# THESIS PROPOSAL: DEEP LEARNING TASK RECOMMENDATION FOR CROWDWORKERS

by

Riley Miller

A thesis submitted to the Faculty and the Board of Trustees of the Colorado School of Mines in partial fulfillment of the requirements for the degree of Master of Science (Computer Science).

Golden, Colorado

Date ____________________

Signed: ____________________
Riley Miller

Signed: ____________________
Dr. Chuan Yue
Thesis Advisor

Golden, Colorado

Date ____________________

Signed: ____________________
Dr. Tracy Camp
Department Head
Department of Computer Science

ABSTRACT

TODO: Adding after draft of proposal

TABLE OF CONTENTS

# CHAPTER 1
# RESEARCH PROPOSAL

The following contains my proposal for the research I plan to conduct under the guidance of Dr. Chuan Yue in pursuit of my Masters of Computer Science degree from Colorado School of Mines. My research will be centered around the development of a machine learning approach to improve the efficiency of crowdworkers in crowdsourcing platforms.

[**?** ]

## 1.1 Introduction

In this section I will outline the problem that we are trying to solve, provide a high level overview of crowdsourcing platforms, ecosystems, and tools, as well as highlight traditional approaches to developing recommendation systems.

### 1.1.1 Background

Crowdsourcing platforms have become increasingly popular over the course of the last several years, introducing the concept of utilizing "crowd" intelligence to complete work. These platforms are historically architected to facilitate the relationship between requesters, who create work on the platform in the form of items commonly referred to as tasks, and crowdworkers who are real people that complete tasks that are curated by the requesters. There are a variety of different crowdsourcing platforms that have entered the crowdsourcing market such as Figure Eight, ClickWorker, Spare5, Respondent, Swagbucks, and the most popular platform: Amazon Mechanical Turk (MTURK). These platforms are used for a variety of different reasons ranging from collecting data via surveys, image/video annotation, translations, spreadsheet modifications and analysis, writing, to really anything imagineable. The typical workflow in crowdsourcing platforms starts with requesters who break work that they need to get completed down into tasks. This process usually involves writing a description of what needs to be done, how it needs to be completed, providing the input the crowdworker needs to complete the work, setting a qualification so that only certain crowdworkers may have access to the task based off their work history, setting a lifetime on the task for when the work needs to be completed, and setting a reward for successful completion of the task. All of these specifications vary by platform since a universal specification standard between different crowdsourcing platforms fails to exist. After the requesters finish a task and publish it to the crowdsourcing platform, the task becomes available to crowdworkers (depending on the platform it may only be made available to crowdworkers who match the qualification set by the requesters) who can then attempt to complete the task. After the crowdworkers are satisfied with their work on a given task they can then submit their work on a task to the requesters for approval. After this work is received by the requesters, the requesters have the ability to review the work that was completed by the crowdworker and determine whether they will accept the work, reject the work, or send the task back to the crowdworker because the quality of the work was insufficient. This workflow in crowdsourcing platforms presents several different areas that need to be improved in order for crowdsourcing platforms to continue to grow.

### 1.1.2 Existing Tools

Provide context on existing tools in the crowdsourcing ecosystem

- This little listy went to market
- This little listy stayed home
- This little listy had roast beef
- This little listy had none
- And this little listy graduated, and went "wee wee wee" all the way home

#### 1.1.3 Approaches in Recommendation Systems

Provide context on traditional reccomender systems

##### 1.1.3.1 Collaborative Filtering

##### 1.1.3.2 Content Based

### 1.2 Related Work

related work on crowdsourcing platforms and recommender systems. Think about what I'd really like to include. No closely related work for crowdsourcing recommender systems. Check that there aren't any existing (Google Scholar/Google Search). Include more work on the recommender systems.

### 1.3 Proposed Work

Explain that Dr. Yue will be collecting the data needed for reccomendation system.

#### 1.3.1 Dataset

Review 5 different types of recommendation systems. Focus more on what type of algorithm/approach I should consider. Itemize potential approaches. Existing approaches have open data sets. Could explore data set for algorithms used in proposed work. Try corresponding approach. Talk about how the team is collecting the data and how that will tie to the crowdsourcing platform. Speak on how I will adapt or create a new algorithm based on the nature of the crowdsourcing platform. More concrete from a technical perspective. Talk about what types of metric I plan on using. Consider all of the metrics discussed with Dr. Yue.

