No.	Technical field or topic
1	Optimization of storage and processing of sparse matrices
2	Adaptive ODE solvers
3	Sparse direct solvers
4	Elementary math functions calculation
5	Direct solvers
6	Eigenvalue/singular value solvers
7	Sparse eigenvalue/singular value solvers
8	Optimization of processing of dense matrices

9	Hybrid computing usings a neural networks with external memory
10	Probabilistic Knowledge Graphs
11	Machine learning, Few-shot learning
12	AI
13	Data-efficient computing
14	Energy-efficient computing

15	Acceleration of ARM-based machine learning algorithms
16	Remote Direct Memory Access (RDMA) for Distributed Large Data Streaming Clusters
17	AI Challenges
18	Formal verification of class libraries
19	Probabilistic Knowledge Graphs

20	Hybrid computing usings a neural networks with external memory
21	AutoML in “general sense”, i.e. automated methods of ML pipeline building, including automatic feature generation, model selection using Bayesian optimization, special cases of genetic algorithms, etc.
22	Zero-shot and few-shot learning for fast and efficient tuning of ML pipeline to new tasks, new data, new languages (if we work in the computational linguistics area)
23	Bayesian methods of deep learning, including models with latent variables, GANs, stochastic neural networks and other Bayesian approaches to model an epistemic and aleatory uncertainty, to do distillation of ML ensembles, etc.

Requirement Descriptition of Field or Topic
design and develop sparse matrix storage schemes that perform well on a given set of matrix-vector operations for given ARM architecture, design and develop algorithms for sparse matrix structure analysis, may research applicability of artificial intelligence techniques to solve the problems indicated.
design and develop ODE solvers that perform well for broad set of problems for given ARM architecture, design and develop algorithms for automatic selection of most suitable ODE solver at the given integration interval following the current problem properties. May consider cluster implementation of the algorithms.
design and develop sparse direct solvers that perform well for broad set of problems (e.g. taken from SuiteSparse) for given ARM architecture, design and develop algorithms for matrix reordering given smallest memory footprint and maximal parallelization for LU decomposition, design and develop algorithms that do fast LU solve step (e.g. using DAG technique). May consider cluster implementation of the algorithms.
design and develop IEEE-754 compliant elementary math function calculations (trigonometric, hyperbolic, logarithmic, power, etc) that perform well for the full range of machine-representable float and double numbers for given ARM architecture, design and develop algorithms for validation of IEEE-754 compliancy and performance.
design and develop LAPACK direct solvers that perform well for broad set of sizes for given ARM architecture, design and develop algorithms for matrix reordering maximal stability for LU decomposition, design and develop algorithms that do fast LU solve step (e.g. using DAG technique). May consider cluster implementation of the algorithms.
design and develop LAPACK eigenvalue/singular value solvers that perform well for broad set of sizes for given ARM architecture, design and develop algorithms with maximal stability properties, design and develop algorithms that do fast solve step (e.g. using DAG technique). May consider cluster implementation of the algorithms.
design and develop sparse eigenvalue/singular value solvers (on a given interval like FEAST or Krylov subspaces based solvers like in LAPACK) that perform well for broad set of sizes for given ARM architecture, design and develop algorithms with maximal stability properties, design and develop algorithms that do fast solve step (e.g. using DAG technique). May consider cluster implementation of the algorithms.
design and develop dense matrix storage schemes like those from BLAS that perform well on a given set of matrix-vector operations for given ARM architecture, design and develop algorithms for broad set of sizes, may research applicability of artificial intelligence techniques to solve the problems indicated.

Artificial neural networks are remarkably adept at sensory processing, sequence learning and reinforcement learning, but are limited in their ability to represent variables and data structures and to store data over long timescales, owing to the lack of an external memory. Design and develop architectures and algorithms that combine neural networks with dynamic/static external memory.
Design and develop probabilistic extensions of embedding models for link prediction in relational knowledge graphs. Design and develop NLP systems that use knowledge graphs to solve downstream tasks line entity recognition, coreference resolution, e.t.c
Telecom-network data volume is huge. It will be very hard to get quite few abnormal samples from such huge volume data. The data labeling is high cost, lots money and labor work. We hope automatic labeling in the future. Telecom-network has high reliability, hard to get failure samples, especially for critical failures. E.g., there could be no key node failure sample in one certain region. We hope finishing the training at the cloud in the future. Based on this context, we want to achieve the good result with few samples volume, which needs the breakthrough of few-shot learning AI technology.
Understanding of machine/deep learning principle deeply. so that some algorithms can be deployed into our smart devices to improve the performance of some applications, such as cameral, video. the algorithms can be computer vision(segmentation, face recognition, Demosaic, superResolution and so on), imaging process(denoising, HDR), model acceleration(quantization, prunning, distillition), NAS(neuronetwork automatic search) and synthetic data generation etc.
To achieve excellent performance, a large amount of data needs to be marked. Reducing the dependence of machine learning on a large amount of labeled data is of great significance to the actual scenario. The topics are: improving the training model quality and data validity evaluation, under the circumstances like insufficient data, poor quality data, data label inconsistency, and extreme imbalance between positive and negative samples due to the dynamic attributes of the network system.
Machine learning with low power consumption are desired in many places, such as base stations, mobile phones, and automobiles. Breakthroughs in energy efficiency theories and technologies can greatly improve the application scope of artificial intelligence. The research topics include quantization compression, knowledge distillation, neural architecture search, and pruning.

Algorithms like SparkML/XGBoost were tested on the ARM platform, to compare with based on X86 platform machine learning acceleration environment, there are large differences in performance. Optimization and adaptation of machine learning algorithms on ARM platform can bring about a performance improvement by more than 50-300%. The goal: on the assumption that computing power and evaluation indicators (such as precision, recall etc.) stay constant, an optimized ARM-based algorithm compared with the most efficient open source algorithm running on x86 has more than 50% better training performance.
Performance: today's data processing stacks employ many software layers, which is key to making the stacks modular and flexible to use. But the software layers also impose overheads during I/O operations that prevent applications from enjoying the full potential of the high-performance hardware. To eliminate these overheads, I/O operations must interact with the hardware directly from within the application context using principles like RDMA. The goal is to increase cluster performance by at least twice with dynamic data distribution between cluster nodes based on RDMA technologies
1. One-shot and 0-shot Training There are very few training samples in many industries, not every use case allow batch training due to Data Privacy In many cases like IoT, data is arrived in streaming 2. Explainable Deep Neural Network Transform AI from train-by-luck to an train-by-engineering Reason about relationship between Data features and model effectiveness
The ultimate goal is the creation of a cost-effective technology of formal verification of class libraries implemented in industrial programming languages. Under verification we mean computer-aided automatic proof of conformance of a class library implementation to its formal specification. Basically, we are seeking co-operation in the area of automatic reasoning applied to partially evaluated programs. Particular formal methods may vary from symbolic computation to deductive verification backed by one of the mainstream theorem proving systems. The only requirement is that the proof should be accomplished without modifying/manually re-writing the residual program. Examples are available upon request.
Design and develop probabilistic extensions of embedding models for link prediction in relational knowledge graphs. Design and develop NLP systems that use knowledge graphs to solve downstream tasks line entity recognition, coreference resolution, e.t.c.

Artificial neural networks are remarkably adept at sensory processing, sequence learning and reinforcement learning, but are limited in their ability to represent variables and data structures and to store data over long timescales, owing to the lack of an external memory. Design and develop architectures and algorithms that combine neural networks with dynamic/static external memory.