Research Statement

Ilya Razenshteyn

Introduction

My research interests are in developing efficient and practical algorithms for processing large data sets. I am particularly interested in algorithms for data with geometric structure. Such structure often arises after mapping the dataset into a high-dimensional *feature vector space* (e.g., representing a text document as a vector of word counts), and makes it possible to reduce the original question to a generic geometric problem. In many applications, the dataset size and the number of features is huge[1], hence, the algorithms must be extremely efficient and scale well as the dimensionality grows. Unfortunately, most of the classic geometric algorithms are tailored to two or three dimensions and do not apply to the high-dimensional regime, due to the *exponential* dependence of the running time on the dimension.

Most of my research involves designing algorithms for high-dimensional datasets by developing *efficient representations* of geometric data: randomized hashing, sketching (succinct summarization), dimensionality reduction and others. This line of work is a "sweet spot" between theory and practice: despite requiring sophisticated tools originating in probability theory and geometric functional analysis, the resulting algorithms are often relatively simple and implementable.

Main Contributions: an Overview

- **Nearest Neighbor Search**. Perhaps the most natural task for geometric datasets is *similarity search*: to retrieve objects from a dataset similar to a query object, one needs to solve the *Nearest Neighbor Search* problem (NNS) in the feature space.

  I designed new algorithms for high-dimensional NNS, significantly improving the state of the art, both in theory and in practice. Notably, the new algorithms *provably* improve upon the best possible algorithms based on *Locality-Sensitive Hashing (LSH)* on *worst-case* data. These are the first improvements to NNS algorithms since 1998 for the Hamming distance, and since 2006 for the Euclidean distance. The main new insight is to use *data-dependent hash functions*, which are carefully tailored to a given dataset, instead of the generic, data-oblivious hash families. The core component of the above algorithms can be made efficient in practice. I have implemented it and released it as a part of FALCONN, a new similarity search library for high-dimensional data. Besides designing new algorithms, I have shown strong theoretical guarantees for *LSH Forest*, a popular heuristic for speeding-up LSH-based algorithms. Finally, I have shown several *impossibility results,* which expose limitations of current techniques for NNS and suggest avenues for further theoretical and practical improvements.
- **Sketching**. Sketching is a remarkable algorithmic technique that allows to *summarize* a large object using only a few bits. These summaries can then be used to speed up a computation on original objects or to index the dataset. For high-dimensional vectors, one is interested in sketches for estimating a given *distance function* between vectors.

  I have completely characterized distances which admit efficient sketches, if a distance is given as a *norm* (e.g., an ℓ_p norm or a matrix norm). This general theory enabled concrete progress on understanding sketching complexity of two important cases: the nuclear norm and the Earth Mover's Distance (EMD). The result for EMD is the first progress made on a question which was open since 2002. I have also used sketches as a tool for designing efficient algorithms for *weighted low rank approximation*, which is a problem arising in machine learning applications.

[1] For instance, Twitter users post around *200 billion tweets* per year, and—for the purposes of searching for similarly-looking tweets—each tweet may be mapped into a *100 000-dimensional* feature vector!

- **Sparse Recovery**. The *stable sparse recovery* problem is: given an *approximately sparse* high-dimensional vector, how many linear measurements does one need in order to reconstruct it? Through a novel connection to the nearest neighbor search and sketching, I was able to completely characterize the measurement complexity for sparse recovery with respect to a large class of distances, and, as a byproduct, obtain a new streaming algorithm for clustering.
- **Graph partitioning**. I am generally interested in algorithms that lead to real-world impact. As an intern at Microsoft Research, I have designed, implemented and tuned PUNCH: a fast algorithm for partitioning large graphs [10], which is tailored to road networks. Later PUNCH became one of the core components of the routing engine of Bing Maps (for some details see, e.g., [16]).

Nearest Neighbors Search

The Nearest Neighbor Search problem is known to suffer from the so-called "curse of dimensionality", both in theory and practice. That is, all known data structures other than the naïve linear scan require space *exponential* in the dimension. Luckily, things change drastically if one allows answers to be *approximate*. The corresponding Approximate Nearest Neighbor Search problem (ANN) admits efficient data structures with query time sub-linear in the dataset size and *polynomial* dependence on the dimension. Furthermore, the algorithms for ANN turn out to be useful even for finding *exact* nearest neighbors in practice, since for many datasets most of the data points lie much further from a query than the nearest neighbor. The classic technique for solving ANN is that of *Locality-Sensitive Hashing (LSH)* introduced by Indyk and Motwani in 1998 [22]. The technique boils down to designing good *random partitions* of the ambient space $\mathbb{R}^d$, which one uses to hash the dataset and the queries.

