课程报告要求：

1. 分组人数：3-5人，每个小组选一个主题。比如：1.1图数据处理系统，或者动态图。
2. 阅读内蓉：每人至少阅读一篇论文。
3. 汇报内容包括：（1）问题及挑战；（2）相关工作及对比分析；（3）技术介绍：每篇论文的技术细节及实验。（4）总结及趋势分析。
4. 提交word文档报告；
5. 补充：除了下面的主题列表，与课程相关的研究主题，也可以作为候选主题。所选论文建议2016年至今的新论文。

# 图数据处理

## 1.1系统：

1. Chi Y, Dai G, Wang Y, et al. NXgraph: An efficient graph processing system on a single machine[C]// IEEE, International Conference on Data Engineering. IEEE, 2016:409-420.
2. Yan D, Cheng J, Lu Y, et al. Blogel: A Block-Centric Framework for Distributed Computation on Real-World Graphs[C]// VLDB Endowment. 2014:1981-1992.
3. Borkar V, Borkar V, Jia J, et al. Pregelix: Big(ger) graph analytics on a dataflow engine[J]. Proceedings of the Vldb Endowment, 2014, 8(2):161-172.
4. Sundaram N, Satish N, Patwary M M A, et al. GraphMat: high performance graph analytics made productive[J]. Proceedings of the Vldb Endowment, 2015, 8(11):1214-1225.
5. Fan W, Lu P, Luo X, et al. Adaptive asynchronous parallelization of graph algorithms[C]//Proceedings of the 2018 International Conference on Management of Data. ACM, 2018: 1141-1156.
6. Fan W, Yu W, Xu J, et al. Parallelizing sequential graph computations[J]. ACM Transactions on Database Systems (TODS), 2018, 43(4): 18.

## 1.2算法：

1. Peng Y, Choi B, He B, et al. VColor: A practical vertex-cut based approach for coloring large graphs[C]// IEEE, International Conference on Data Engineering. IEEE, 2016:97-108.
2. Lyu B, Qin L, Lin X, et al. Scalable supergraph search in large graph databases[C]// IEEE, International Conference on Data Engineering. IEEE, 2016:157-168.
3. Martella C, Logothetis D, Loukas A, et al. Spinner: Scalable Graph Partitioning in the Cloud[C]// IEEE, International Conference on Data Engineering. IEEE, 2017:1083-1094.
4. Yan D, Cheng J, Yang F, et al. A general-purpose query-centric framework for querying big graphs[J]. Proceedings of the Vldb Endowment, 2016, 9(7):564-575.
5. Dubey A, Hill G D, Escriva R. Weaver: a high-performance, transactional graph database based on refinable timestamps [J]. Proceedings of the Vldb Endowment, 2016, 9(11):852-863.
6. Ma H, Shao B, Xiao Y, et al. G-SQL: fast query processing via graph exploration[J]. Proceedings of the Vldb Endowment, 2016, 9(12):900-911.
7. Boehm M, Dusenberry M W, Eriksson D, et al. SystemML: declarative machine learning on spark[J]. Proceedings of the Vldb Endowment, 2016, 9(13):1425-1436.
8. Sutanay Choudhury, Lawrence Holder, George Chin, Khushbu Agarwal, and John Feo. A selectivity based approach to continuous pattern detection in streaming graphs. EDBT, 2015.
9. Sun S, Che Y, Wang L, et al. Efficient Parallel Subgraph Enumeration on a Single Machine[C]//2019 IEEE 35th International Conference on Data Engineering (ICDE). IEEE, 2019: 232-243.
10. Qiao M, Zhang H, Cheng H. Subgraph matching: on compression and computation[J]. Proceedings of the VLDB Endowment, 2017, 11(2): 176-188.
11. Lai L, Qin L, Lin X, et al. Scalable distributed subgraph enumeration[J]. Proceedings of the VLDB Endowment, 2016, 10(3): 217-228.
12. Bi F, Chang L, Lin X, et al. Efficient subgraph matching by postponing cartesian products[C]//Proceedings of the 2016 International Conference on Management of Data. ACM, 2016: 1199-1214.
13. Mhedhbi A, Salihoglu S. Optimizing subgraph queries by combining binary and worst-case optimal joins[J]. arXiv preprint arXiv:1903.02076, 2019.

