Semi-Unsupervised Lifelong Learning for Sentiment Classification: Less Manual Data Annotation and More Self-Studying*

Xianbin Hong[1,2][0000−0003−1678−0948], Gautam Pal[1,2][0000−0002−2594−9699], Sheng-Uei Guan[1,2][0000−0002−3968−9752], Prudence Wong[2][0000−0001−7935−7245], Dawei Liu[1,2][0000−0001−5807−3884], Ka Lok Man[1,2][0000−0002−5787−4716], and Xin Huang[1,2][0000−0002−1668−9696]

[1] Research Institute of Big Data Analytics, Xi'an Jiaotong-Liverpool University, 111 Ren Ai Road, Suzhou, China
{Xianbin.Hong,Gautam.Pal,Steven.Guan,Dawei.Liu,Ka.Man,Xin.Huang}@xjtlu.edu.cn
https://www.xjtlu.edu.cn/en/research/institutes-centres-and-labs/research-institute-of-big-data-analytics

[2] Department of Computer Science, The University of Liverpool, Ashton Street, Liverpool, UK PWong@liverpool.ac.uk
https://www.liverpool.ac.uk/computer-science

Abstract. Lifelong machine learning is a novel machine learning paradigm which can continually accumulate knowledge during learning. The knowledge extracting and reusing abilities enable the lifelong machine learning to solve the related problems. The traditional approaches like Naïve Bayes and some neural network based approaches only aim to achieve the best performance upon a single task. Unlike them, the lifelong machine learning in this paper focus on how to accumulate knowledge during learning and leverage them for the further tasks. Meanwhile, the demand for labeled data for training also be significantly decreased with the knowledge reusing. This paper suggests that the aim of the lifelong learning is to use less labeled data and computational cost to achieve the performance as well as or even better than the supervised learning.

Keywords: lifelong machine learning · sentiment classification

1 Introduction

Over the past 30 years, machine learning have achieved a significant development. However, we are still in a era of Weak AI rather than Strong AI. Current machine learning algorithms only know how to solve a specific problem but have no idea when they meet some new related problems. Hence, the lifelong machine learning (simply named as lifelong learning or "LML" below) [8] was raised to

* This research is supported by the Research Institute of Big Data Analytics, Xian Jiaotong Liverpool University and the CERNET Innovation Project under Grant NGII20161010.

solve a infinite sequence of related tasks by knowledge accumulation and reusing. For the related problems, an integrated model with knowledge reusing could decrease the cost for the sample annotation.

For instance, in the sentiment classification tasks, we need to predict the sentiment (positive or negative) of a sentence or a document. For different sentiment classification tasks, traditional approaches need to train an independent model on each domain to obtain the best performance. Hence, for each domain we need to collect labeled data for the supervised learning. In this way, the algorithm will never know how to solve a problem without new labeled data. This is what a typical Weak AI.

To achieve the goal of Strong AI, we need to change our learning goal to really understand the sentiment of words. Which means that the algorithm should know how each word influences the sentiment of a document in different tasks. If we can achieve this learning goal, the algorithms are able to solve new tasks without teaching. Zhiyuan Chen and etc. [2] ever proposed a approach to close the goal. They made a big progress but the supervised learning still is needed. Guangyi Lv and etc. [4] extend the work of [2] with a neural network based approach. However, the supervised learning still is necessary under their setting and huge volume of labeled data are required. Hence, this paper aims to decrease the usage of labeled data while maintain the performance.

2 Lifelong Machine Learning

It was firstly called as lifelong machine learning since 1995 by Thrun [9, 7]. Efficient Lifelong Machine Learning (ELLA) [6] raised by Ruvolo and Eaton. Comparing with the multi-task learning [1], ELLA is much more efficient. Zhiyuan and etc. [2] improved the sentiment classification by involving knowledge. The object function was modified with two penalty terms which corresponding with previous tasks.

