC Staff Report, December 2012 (`http://www.ftc.gov/os/2010/12/101201privacyreport.pdf`)..
12. S. Goel, A. Broder, E. Gabrilovich, and B. Pang. Anatomy of the long tail: ordinary people with extraordinary tastes. In *Proceedings of the third ACM international conference on Web search and data mining*, WSDM '10, pages 201–210, New York, NY, USA, 2010.
13. R. Jones, R. Kumar, B. Pang, and A. Tomkins. "i know what you did last summer": query logs and user privacy. In *CIKM '07: Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, pages 909–914, New York, NY, USA, 2007. ACM.
14. P. Mika. Big Data Conferences, Here We Come!. IEEE Internet Computing, vol. 17, no. 3, May/June 2013, pp. 3-5.
15. J. Mullin. FTC commissioner: If companies don't protect privacy, we'll go to congress. *paidContent.org, the Economics of Digital Content*, Feb 2011.
16. F. Radlinski, P.N. Bennett, and E. Yilmaz. Detecting duplicate web documents using click-through data. In *Proceedings of the fourth ACM international conference on Web search and data mining* pp. 147–156, 2011.
17. N. Spirin and J. Han. Survey on web spam detection: principles and algorithms. ACM SIGKDD Explorations Newsletter archive. Volume 13 Issue 2, pp. 50–64, Dec 2011.
18. J. Surowiecki. *The Wisdom of Crowds: Why the Many Are Smarter Than the Few and How Collective Wisdom Shapes Business, Economies, Societies and Nations.* Random House, 2004.
19. L. Sweeney. k-anonymity: a model for protecting privacy. *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems*, 10(5):557–570, 2001.