## 1.3动态图：

1. Ma S, Hu R, Wang L, et al. Fast Computation of Dense Temporal Subgraphs[C]// IEEE, International Conference on Data Engineering. IEEE, 2017:361-372.
2. Zhou Y, Liu L, Lee K, et al. GraphTwist: fast iterative graph computation with two-tier optimizations[J]. Proceedings of the Vldb Endowment, 2015, 8(11):1262-1273.
3. Huang J, Abadi D J. Leopard: lightweight edge-oriented partitioning and replication for dynamic graphs[M]. VLDB Endowment, 2016.
4. Peter Macko, Virendra J. Marathe, Daniel W. Margo, and Margo I. Seltzer. LLAMA: efficient graph analytics using large multiversioned arrays. In ICDE, pages 363–374, 2015.
5. Udayan Khurana and Amol Deshpande. Storing and analyzing historical graph data at scale. In EDBT, pages 65–76, 2016.

# 二、近似查询处理

## 2.1系统：

1. **Yongjoo Park, Barzan Mozafari, Joseph Sorenson, and Junhao Wang. 2018. VerdictDB: Universalizing Approximate Query Processing. In Proceedings of the 2018 International Conference on Management of Data (SIGMOD '18).**
2. **S. Agarwal, B. Mozafari, A. Panda, H. Milner, S. Madden, and I. Stoica. BlinkDB: queries with bounded errors and bounded response times on very large data. In EuroSys, 2013.**
3. **S. Agarwal, H. Milner, A. Kleiner, A. Talwalkar, M. Jordan, S. Madden, B. Mozafari, and I. Stoica. Knowing when you’re wrong: Building fast and reliable approximate query processing systems. In SIGMOD, 2014.**
4. **B. Ding, S. Huang, S. Chaudhuri, K. Chakrabarti, and C. Wang. Sample + seek: Approximating aggregates with distribution precision guarantee. In SIGMOD, 2016.**
5. **Jinglin Peng, Dongxiang Zhang, Jiannan Wang, and Jian Pei. 2018. AQP++: Connecting Approximate Query Processing With Aggregate Precomputation for Interactive Analytics. In Proceedings of the 2018 International Conference on Management of Data (SIGMOD '18).**
6. **S. Kandula, A. Shanbhag, A. Vitorovic, M. Olma, R. Grandl, S. Chaudhuri, and B. Ding. Quickr: Lazily approximating complex adhoc queries in bigdata clusters. In SIGMOD, 2016.**

## 2.2近似算法：

1. **J. Acharya, I. Diakonikolas, C. Hegde, J. Z. Li, and L. Schmidt. Fast and near-optimal algorithms for approximating distributions by histograms. In PODS, 2015.**
2. **A. Galakatos, A. Crotty, E. Zgraggen, C. Binnig, and T. Kraska. Revisiting reuse for approximate query processing. PVLDB, 2017.**
3. **Y. Cao, W. Fan, and C. Hu. Data driven approximation with bounded resources. PVLDB, 10, 2017.**
4. **Xiaochun Yang, Yaoshu Wang, Bin Wang, and Wei Wang. 2015. Local Filtering: Improving the Performance of Approximate Queries on String Collections. In Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data (SIGMOD '15).**
5. **N. Potti and J. M. Patel. Daq: a new paradigm for approximate query processing. PVLDB, 2015.**
6. **H. Su, M. Zait, V. Barrière, J. Torres, and A. Menck. Approximate aggregates in oracle 12c, 2016.**
7. **D. Moritz, D. Fisher, B. Ding, and C. Wang. Trust, but verify: Optimistic visualizations of approximate queries for exploring big data. In CHI, 2017.**
8. **W. Gatterbauer and D. Suciu. Approximate lifted inference with probabilistic databases. PVLDB, 2015.**