2.1 Components of LML

The knowledge system contains the following components:

- Knowledge Base (KB): The knowledge Base[2] mainly used to maintain the previous knowledge. Based on the type of knowledge, it could be divided as Past Information Store (PIS), Meta-Knowledge Miner (MKM) and Meta-Knowledge Store (MKS).
- Knowledge Reasoner (KR): The knowledge reasoner is designed to generate new knowledge upon the archived knowledge by logic inference. A strict logic design is required so the most of the LML algorithms lack of the component.
- Knowledge-Base Learner (KBL): The Knowledge-Based Learner[2] aims to retrieve and transfer previous knowledge to the current task. Hence, it contains two parts: task knowledge miner and leaner. The miner seeks and determines which knowledge could be reused, and the learner transfers such knowledge to the current task.

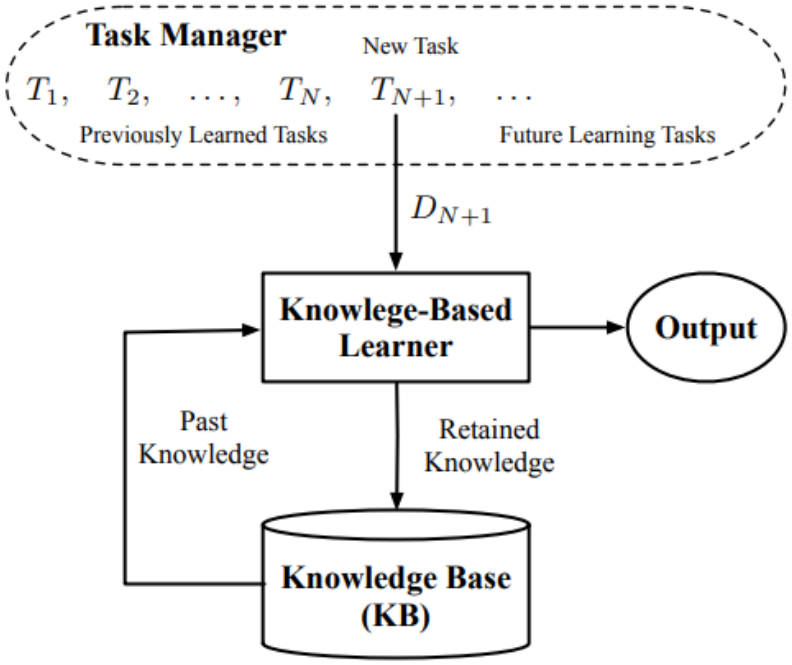

Fig. 1. Knowledge System in the Lifelong Machine Learning [2]

2.2 Sentiment Classification

Hong and etc.[3] had discussed that the NLP field is most suitable for the lifelong machine learning researches due to its knowledge is easy to extract and to be understood by human. Previous classical paper[2] chose the sentiment classification as the learning target because it could be regarded as a large task as well as a group of related sub-tasks in the different domains. Although these sub-tasks are related to each other but a model only trained on a single sub-tasks is unable to perform well in the rest sub-tasks. This requires the algorithms could know when the knowledge can be used and when can not due to the distribution of each sub-tasks is different. Known these, an algorithm can be called as "lifelong" because it is able to transfer previous knowledge to new tasks to improve performance.

Although deep learning already is applied in sentiment classification, it still could not leverage past knowledge well. This because the complexity of neural network limits the researches to define and extract knowledge from the data. As the previous work[2], this paper also uses Naïve Bayes as the knowledge can be

presented by the probability. In this way, we need to know the probability of each word that shows in the positive or negative content. We also need to know well that some words may only have sentiment polarity in some specific domains(equal to tasks in this paper). "Lifelong Sentiment Classification" ("LSC" for simple below) [2] records that which domain does a word have the sentiment orientation. If a word always has sentiment polarity or has significant polarity in current domain, a higher weight will sign to it more than other words. This approach contains a knowledge transfer operation and a knowledge validation operation.

3 Contribution of This Paper

Although LSC[2] already raised a lifelong approach, it only aims to improve the classification accuracy. It still is under the setting of the supervised learning and also is unable to deliver an explicit knowledge to guild further learning.