- **Data-dependent LSH**. In [3, 6], together with Alexandr Andoni, Piotr Indyk, and Huy Nguyen, we show new data structures for ANN over Hamming and Euclidean distances, which, for the first time, improve upon the *best possible* LSH-based data structures [14, 22]. This is accomplished by introducing and implementing *data-dependent LSH*, where we tailor random space partitions to a given dataset; in contrast, the "vanilla" LSH uses universal, data-oblivious partitions. Data-dependent space partitions are extremely popular in practice [34, 35], but our results show for the first time that such an approach gives rise to significantly better *theoretical guarantees* for the *worst-case* data[2]. The new results give the *first improvement* for ANN for the *Hamming distance* since the original LSH paper [22] back from 1998. Later, together with Alexandr Andoni, we showed that the algorithm from [3] is optimal in the data-dependent LSH framework [4].
- **Implementation**. In [7], together with Alexandr Andoni, Piotr Indyk, Thijs Laarhoven, and Ludwig Schmidt, we show how to make the core component of the above theoretical results [3, 6]—the optimal LSH for the *cosine similarity*—practical. This requires bringing together several algorithmic ideas as well as lots of careful engineering. The experimental evaluation shows that the implementation is competitive with the state of the art practical ANN algorithms on a variety of datasets. Together with Ludwig Schmidt, we released the above implementation as a part of FALCONN [11]: a new C++ library for similarity search over high-dimensional data, which includes implementations of several LSH-based data structures and significantly improves upon state of the art LSH implementations E2LSH [13] and LSHKIT [19].
- **Analyzing heuristics**. In [5], together with Alexandr Andoni and Negev Shekel Nosatzki, we analyze *LSH Forest* [15]: a popular heuristic for speeding-up LSH-based data structures. We show that (a minor modification of) LSH Forest provably outperforms vanilla LSH algorithms for the Hamming distance for the *worst-case* data. LSH Forest boils down to building a random forest of decision trees, thus, our analysis gives a new explanation of the success of decision trees in the context of Nearest Neighbor Search.

[2] The reason why this is non-trivial is that we need to show not only that "structure in a dataset helps," but that data without any structure are amenable to faster algorithms as well.

- **Time–space trade-offs**. In [8], together with Alexandr Andoni, Thijs Laarhoven, and Erik Waingarten, we extend the framework of the data-dependent LSH developed in [3, 6] to support a smooth *time–space trade-off*. At the two extremes we get: an algorithm with *near-linear* space and *sub-linear* query time ("low memory" regime), and an algorithm with polynomial space and *sub-polynomial* query time ("fast queries" regime). The "low memory" regime is arguably the most relevant in practice [30]. The resulting time–space trade-off significantly improves upon a sequence of works for various regimes: [14, 20, 22, 24, 28, 31]. Furthermore, we show that the trade-off we obtain is *optimal* for algorithms based on *hashing*.
- **Additional impossibility results**. Besides the above-mentioned impossibility results, we are able to show further lower bounds: a fine-grained lower bound on a trade-off between evaluation time and quality for a large natural class of LSH families, a tight *unconditional* lower bound for the amount of space needed for any ANN data structure, if it is allowed to make at most *two* memory probes for each query[3], and the *conditional* hardness[4] of approximate Maximum Inner Search Product problem, which arises in a number of machine learning applications [1].

Sketching

The central question we are interested in is: for which similarity measures there exist *short* sketches that allow to estimate the similarity between two vectors *well*. Oftentimes, similarity is modeled via a *norm*; in what follows we focus on this case.

- **Norms that admit good sketches**. In [2], together with Alexandr Andoni and Robert Krauthgamer, we show that a "good" sketch for a norm implies the existence of a low-distortion *embedding* into an ℓ_p space for some $0 < p \leq 2$. This completely characterizes norms that admit good sketches, since the converse is true due to the result of Indyk [21] that uses p-stable random projections. Besides a complete characterization of "sketchable" norms, our result gives a new way of proving *lower bounds* for sketches. Namely, it is enough to establish *non-embeddability* of a norm of interest into ℓ_p spaces, which is a somewhat simpler task that has been studied by many researchers over a long period of time. We use this approach to establish the *first* sketching lower bounds for two norms of interest: the nuclear norm and the Earth Mover's Distance (EMD). The latter is a popular similarity measure in computer vision [32] and natural language processing [29].
- **External applications**: **weighted low rank approximation**. In [12], together with Zhao Song and David Woodruff, we used sketching to get the first efficient algorithm *with provable guarantees* for the *weighted* low rank approximation problem. The problem was introduced to the machine learning community by Srebro and Jaakkola [33] and was motivated by collaborative filtering. Classical *unweighted* low rank approximation can be found by computing Singular Value Decomposition, while the weighted case is NP-hard. We show the first algorithm that provably computes the optimal low-rank approximation in *polynomial time*, if the weight matrix has *small rank*. The algorithm crucially uses sketches based on *random projections*.