## 2.3采样技术：

1. **Y. Chen and K. Yi. Two-level sampling for join size estimation. In SIGMOD, 2017.**
2. **W. Fan, F. Geerts, Y. Cao, T. Deng, and P. Lu. Querying big data by accessing small data. In PODS, 2015.**
3. **S. Chaudhuri, B. Ding, and S. Kandula. Approximate query processing: No silver bullet. In SIGMOD, 2017.**
4. **N. Kamat and A. Nandi. A session-based approach to fast-but-approximate interactive data cube exploration. ACM Trans. Knowl. Discov. Data, 12(1):9:1–9:26, Feb. 2018.**

# 三、AI使能的数据库

## 3.1索引：

1. **Kraska T, Beutel A, Chi E H, et al. The case for learned index structures[C]//Proceedings of the 2018 International Conference on Management of Data. ACM, 2018: 489-504.**
2. **Galakatos A, Markovitch M, Binnig C, et al. Fiting-tree: A data-aware index structure[C]//Proceedings of the 2019 International Conference on Management of Data. ACM, 2019: 1189-1206.**

## 3.2查询：

1. **Ortiz J, Balazinska M, Gehrke J, et al. Learning state representations for query optimization with deep reinforcement learning[J]. arXiv preprint arXiv:1803.08604, 2018.**
2. **Kipf A, Kipf T, Radke B, et al. Learned cardinalities: Estimating correlated joins with deep learning[J]. arXiv preprint arXiv:1809.00677, 2018.**
3. **Krishnan S, Yang Z, Goldberg K, et al. Learning to optimize join queries with deep reinforcement learning[J]. arXiv preprint arXiv:1808.03196, 2018.**
4. **Marcus R, Papaemmanouil O. Plan-structured deep neural network models for query performance prediction[J]. arXiv preprint arXiv:1902.00132, 2019.**
5. **Park Y, Zhong S, Mozafari B. Quicksel: Quick selectivity learning with mixture models[J]. arXiv preprint arXiv:1812.10568, 2018.**
6. **Liang X, Elmore A J, Krishnan S. Opportunistic view materialization with deep reinforcement learning[J]. arXiv preprint arXiv:1903.01363, 2019.**
7. **Kipf A, Vorona D, Müller J, et al. Estimating Cardinalities with Deep Sketches[J]. arXiv preprint arXiv:1904.08223, 2019.**
8. **Kipf A, Freitag M, Vorona D, et al. Estimating Filtered Group-By Queries is Hard: Deep Learning to the Rescue[J].**
9. **Sun J, Li G. An End-to-End Learning-based Cost Estimator[J]. arXiv preprint arXiv:1906.02560, 2019.**
10. **Marcus R, Negi P, Mao H, et al. Neo: A learned query optimizer[J]. arXiv preprint arXiv:1904.03711, 2019.**

## 3.3调优：

1. **Idreos S, Dayan N, Qin W, et al. Design Continuums and the Path Toward Self-Designing Key-Value Stores that Know and Learn[C]//CIDR. 2019.**
2. **Zhang J, Liu Y, Zhou K, et al. An end-to-end automatic cloud database tuning system using deep reinforcement learning[C]//Proceedings of the 2019 International Conference on Management of Data. ACM, 2019: 415-432.**
3. **Li G, Zhou X, Li S, et al. QTune: a query-aware database tuning system with deep reinforcement learning[J]. Proceedings of the VLDB Endowment, 2019, 12(12): 2118-2130.**
4. **Ma L, Van Aken D, Hefny A, et al. Query-based workload forecasting for self-driving database management systems[C]//Proceedings of the 2018 International Conference on Management of Data. ACM, 2018: 631-645.**
5. **Van Aken D, Pavlo A, Gordon G J, et al. Automatic database management system tuning through large-scale machine learning[C]//Proceedings of the 2017 ACM International Conference on Management of Data. ACM, 2017: 1009-1024.**
6. **Tan J, Zhang T, Li F, et al. iBTune: individualized buffer tuning for large-scale cloud databases[J]. Proceedings of the VLDB Endowment, 2019, 12(10): 1221-1234.**