Based on the LSC, this paper advances the lifelong learning in sentiment classification and have two main contributions:

- **A improved lifelong learning paradigm is proposed to solve the sentiment classification problem under unsupervised learning setting with previous knowledge.**
- **We introduce a novel approach to discover and store the words with sentiment polarity for reuse.**

4 Sentiment Polarity Words

4.1 Naïve Bayesian Text Classification

In this paper, we define a word has sentiment polarity by calculating the probability that it appears in a positive or negative content (sentence or document). If a word has a high probability with sentiment polarity, it also will leads to the document have higher probability of sentiment probability based on the Naïve Bayesian (NB) formula. Hence, to determine the words with polarity is the key to predict the sentiment.

Naïve Bayesian (NB) classifier [5] calculates the probability of each word w in a document d and then to predict the sentiment polarity (positive or negative). We use the same formula below as in the LSC[2]. $P(w|c_j)$ is the probability of a word appears in a class:

$$P(w|c_j) = \frac{\lambda + N_{c_j,w}}{\lambda |V| + \sum_{v=1}^{V} N_{c_j,v}} \tag{1}$$

Where c_j is either positive (+) or negative (-) sentiment polarity. $N_{c_j,w}$ is the frequency of a word w in documents of class c_j . $|V|$ is the size of vocabulary V and $\lambda(0 \leq \lambda \leq 1)$ is used for smoothing (set as 1 for Laplace smoothing in this paper).

Given a document, we can calculate the probability of it for different classes by:

$$P(c_j|d_i) = \frac{P(c_j)\prod_{w\in d_i} P(w|c_j)^{n_w,d_i}}{\sum_{r=1}^{C} P(c_r)\prod_{w\in d_i} P(w|c_r)^{n_w,d_i}} \tag{2}$$

Where d_i is the given document, n_w, d_i is the frequence of a word appears in this document.

To predict the class of a document, we only need to calculate $P(c_+|d_i) - P(c_-|d_i)$. If the difference is lager than 0, the document should be predict as positive polarity:

$$\begin{aligned} P(c_+|d_i) - P(c_+|d_i) = \frac{P(c_+)\prod_{w\in d_i} P(w|c_+)^{n_w,d_i}}{\sum_{r=1}^{C} P(c_r)\prod_{w\in d_i} P(w|c_r)^{n_w,d_i}} - \\ \frac{P(c_-)\prod_{w\in d_i} P(w|c_-)^{n_w,d_i}}{\sum_{r=1}^{C} P(c_r)\prod_{w\in d_i} P(w|c_r)^{n_w,d_i}} \end{aligned} \tag{3}$$

As we only need to know whether $P(c_+|d_i) - P(c_-|d_i)$ is lager that 0, so the formula could be simplify to:

$$\begin{aligned} P(c_+|d_i) - P(c_+|d_i) = P(c_+)\prod_{w\in d_i} P(w|c_+)^{n_w,d_i} - \\ P(c_-)\prod_{w\in d_i} P(w|c_-)^{n_w,d_i} \end{aligned} \tag{4}$$

4.2 Discover Words with Sentiment Polarity

Ideally, if we know the $P(c_+)$, $P(c_-)$ and $P(w|c_j)$ of all words, we can predict the sentiment polarity for all documents. However, above three key components are different in different domains. LSC [2] proposed a possible solution to calculate $P(w|c_j)$, but it uses all words which has high risk to be overfitting. As we known, not all words have sentimental polarity like "a", "one" and etc. while some words always have polarity like "good", "hate", "excellent" and so on. In addition, some words only have sentiment polarity in specific domains. For example, "tough" in reviews of the diamond indicates that the diamond have a good quality while it means hard to chew in the domain of food. Hence, in order to achieve the goal of the lifelong learning. We need to find the words always have sentiment polarity and be careful for those words only shows polarity in specific domains.

5 Lifelong Semi-supervised Learning for Sentiment Classification

Although LSC [2] considered the difference among domains, it still is a typical supervised learning approach.In this paper, we proposed to learn as two stages:

1. Initial Learning Stage: to explore a basic set of sentiment words. After that, the model should be able to basically classify a new domain with a good performance.
2. Self-study Stage: Use the knowledge accumulated from the initial stage to handle new domains, also fine-tune and consolidate the knowledge generated from the initial learning stage.