Sparse Recovery

The *stable sparse recovery* problem considers *approximately* sparse signals. The approximation is defined with respect to a distance function of interest. For the ℓ_1 and ℓ_2 distances, the stable sparse recovery problem is well-understood: in both cases we can design a measurement matrix with $O(k \log n)$ rows, which

[3] The lower bound for two probes establishes and exploits an intriguing connection to *locally decodable codes* (LDC) [37]. In particular, we use a modification of a *quantum* lower bound for two-query LDCs from [26]. This is the first space two-probe lower bound for *any* static data structure problem, which is not polynomially worse than the corresponding one-probe lower bound.

[4] Assuming the Strong Exponential Time Hypothesis, which is a basis of many recent developments in fine-grained complexity [36].

allows to recover a n-dimensional signal that is approximately k-sparse [17]. It is known that this bound can not be improved in general [18]. However, in some applications ℓ_1 and ℓ_2 distances are inadequate: for example, compressive sensing of *astronomical images* requires sparse recovery with respect to the Earth Mover's Distance (EMD) [23].

In [9], together with Arturs Backurs, Piotr Indyk and David Woodruff, we show that sparse recovery with respect to EMD is *strictly* easier than the usual ℓ_1/ℓ_2 setting. Namely, we show how to achieve $O(k \log \log n)$ measurements, which gives a significant improvement for small values of k. Perhaps even more interestingly, we obtain this result via a general theory, which characterizes the measurement complexity of sparse recovery for *any* norm that allows *efficient sketches*. Namely, we show that in this case the measurement complexity is equal to the doubling dimension of k-sparse vectors under the norm of interest. To show this result, we crucially use the connection between sparse recovery and the *Nearest Neighbor Search* problem, in particular, we utilize known data structures for NNS over low doubling dimension metrics [27].

Future Directions

Below are the main research directions that I would like to pursue.

- My goal is to come as close as possible to building a "unified theory" of efficient representations for high-dimensional data. My past research makes first steps in this direction: in [2], we completely characterize high-dimensional norms that admit efficient sketches, and in [9] we compute sample complexity for sparse recovery for a large class of distances. Besides published work, together with Alexandr Andoni, Aleksandar Nikolov, and Erik Waingarten, we are currently working on classifying norms according to the hardness of approximate similarity search with respect to them. In particular, we have recently settled this problem for *symmetric* norms.
- The second direction is to explore how external applications would benefit from such efficient representations. The above-mentioned algorithm for the weighted low rank approximation problem [12] is an example of such an application.
- As a natural continuation of my applied work [7, 10], I would like to collaborate with applied AI and systems researchers to deploy clean, theory-inspired algorithms for processing large datasets into real-world systems.

Besides these general (and perhaps a bit vague) directions, there are many concrete open questions related to the nearest neighbor search and the other problems I have been studying, which I plan to pursue. Instead of listing all of them, let me mention two most important ones.

- **Break the hashing barrier for NNS**. Perhaps the most important theoretical challenge that my work suggests is to *break the (data-dependent) hashing barrier* for similarity search. All the known algorithms for high-dimensional NNS carefully put data points into buckets, and then, during the query stage, look at a few buckets and try data points from them. In our work [8], we obtain essentially tight bounds on space and query time for such algorithms. But can we go beyond this approach? One promising line of attack is to try to use *algebraic* techniques such as *locally decodable codes* [37] or data structures for *polynomial evaluation* [25].
- **NNS: theory vs. practice**. I plan to continue working on unifying theory and practice of similarity search. In particular, I am interested in making our theoretical results [3, 6, 8] fully practical (our paper [7] makes a first solid step in this direction). I would like to investigate how the modern hardware architectures—multi-core computers, graphic processing units (GPU), clusters of commodity machines, etc.—benefit the NNS problem. Going the opposite way, I am interested in understanding why the existing heuristics [34, 35] for NNS are so successful. I would like to find a good model that captures real-world datasets and, at the same time, enables better analysis of existing heuristics for NNS and, ultimately, leads to new algorithms.

My Publications

[1] Thomas D. Ahle, Rasmus Pagh, **Ilya Razenshteyn**, and Francesco Silvestri. "On the Complexity of Inner Product Similarity Join". In: *Proceedings of the 35th ACM Symposium on Principles of Database Systems (PODS '2016)*. Available as arXiv:1510.02824. 2016, pp. 151–164.

