applied sciences

MDPI

Article

Dual-Track Lifelong Machine Learning-Based Fine-Grained Product Quality Analysis

Xianbin Hong [1,2], Sheng-Uei Guan [1,2,*], Nian Xue [3,4], Zhen Li [5,6,*], Ka Lok Man [1,2], Prudence W. H. Wong [2] and Dawei Liu [7,*]

1 Department of Computing, School of Advanced Technology, Xi'an Jiaotong-Liverpool University, Suzhou 215123, China
2 Department of Computer Science, The University of Liverpool, Liverpool L69 3BX, UK
3 Department of CSE, Tandon School of Engineering, New York University, Brooklyn, NY 11201, USA
4 Division of Engineering, New York University Abu Dhabi, Saadiyat Island, Abu Dhabi P.O. Box 129188, United Arab Emirates
5 Department of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China
6 Shanghai Grandhonor Information Technology Co., Ltd., Shanghai 200333, China
7 Data Science Research Center, Duke Kunshan University, Kunshan 215316, China
* Correspondence: steven.guan@xjtlu.edu.cn (S.-U.G.); lizh0019@gmail.com (Z.L.); dawei.liu@dukekunshan.edu.cn (D.L.)

Abstract: Artificial intelligence (AI) systems are becoming wiser, even surpassing human performances in some fields, such as image classification, chess, and Go. However, most high-performance AI systems, such as deep learning models, are black boxes (i.e., only system inputs and outputs are visible, but the internal mechanisms are unknown) and, thus, are notably challenging to understand. Thereby a system with better explainability is needed to help humans understand AI. This paper proposes a dual-track AI approach that uses reinforcement learning to supplement fine-grained deep learning-based sentiment classification. Through lifelong machine learning, the dual-track approach can gradually become wiser and realize high performance (while keeping outstanding explainability). The extensive experimental results show that the proposed dual-track approach can provide reasonable fine-grained sentiment analyses to product reviews and remarkably achieve a 133% promotion of the Macro-F1 score on the Twitter sentiment classification task and a 27.12% promotion of the Macro-F1 score on an Amazon iPhone 11 sentiment classification task, respectively.

Keywords: lifelong machine learning; fine-grained sentiment classification; reinforcement learning; expert system; knowledge graph

Citation: Hong, X.; Guan, S.-U.; Xue, N.; Li, Z.; Man, K.L.; Wong, P.W.H.; Liu, D. Dual-Track Lifelong Machine Learning-Based Fine-Grained Product Quality Analysis. *Appl. Sci.* **2023**, *13*, 1241. https://doi.org/10.3390/app13031241

Academic Editor: Vincent A. Cicirello

Received: 10 December 2022
Revised: 8 January 2023
Accepted: 9 January 2023
Published: 17 January 2023

1. Introduction

Artificial intelligence (AI) is more intelligent than humans in a few fields, such as image recognition competitions, chess, Go, and intellectual television questions and answers [1–6]. Furthermore, lifelong machine learning (LML) [7–12] aims to make AI more effective [13–15]. Under lifelong machine learning, AI is expected to become stronger and reach superhuman levels. While the algorithms become more effective, humans also want them to be more explainable. When AI was weak, scientists only used it to replace people for straightforward labor-intensive work. At that time, humans took the roles of teachers and only wondered whether the results were correct. In the future, when AI can develop its algorithms and gain higher performances in specific tasks, people will wonder how AI solves these tasks. At that moment, the interpretability of AI is essential because humans need to learn from AI. Hence, a high performance is not the single critical point one needs to focus on in lifelong learning. We also need to improve the interpretability of lifelong learning algorithms.

However, research into explainable AI [16] is insufficient for current intelligence algorithms. After entering the era of deep learning [17], model interpretability is essential to

developing deep neural network models. Although deep learning models are overwhelming, humans cannot directly learn from them to enhance human society. Thus, the authors suggest scientists improve the interpretability of deep learning. We should also develop other more explainable algorithms at the same time. For example, current deep learning algorithms can precisely classify the sentiment of a product review [18,19]. To optimize the performance, deep neural networks were introduced to increase sentiment classification accuracy [9,20,21], and an ensemble application (involving symbolic and subsymbolic AI for sentiment analyses) was proposed in [22]. These machine learning methods produce reasonable sentiment classification results. Consequently, some downstream applications have been developed based on sentiment analysis, e.g., review-aware recommendations [23] and peer review-based citation count prediction [24]. Nevertheless, the above-mentioned sentiment classification methods cannot explain why a customer likes or dislikes a product. Compared with a simple sentiment classification score, the reasons behind the score are more critical [25,26].