5.1 Initial Learning Stage

In this stage, we need to train the model to remember some sentiment polarity words. This requires us to find the words with sentiment polarity in each domain. We need to answer two questions here:

1. How to determine the polarity of a word?
2. How much domains do we need for the initial learning stage?

For the first question, we need to find which words mainly show in the positive or negative documents. This means for a word w with positive polarity, $P(+|w) >> P(-|w)$ or $P(+|w) >> P(+)$. In this paper, we will use $O(w) = P(+|w)/P(+)$ to represent the polarity. This because that the $P(c_j|w)/P(w)$ is easy to extend into the multi-classes classification problems. According to the Bayesian formula, $P(+|w)/P(+) = P(w|+)/P(w)$.

5.2 Self-study Stage

In this stage, our main task is to explore which words have polarity. We will mainly use these words to predict the new domains and assign the pseudo-labels to them. With the pseudo labels, we are able to discover the new words with polarity. Following is the the procedure for self-study:

1. Using the sentiment words accumulated from the previous tasks to predict a new domain, then assign the prediction results as the pseudo labels.
2. Using the reviews and pseudo labels of above new domain as new training data to run Naïve model.
3. Update the sentiment words knowledge base.

6 Experiment

6.1 Datasets

In the experiment, we use the same datasets as LSC [2] used. It contains the reviews from 20 domains crawled from the Amazon.com and each domain has 1,000 reviews (the distribution of positive and negative reviews is imbalanced).

6.2 Word Polarity Analysis

To answer the first question for the initial learning stage, we need to know which words exactly influence the sentiment classification. Firstly, we calculate $P(w|+)$ and $P(w|-)$ for each words. Then, we define the polarity degree by $O(w) = P(w|+)/P(w)$. Finally, we only choose a specific percentage words to predict and see whether the performance decreases. In addition, we also only consider the words that at least show over average 5 times in per domain. This because that we did not delete the symbols and numbers in the data, and these characters may be noise in the training data.

We firstly sorted the words or symbols (no data pre-processing to the corpus in this paper) by the polarity $O(w)$ and then choose a specific percentage words or symbols from the whole words to only 10%. From Table 1 we can see that using no less than 30% can obtains the best average result. This means that the most of words and symbols do not have obvious sentiment orientation.

Hence, we will only keep 30% of words for Naïve Bayes model and even get better f1 score. Although the performance decrease on a single domain, the better global performance can achieve with only the sentiment words.

Table 1. F1 Score of Naïve Bayesian Classifiers under Decreasing Word Usage Percentage

F1 Percentage / Datasets	100%	80%	60%	50%	40%	30%	20%	10%
AlarmClock	0.8082	0.8082	0.8082	0.8082	0.8082	**0.8082**	0.274	0.2333
Baby	0.6564	0.6564	0.6564	0.6564	0.6564	**0.6564**	0.1759	0.1408
Bag	0.6811	0.6811	0.6811	0.6811	0.6811	**0.6811**	0.3559	0.1056
CableModem	0.6064	0.6064	0.6064	0.6064	0.6064	**0.6064**	0.2195	0.1105
Dumbbell	0.6346	0.6346	0.6346	0.6346	**0.6346**	0.6602	0.1589	0.1383
Flashlight	0.5876	0.5876	0.5876	0.5876	0.5876	**0.5921**	0.3278	0.1036
Gloves	0.6131	0.6131	0.6131	0.6131	0.6131	**0.6131**	0.3205	0.1206
GPS	0.6814	0.6814	0.6814	0.6814	0.6814	**0.6814**	0.2838	0.1629
GraphicsCard	0.5775	0.5775	0.5775	0.5775	0.5775	**0.5775**	0.2776	0.1271
Headphone	0.6578	0.6578	0.6578	0.6578	0.6578	**0.6578**	0.268	0.1745
HomeTheaterSystem	0.8394	0.8394	0.8394	0.8394	0.8394	**0.8394**	0.2404	0.2238
Jewelry	0.604	0.604	0.604	0.604	0.604	**0.604**	0.3371	0.1088
Keyboard	0.653	0.653	0.653	0.653	0.653	**0.653**	0.2117	0.1841
MagazineSubscriptions	0.8042	0.8042	0.8042	0.8042	0.8042	0.8042	**0.8049**	0.2115
MoviesTV	0.5843	0.5843	0.5843	0.5843	0.5843	0.5843	**0.606**	0.0976
Projector	0.7387	0.7387	0.7387	0.7387	0.7387	**0.7387**	0.1814	0.168
RiceCooker	0.7656	0.7656	0.7656	0.7656	**0.7656**	0.7739	0.1683	0.1566
Sandal	0.5987	0.5987	0.5987	0.5987	0.5987	**0.5987**	0.3501	0.1077
Vacuum	0.7362	0.7362	0.7362	0.7362	0.7362	**0.7362**	0.2155	0.1807
VideoGames	0.6835	0.6835	0.6835	0.6835	0.6835	**0.6835**	0.4514	0.173
Average	0.6756	0.6756	0.6756	0.6756	0.6756	**0.6775**	0.3114	0.1514