[2] Alexandr Andoni, Robert Krauthgamer, and **Ilya Razenshteyn**. "Sketching and Embedding are Equivalent for Norms". In: *Proceedings of the 47th ACM Symposium on the Theory of Computing (STOC '2015)*. Available as arXiv:1411.2577. 2015, pp. 479–488.

[3] Alexandr Andoni and **Ilya Razenshteyn**. "Optimal Data-Dependent Hashing for Approximate Near Neighbors". In: *Proceedings of the 47th ACM Symposium on the Theory of Computing (STOC '2015)*. Available as arXiv:1501.01062. 2015, pp. 793–801.

[4] Alexandr Andoni and **Ilya Razenshteyn**. "Tight Lower Bounds for Data-Dependent Locality-Sensitive Hashing". In: *Proceedings of the 32nd International Symposium on Computational Geometry (SoCG '2016)*. Available as arXiv:1507.04299. 2016, 9:1–9:11.

[5] Alexandr Andoni, **Ilya Razenshteyn**, and Negev Shekel Nosatzki. "LSH Forest: Practical Algorithms Made Theoretical". In: *Proceedings of the 28th ACM-SIAM Symposium on Discrete Algorithms (SODA '2017)*. 2017.

[6] Alexandr Andoni, Piotr Indyk, Huy L. Nguyen, and **Ilya Razenshteyn**. "Beyond Locality-Sensitive Hashing". In: *Proceedings of the 25th ACM-SIAM Symposium on Discrete Algorithms (SODA '2014)*. Available as arXiv:1306.1547. 2014, pp. 1018–1028.

[7] Alexandr Andoni, Piotr Indyk, Thijs Laarhoven, **Ilya Razenshteyn**, and Ludwig Schmidt. "Practical and Optimal LSH for Angular Distance". In: *Proceedings of Advances in Neural Information Processing Systems 28 (NIPS '2015)*. Available as arXiv:1509.02897. 2015, pp. 1225–1233.

[8] Alexandr Andoni, Thijs Laarhoven, **Ilya Razenshteyn**, and Erik Waingarten. "Optimal Hashing-based Time–Space Trade-offs for Approximate Near Neighbors". In: *Proceedings of the 28th ACM-SIAM Symposium on Discrete Algorithms (SODA '2017)*. Available as arXiv:1608.03580. 2017.

[9] Arturs Backurs, Piotr Indyk, **Ilya Razenshteyn**, and David P. Woodruff. "Nearly-optimal bounds for sparse recovery in generic norms, with applications to k-median sketching". In: *Proceedings of the 27th ACM-SIAM Symposium on Discrete Algorithms (SODA '2016)*. Available as arXiv:1504.01076. 2016, pp. 318–337.

[10] Daniel Delling, Andrew V. Goldberg, **Ilya Razenshteyn**, and Renato F. Werneck. "Graph Partitioning with Natural Cuts". In: *Proceedings of the 25th IEEE International Symposium on Parallel and Distributed Processing (IPDPS '2011)*. 2011, pp. 1135–1146.

[11] **Ilya Razenshteyn** and Ludwig Schmidt. *FALCONN: Similarity Search over High-Dimensional Data*. Available as `https://falconn-lib.org/`. 2015.

[12] **Ilya Razenshteyn**, Zhao Song, and David P. Woodruff. "Weighted low rank approximations with provable guarantees". In: *Proceedings of the 48th ACM Symposium on the Theory of Computing (STOC '2016)*. 2016, pp. 250–263.

Related Work

[13] Alexandr Andoni and Piotr Indyk. *E2LSH: Exact Euclidean Locality-Sensitive Hashing*. Available as `http://web.mit.edu/andoni/www/LSH/index.html`. 2005.

[14] Alexandr Andoni and Piotr Indyk. "Near-Optimal Hashing Algorithms for Approximate Nearest Neighbor in High Dimensions". In: *Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS '2006)*. 2006, pp. 459–468.

[15] Mayank Bawa, Tyson Condie, and Prasanna Ganesan. "LSH forest: self-tuning indexes for similarity search". In: *Proceedings of the 14th international conference on World Wide Web (WWW '2005)*. 2005, pp. 651–660.

[16] *Bing Maps New Routing Engine*. Available at `https://blogs.bing.com/maps/2012/01/05/bing-maps-new-routing-engine`. 2012.

[17] Emmanuel J. Candes, Justin K. Romberg, and Terence Tao. "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information". In: *IEEE Transactions on Information Theory* 52.2 (2006), pp. 489–509.