# 四、分布式数据管理与区块链

## 4.1存储：

1. **Xu Z, Han S, Chen L. CUB, a consensus unit-based storage scheme for blockchain system[C]//2018 IEEE 34th International Conference on Data Engineering (ICDE). IEEE, 2018: 173-184.**
2. **Wang S, Dinh T T A, Lin Q, et al. Forkbase: An efficient storage engine for blockchain and forkable applications[J]. Proceedings of the VLDB Endowment, 2018, 11(10): 1137-1150.**
3. **Zheng J, Lin Q, Xu J, et al. PaxosStore: high-availability storage made practical in WeChat[J]. Proceedings of the VLDB Endowment, 2017, 10(12): 1730-1741.**
4. **Ali M, Nelson J, Shea R, et al. Blockstack: A global naming and storage system secured by blockchains[C]//2016 {USENIX} Annual Technical Conference ({USENIX}{ATC} 16). 2016: 181-194.**
5. **Dang H, Dinh T T A, Loghin D, et al. Towards scaling blockchain systems via sharding[C]//Proceedings of the 2019 International Conference on Management of Data. ACM, 2019: 123-140.**
6. **Nathan S, Govindarajan C, Saraf A, et al. Blockchain Meets Database: Design and Implementation of a Blockchain Relational Database[J]. arXiv preprint arXiv:1903.01919, 2019.**
7. **El-Hindi M, Binnig C, Arasu A, et al. BlockchainDB: a shared database on blockchains[J]. Proceedings of the VLDB Endowment, 2019, 12(11): 1597-1609.**

## 4.2分布式协议：

1. **Eyal I, Gencer A E, Sirer E G, et al. Bitcoin-ng: A scalable blockchain protocol[C]//13th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 16). 2016: 45-59.**
2. **Dziembowski S, Eckey L, Faust S. Fairswap: How to fairly exchange digital goods[C]//Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security. ACM, 2018: 967-984.**
3. **Kiayias A, Russell A, David B, et al. Ouroboros: A provably secure proof-of-stake blockchain protocol[C]//Annual International Cryptology Conference. Springer, Cham, 2017: 357-388.**
4. **Gilad Y, Hemo R, Micali S, et al. Algorand: Scaling byzantine agreements for cryptocurrencies[C]//Proceedings of the 26th Symposium on Operating Systems Principles. ACM, 2017: 51-68.**

## 4.3系统视角：

1. **Dinh T T A, Wang J, Chen G, et al. Blockbench: A framework for analyzing private blockchains[C]//Proceedings of the 2017 ACM International Conference on Management of Data. ACM, 2017: 1085-1100.**
2. **Dinh T T A, Liu R, Zhang M, et al. Untangling blockchain: A data processing view of blockchain systems[J]. IEEE Transactions on Knowledge and Data Engineering, 2018, 30(7): 1366-1385.**
3. **Dinh T T A, Liu R, Zhang M, et al. Untangling blockchain: A data processing view of blockchain systems[J]. IEEE Transactions on Knowledge and Data Engineering, 2018, 30(7): 1366-1385.**
4. **Ruan P, Chen G, Dinh T T A, et al. Fine-grained, secure and efficient data provenance on blockchain systems[J]. Proceedings of the VLDB Endowment, 2019, 12(9): 975-988.**
5. **Amiri M J, Agrawal D, Abbadi A E. CAPER: a cross-application permissioned blockchain[J]. Proceedings of the VLDB Endowment, 2019, 12(11): 1385-1398.**