6.3 Requirement for the Initial Learning

For the second question of the initial learning stage, the answer depends on the tasks. In the practice, all of the labeled data definitely need to be used for training. The only question should be conceded is that how much labeled data can meet the minimum requirement. For this sentiment classification task, one domain is absolutely insufficient. Based on the experiment result, the initial learning stage at least needs two domains, and can achieve much better performance with three domains. Increase more domains will not significant influence the performance. Hence, three domains are enough for this task. For different tasks, two labeled domains are necessary. More labeled domains are suggested to continue collect until the performance on the new domains tends to steady.

6.4 Self-study Learning

In the self-study learning stage, the learning is designed under the unsupervised learning setting. In this stage, there is any labeled data. Instead of that, we uses the model generate from the initial learning stage to predict each domain and assign the pseudo labels to them. After that, the model will regard the pseudo labels as the real labels and continue the training on the new domain. With this method, self-study learning stage can learn new domains well without any labeled data.

Table 2. F1 Score for NB-S, NB-T, SU-LML

F1 Score \ Model / Datasets	NB-S	NB-T	SU-LML
CableModem	0.4774	0.6633	**0.8694**
Dumbbell	0.6539	0.764	**0.8748**
Flashlight	0.6536	0.6251	**0.8259**
Gloves	0.5973	0.6943	**0.785**
GPS	0.6447	0.7465	**0.9121**
GraphicsCard	0.4797	0.7346	**0.8768**
Headphone	0.5938	0.7356	**0.8858**
HomeTheaterSystem	0.6242	0.8611	**0.9236**
Jewelry	0.6927	0.7088	**0.7599**
Keyboard	0.6905	0.7289	**0.8707**
MagazineSubscriptions	0.6284	0.8056	**0.8932**
MoviesTV	0.4991	0.6785	**0.8381**
Projector	0.6565	0.7525	**0.8575**
RiceCooker	0.6833	0.8027	**0.8475**
Sandal	0.6972	0.6904	**0.8059**
Vacuum	0.7728	0.8	**0.8992**
VideoGames	0.5665	0.7564	**0.9068**
Average	0.6242	0.7381	**0.8607**

Table 2 is the F1 score of three models on 17 domains. The first three domains was used for the initial learning stage. And we use the Macro-F1 score because the datasets are imbalanced and it can prove our performance on the minor classes. We compared our model (Semi-Unsupervised Learning, SU-LML for short) with Naïve Bayes model which only trained on the first three (source) domains (NB-S) and Naïve Bayes model trained on each domain with labels by 5-fold cross validation (NB-T). We can see that our approach is significantly better than other two approaches. It even perform better than the NB-T, a typically supervised learning. The figure 2 shows the result more clearly. The comparisons to LSC and neural based lifelong learning [4] are not going to show here, because firstly their codes are still unavailable and secondly their approaches are totally supervised learning.