### 1.3.2 Stand-alone Deep Learning

### 1.3.3 Shallow Learning

### 1.3.4 Content-based Deep Learning (NLP)

### 1.3.5 Hybrid Approach

### 1.3.6 Training and Accuracy

## 1.4 Outline of Project Timeline

- This little listy went to market
- This little listy stayed home
- This little listy had roast beef
- This little listy had none
- And this little listy graduated, and went ”wee wee wee“ all the way home

REFERENCES CITED


[] H. Shah H. Cheng L. Koc J. Harmsen T. Shaked T. Chandra H. Aradhye G. Anderson G. Corrado W. Chai M. Ispir R. Anil Z. Haque L. Hong V. Jain X. Liu. Wide & deep learning for recommender systems. In *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems*, September 2016.

[] Chong Sun, Narasimhan Rampalli, Frank Yang, and AnHai Doan. Chimera: Large-scale classification using machine learning, rules, and crowdsourcing. *Proceedings of the VLDB Endowment*, 7(13):1529–1540, 2014.

[] Mohammad Allahbakhsh, Boualem Benatallah, Aleksandar Ignjatovic, Hamid Reza Motahari-Nezhad, Elisa Bertino, and Schahram Dustdar. Quality control in crowdsourcing systems: Issues and directions. *IEEE Internet Computing*, 17(2):76–81, 2013.

[] Ali Mamdouh Elkahky, Yang Song, and Xiaodong He. A multi-view deep learning approach for cross domain user modeling in recommendation systems. In *Proceedings of the 24th International Conference on World Wide Web*, pages 278–288. International World Wide Web Conferences Steering Committee, 2015.

[] Amirata Ghorbani and James Y. Zou. Data shapley: Equitable valuation of data for machine learning. In *ICML*, 2019.

[] Ivens Portugal, Paulo Alencar, and Donald Cowan. The use of machine learning algorithms in recommender systems: A systematic review. *Expert Systems with Applications*, 97:205–227, 2018.

[] Marco De Gemmis, Leo Iaquinta, Pasquale Lops, Cataldo Musto, Fedelucio Narducci, and Giovanni Semeraro. Preference learning in recommender systems. *Preference Learning*, 41:41–55, 2009.

[] H. Wang N. Wang and D.-Y Yeung. Collaborative deep learning for recommender systems. *Proc. KDD*, pages 1235–1244, 2015.

[] Francois Chollet. *Deep Learning With Python*. Manning, Shelter Island, NY, 2018.

[] Greg Linden, Brent Smith, and Jeremy York. Amazon. com recommendations: Item-to-item collaborative filtering. *IEEE Internet computing*, (1):76–80, 2003.

[] Shumpei Okura, Yukihiro Tagami, Shingo Ono, and Akira Tajima. Embedding-based news recommendation for millions of users. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1933–1942. ACM, 2017.

[] Xuejian Wang, Lantao Yu, Kan Ren, Guanyu Tao, Weinan Zhang, Yong Yu, and Jun Wang. Dynamic attention deep model for article recommendation by learning human editors' demonstration. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 2051–2059. ACM, 2017.

[] Sai Wu, Weichao Ren, Chengchao Yu, Gang Chen, Dongxiang Zhang, and Jingbo Zhu. Personal recommendation using deep recurrent neural networks in netease. In *2016 IEEE 32nd International Conference on Data Engineering (ICDE)*, pages 1218–1229. IEEE, 2016.

[] Toni Kaplan, Susumu Saito, Kotaro Hara, and Jeffrey P Bigham. Striving to earn more: a survey of work strategies and tool use among crowd workers. 2018.

[] Sihem Amer-Yahia, Eric Gaussier, Vincent Leroy, Julien Pilourdault, Ria Borromeo, and Motomichi Toyama. Task composition in crowdsourcing. 2016.

[] A. G. Reference, Magic Man, and Cool Cat. The Title of a GOOD Reference. *The Journal of Referency-Goodness*, pages 1–1000, 2009.