Deep learning and other black-box approaches have weak interpretability because they involve knowledge that humans cannot understand. If we want to make them fully explainable, the algorithms should only use the knowledge of humans and change them to be white-box approaches. This way can give algorithms high interpretability, but it also limits the development of AI. When people cannot solve tasks, we still wish to solve them via a black-box approach. Although interpretability is essential, giving up those high-performance black-box algorithms is not a reasonable solution. Therefore, the authors suggest using black-box approaches for tasks that human intelligence is insufficient to solve. In the meantime, we also need to collect knowledge and solve those tasks by white-box approaches to develop explainable AI. To be noticed, black-box approaches are also tools of knowledge mining since their learning outcomes are also sources of explainable AI in lifelong learning.

Besides deep learning, reinforcement learning (RL) is another popular area of machine learning concerned with how an intelligent agent should take action in an environment to maximize the cumulative reward. Reinforcement learning aims to learn the best action for each state in an environment. Different from supervised learning, RL does not need labeled input/output pairs to be presented. Given a group of possible states (state space) S, reinforcement learning will choose an action from available actions (action space) A. The choices $A = \pi(S)$ form a strategy that simulates human decisions for different situations. When all of the states and actions are defined by humans or those that are comprehensible for humans, reinforcement learning is a white-box approach (where system inputs, outputs, and internal processing are transparent). The state space and action space are known in a close environment, such as chess. The environment is typically shown as a Markov decision process (MDP). However, in an open environment, the state and action space always change over time. Therefore, it is impossible for scientists to define all of the states and actions well. In this situation, we can only define states and actions based on current knowledge and leave all unknown situations to a black-box algorithm. This is why we propose a dual-track approach.

Hence, this work proposes a dual-track principle to develop lifelong learning and make it more powerful and explainable at the same time. This work uses fine-grained sentiment analysis [25–27] as a scenario to demonstrate the integration of both white-box and black-box approaches in lifelong learning on dual tracks. In this approach, the authors use reinforcement learning to collect and evaluate knowledge in a white-box way and deep learning to supplement when white-box approaches lack the necessary knowledge.

Throughout the rest of this paper, Section 2 will further introduce the works of explainable AI. Then Section 3 will briefly explain how the dual-track system works for fine-grained sentiment analysis. A reinforcement learning-based approach will be first introduced in Section 4. In addition to learning the sentiment of a product review, reinforcement learning has another function involving knowledge validation by entropy. Section 5 will introduce the knowledge management system in the dual-track system.

Section 6.1 will introduce the datasets collected for this work. The experimental results are available in Section 6. Deep learning, expert systems, and reinforcement learning provide sentiment analyses from different aspects. The experiment results show that the dual-track can handle the sentiment analysis task well and achieve better sentiment classification performance than a deep learning algorithm. Finally, Section 7 will summarize and suggest a knowledge-sharing standard for lifelong learning.

In this work, we provide the following main contributions to the AI community:

1. We propose applying reinforcement learning to learn the sentiment of a product review and introduce the concept of the "environment" into sentiment analysis to enable reinforcement learning. To the best of our knowledge, this is the first time reinforcement learning is involved in product review systems.
2. We propose generating knowledge and evaluating model correctness via reinforcement learning and entropy; we achieved remarkable enhancements on the sentiment classification performances of product reviews.
3. We propose using explicit knowledge to assist and monitor deep learning to implement a generic purpose sentiment analysis architecture; we utilized explicit knowledge in lifelong sentiment analysis and in improving the interpretability of the product review system.
4. We created two datasets for lifelong sentiment analysis, including 12,740 product reviews from Amazon and over 15,000 product reviews from Twitter.

2. Explainable Artificial Intelligence

Humans usually act as teachers to AI, but AI can also teach humans. AlphaGo [6] is an example. After AplphaGo defeated Lee Sedol, many players worried that AI would rule Go and force human players to give up. On the contrary, human players were inspired and learned much from it. AlphaGo deeply leverages reinforcement learning to learn match strategies, which provide precise observations of the learning process. Reinforcement learning learned and demonstrated many new strategies in Go, which gave human players a new understanding of matches. Humans created AlphaGo, but it also taught humans a lot. AI is not only a competitor, it is also a collaborator.

Human players can observe AlphaGo's behaviors to understand its strategies, which work but are inefficient. When humans only use AlphaGo to play games, understanding each action is unimportant. Nevertheless, in finance or even medical treatment, any mistake can threaten the properties and lives of humans. Thus, it is necessary to understand each AI activity. In other words, AI should be explainable.