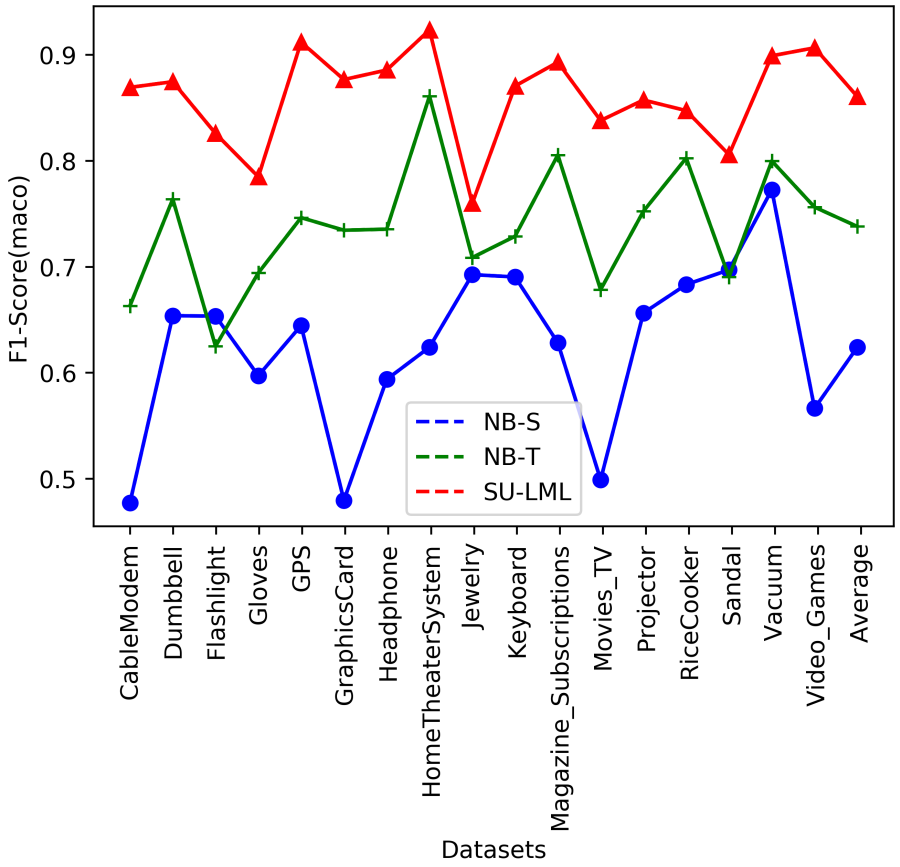

Fig. 2. F1 Score in Self-Study Stage

6.5 Knowledge Generated during Learning

In this paper, we done one more important things is that we discovered which words have sentiment polarity. If a word was regarded with sentiment polarity, we increase the polarity score of it with one. In addition, we will plus an additional score from 0 to 1 to 1 based on the $O(w)$ rank. From table 3, we can see that most top words with negative emotion and most of them make sense.

Table 3. Top 20 Words with Negative Sentiment

Word	Degree for Negative Sentiment
refund	32.99921813917123
garbage	32.994266353922335
junk	32.985405264529575
waste	32.984102163148286
worst	32.97185301016418
rma	32.96846494657285
poorly	32.96194943966641
terrible	32.95569455303623
disappointed	32.949960906958566
trash	32.948918425853535
useless	32.94683346364347
worthless	32.94057857701329
awful	32.92520198071411
defective	32.917904612978894
return	32.913734688558776
exchange	32.908001042481104
respond	32.90487359916601
poor	32.90409173833724
disappointment	32.90278863695596
crap	32.89653375032577

7 Discussion

7.1 Choice of Initial Domains

In this research, only the initial domains use labeled data. Hence, the choice of the initial domains is the foundation of the algorithm. In the practice, there are two scenarios for the choice of the initial domains:

1. Only the labels of a few of domain are available
2. Human resource is limited and only could annotation a few of selected domains

In the first scenario, it is impossible to choose the initial domains. Hence, choice of initial domains is not a concern. As for the second scenario, the choice

of the domains is available. Hence, all valuable human resource should be use to annotate the most valuable domains and a priority of the domains is needed. However, if the priority of the domains is unavailable while the data annotation, the choice becomes blind.