[18] Khanh Do Ba, Piotr Indyk, Eric Price, and David P. Woodruff. "Lower Bounds for Sparse Recovery". In: *Proceedings of the 21st ACM-SIAM Symposium on Discrete Algorithms (SODA '2010)*. 2010, pp. 1190–1197.

[19] Wei Dong. *LSHKIT: A C++ Locality Sensitive Hashing Library*. Available as `http://lshkit.sourceforge.net/`. 2009.

[20] Piotr Indyk. "High-Dimensional Computational Geometry". PhD thesis. Stanford University, 2001.

[21] Piotr Indyk. "Stable distributions, pseudorandom generators, embeddings, and data stream computation". In: 53.3 (2006), pp. 307–323.

[22] Piotr Indyk and Rajeev Motwani. "Approximate Nearest Neighbors: Towards Removing the Curse of Dimensionality". In: *Proceedings of the 30th ACM Symposium on the Theory of Computing (STOC '1998)*. 1998, pp. 604–613.

[23] Piotr Indyk and Eric Price. "K-median clustering, model-based compressive sensing, and sparse recovery for earth mover distance". In: *Proceedings of the 43rd ACM Symposium on the Theory of Computing (STOC '2011)*. 2011, pp. 627–636.

[24] Michael Kapralov. "Smooth Tradeoffs between Insert and Query Complexity in Nearest Neighbor Search". In: *Proceedings of the 34th ACM Symposium on Principles of Database Systems (PODS '2015)*. 2015, pp. 329–342.

[25] Kiran Kedlaya and Christopher Umans. "Fast Polynomial Factorization and Modular Composition". In: *SIAM Journal on Computing* 40.6 (2011), pp. 1767–1802.

[26] Iordanis Kerenidis and Ronald de Wolf. "Exponential lower bound for 2-query locally decodable codes via a quantum argument". In: *Journal of Computer and System Sciences* 69.3 (2004), pp. 395–420.

[27] Robert Krauthgamer and James R. Lee. "Navigating nets: simple algorithms for proximity search". In: *Proceedings of the 15th ACM-SIAM Symposium on Discrete Algorithms (SODA '2004)*. 2004, pp. 798–807.

[28] Eyal Kushilevitz, Rafail Ostrovsky, and Yuval Rabani. "Efficient Search for Approximate Nearest Neighbor in High Dimensional Spaces". In: *SIAM Journal on Computing* 30.2 (2000), pp. 457–474.

[29] Matt J. Kusner, Yu Sun, Nicholas I. Koklin, and Kilian Q. Weinberger. "From Word Embeddings To Document Distances". In: *Proceedings of the 32nd International Conference on Machine Learning (ICML '2015)*. 2015, pp. 957–966.

[30] Qin Lv, William Josephson, Zhe Wang, Moses Charikar, and Kai Li. "Multi-Probe LSH: Efficient Indexing for High-Dimensional Similarity Search". In: *Proceedings of the 33rd International Conference on Very Large Data Bases (VLDB '2007)*. 2007, pp. 950–961.

[31] Rina Panigrahy. "Entropy based nearest neighbor search in high dimensions". In: *Proceedings of the 17th ACM-SIAM Symposium on Discrete Algorithms (SODA '2006)*. 2006, pp. 1186–1195.

[32] Yossi Rubner, Carlo Tomasi, and Leonidas J. Guibas. "The Earth Mover's Distance as a Metric for Image Retrieval". In: *International Journal of Computer Vision* 40.2 (2000), pp. 99–121.

[33] Nathan Srebro and Tommi Jaakkola. "Weighted Low-Rank Approximations". In: *Proceedings of the 20th International Conference on Machine Learning (ICML '2003)*. 2003, pp. 720–727.

[34] Jingdong Wang, Heng Tao Shen, Kingkuan Song, and Jianqiu Ji. "Hashing for Similarity Search: a Survey". Available as arXiv:1408.2927. 2014.

[35] Jun Wang, Wei Liu, Sanjiv Kumar, and Shih-Fu Chang. "Learning to Hash for Indexing Big Data - A Survey". Available as arXiv:1509.05472. 2015.

[36] Virginia Vassilevska Williams. "Hardness of Easy Problems: Basing Hardness on Popular Conjectures such as the Strong Exponential Time Hypothesis (Invited Talk)". In: *Proceedings of the 10th International Symposium on Parameterized and Exact Computation (IPEC '2015)*. 2015, pp. 17–29.

[37] Sergey Yekhanin. *Locally Decodable Codes*. Vol. 6. Foundations and Trends in Theoretical Computer Science 3. Now Publishers, 2012.