1. What is the key need for this algorithm/approach/system? What are some short comings of existing approaches?

The capabilities of statistical machine learning are constrained by the computing time. When the computation complexity for large datasets becomes higher, the old method cannot deal with it. Therefore, we need a new idea to solve it such that stochastic gradient descent might the answer to this problem.

2. What is the key object (term) that is the solution to this need? (e.g. tweet, RDD, Tensor) Describe this object in a few sentences. This is the fundamental concept or "thing" proposed to solve the need addressed in question #1.

Cost function and its gradient is necessary when implementing this method. The goal of gradient descent is to minimize the cost function by using gradient. As the cost becomes lower, our model might get higher accuracy.

3. What has the author identified as a weakness or limitation of the proposed algorithm / approach / system? Or what has the author proposed as next steps? If the author does not provide this information, what do you think could be improved?

When the data set is very large, calculating the gradient will be very expensive, and the step size is still a critical issue. Stochastic gradient descent or batch stochastic gradient descent become very popular recently since they both theoretically will converge to local minimum without calculating all the data. Hence, it will save lots of computing power and time if we set the appropriate size of each batch. Moreover, how to get out of the local minimum is another issue. The simplest way is to reset the start point and converge it again.

4. What is something interesting your learned from this paper, or your thoughts about its strengths and/or weaknesses. Is there anything else interesting about this paper, or your interpretation of it that you want to share?

I am interested in how to tune the step size. The step size will be an issue since if we would like to find the global minimum, the step cannot be too large because it might cross it. However, if the step is too small, it takes much time to get global minimum. Moreover, it is very often to get stuck in the local minimum because of the step size. However, the performance of the stochastic gradient descent is quite well in large scale learning if we use the correct hyperparameter. Therefore, it is popular in machine learning area.

5. Read an additional related paper. Provide a citation for this paper. How does this paper relate to the assigned one? What new information did you learn by reading it? This answer should be a decent sized paragraph describing the content of this paper (be specific) and how it relates to the assigned paper.

Andrychowicz, M., Denil, M., Gomez, S., Hoffman, M. W., Pfau, D., Schaul, T., Shillingford, B. & de Freitas, N. (2016). Learning to learn by gradient descent by gradient descent. Presented at the 2016 Neural Information Processing Systems conference, Barcelona, Spain, December 5–10, 2016.
In: Advances in neural information processing systems 29 (NIPS 2016), ed. Lee, D. D., Sugiyama, M., Luxburg, U. V.,Guyon, I. & Garnett, R., pp. 3981–89). Neural Information Processing Systems.

In this paper, I learned lots of different learning strategies such as RMSprop, ADAM and Momentum. The authors also implemented how to label picture by using LSTM and RNN. Although they are different from vanilla gradient descent, they are all gradient descent optimization algorithms. From this paper, I understand the limitation of the SGD and how to use optimizer to overcome them. There is no perfect one, so learning which kind of optimizer I should choose in certain situation is very important. Nevertheless, optimizers are infeasible to calculate in practice for high-dimensional data sets. Therefore, although there are lots of different learning methods, building a correct and efficient model is still a difficult task.