Ideally, the best choice will maximize the average performance over all the tasks. However, under the lifelong learning setting, the future domains are unknown and so such a best choice is hard to determine. In this case, a more reasonable and flexible solution is to minimize the influence which caused by the change of the initial domains.

To best of the authors' knowledge, to increase the number of the domains within the self-study stage could significantly decrease the influence from the initial domains. Therefore, although the choice of the initial domains is very important, the authors suggest to improve the robustness by proposing better self-study approaches.

7.2 Learning Order

As for the learning order, only the self-study stage needs to be discussed. This because that the order of the initial domains is unable to influence the performance. As the number of domains is increasing under the lifelong learning setting, the most economical approach is learning as the order of task arriving. In the previous works[2], the order should be changed to obtain the best performance on the different task. However, the time complexity of this kind of approach is $O(n^2)$. It is easy to image that this approach is infeasible when much more domains arriving.

Considering with the big data scenario, lifelong learning must follow the design of the online learning. This means that the algorithm will only fine-tine itself when the new domain arriving rather than retrain with all data again. In addition, not only new tasks need to be focus, the previous tasks also should be paid by more attention. In other words, the learning of the new tasks should also improves the performance of the previous.

In summary, although the learning order could influence of a single task, it is not worth to change order for each task. The most important point is how to use new learning to improve old tasks.

8 Conclusion and Outlook

We proposed a semi-unsupervised lifelong sentiment classification approach in this paper. It can accumulate knowledge from the previous learning and turn to self-study. A very few labeled data required in our approach so it is very suitable for the industry scenario. The performance of it even exceeds the supervised learning, which shows that the knowledge reusing of the lifelong learning is useful.

Although we only show two classes classification here, but the ideal is also suitable for the multi-classes classification. All text classification can use this approach, not only sentiment classification. Our model classify documents by the

knowledge of the sentiment polarity of the words, which uses the same approach of we human being. We shows that to focus the goal behind the learning tasks is more meaningful than just to find a solution. Understanding the words is much important than solve a sentiment classification task. We should learn the knowledge and skills for all tasks rather than a solution for a single task.

We must indicate that the choice of the initial learning domains will significantly influence the performance. SU-LML also hasn't answer the question how to validate the knowledge from previous domains. In order to validate the knowledge and measure the task similarity, semi-supervised learning is needed. Like the learning of human, not all of data need to be labeled, but some of them will help to monitor the learning process. Without knowledge validation and learning monitoring, $\mathrm{SU}_L MLstilllacksofrobustness$.

In the further researches, the robustness of lifelong learning system has the highest priority. The knowledge validation and learning monitoring system will be included to make sure the learning keep better.

References

1. Caruana, R.: Multitask learning. Machine learning **28**(1), 41–75 (1997)
2. Chen, Z., Ma, N., Liu, B.: Lifelong learning for sentiment classification. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers). vol. 2, pp. 750–756 (2015)
3. Hong, X., Wong, P., Liu, D., Guan, S.U., Man, K.L., Huang, X.: Lifelong machine learning: Outlook and direction. In: Proceedings of the 2nd International Conference on Big Data Research. pp. 76–79. ACM (2018)
4. Lv, G., Wang, S., Liu, B., Chen, E., Zhang, K.: Sentiment classification by leveraging the shared knowledge from a sequence of domains. In: International Conference on Database Systems for Advanced Applications. pp. 795–811. Springer (2019)
5. McCallum, A., Nigam, K.: Text classification by bootstrapping with keywords, em and shrinkage. Unsupervised Learning in Natural Language Processing (1999)
6. Ruvolo, P., Eaton, E.: Ella: An efficient lifelong learning algorithm. In: International Conference on Machine Learning. pp. 507–515 (2013)
7. Thrun, S.: Is learning the n-th thing any easier than learning the first? In: Advances in neural information processing systems. pp. 640–646 (1996)
8. Thrun, S.: Lifelong learning algorithms. In: Learning to learn, pp. 181–209. Springer (1998)
9. Thrun, S., Mitchell, T.M.: Lifelong robot learning. Robotics and autonomous systems **15**(1-2), 25–46 (1995)