Explainable AI [16,28,29] involves AI where every calculation and output is explainable. "Explainable" means that scientists can fully understand how each calculation of an algorithm leads to the final results and they can assess whether the calculations and outputs are correct. The decision process of explainable AI is intelligible, so it is possible to evaluate whether such an AI system is reliable and makes people more confident when determining whether to use it. Meanwhile, after adopting this explainable system, people can learn how to solve problems and gain new knowledge from it. Explainable AI was raised with expert systems (ES) [30]. In the early era of AI development, people only had a minimal understanding of AI and did not believe AI could provide a correct solution. Thus, it is natural for managers to doubt AI and want a reasonable explanation. Scientists then used expert systems, especially the decision tree, to explain their algorithms to solve this problem. Factual explanations attempt to dive into the black box. Previous research studies [31,32] still relied on decision trees. Although there are works [33,34] about deep learning, the outcomes are still in the early stages. Argument mining [35] is also a hot topic in explainable AI and NLP. Researchers [36–38] have attempted to automatically extract and identify the argumentative structures from text with the aid of computer programs. It is helpful for scenarios where facts support the decisions.

2.1. Expert Systems

The AI community in the mid-1960s developed expert systems (ES) [30]. The basic idea of ES is to leverage expertise from humans to solve problems. Thus, they are also called consultation systems. ES has many forms, such as rule-based, knowledge-based, case-based, etc.

Expert systems were popular in the last century. However, people quickly found that ES could only solve a few problems and their enthusiasm faded. Although people do care about the explainability of AI, it should have the ability to solve problems first. Thus, scientists developed other machine learning models, such as SVM [39], Bayesian networks [40], and deep neural networks [17].

2.2. Deep Learning

After the deep neural network (also called deep learning) [17] became popular in various areas, such as machine translation [2,41], chatbot [42], unmanned aerial vehicle [43], person re-identification [44,45], multiple object tracking [46], image recognition [47,48] and signal processing [49], explainable AI received attention again. This time, people saw the power of deep learning and never doubted its ability. However, this does not mean AI will not make mistakes. Thus, people still want to know how AI decides to avoid mistakes. For instance, when doctors use AI to generate a diagnosis or a treatment plan, they wonder why AI gives such results. If the treatment fails, doctors are responsible for the consequences rather than AI.

However, it is too difficult to explain what is behind the deep neural network by current technologies. Thus, scientists mainly attempt to visualize the network parameters. For example, BERT (bidirectional encoder representations from transformers) is based on the "attention" mechanism, so scientists visualize its attention connection to help with understanding. However, it is still far from expectations.

In sentiment classification, transformer-based approaches achieve state-of-the-art performances, so scientists wonder why they are successful and visualize it. Figure 1 shows how attention works in product sentiment classification. In this example, a consumer wrote two sentences: "The screen of this phone is great. It is (Its) battery life is terrible". In the visualization, if the transformer thinks two words have a connection, there will be a line between them. This situation means the transformer pays "attention" to another word. For example, the transformer thinks "battery life" when it reads "phone" and thinks "screen" and "phone" when it reads "battery". Thus, we know that the transform thinks that there is a connection between the phone, battery, and screen. Visualization can help people better understand attention, but attention only plays a minor role in the transformer. Humans still do not understand how the transformer analyzes the sentiment of a review. Deep learning in image processing also faces the same problem. An example is shown in Figure A2 in Appendix A.3.

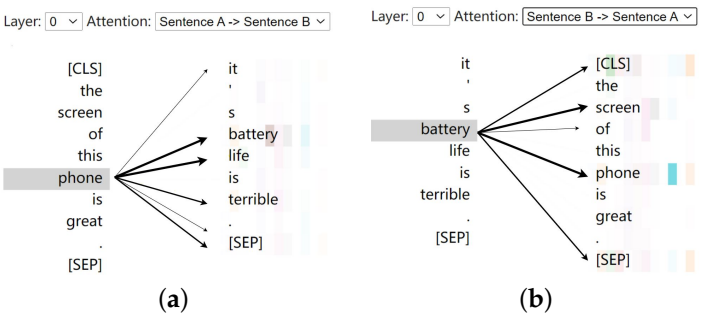

Figure 1. Visualization of BERT Attention. (**a**) Sentence A to B. (**b**) Sentence B to A.

2.3. Reinforcement Learning

Reinforcement learning (RL) is also a powerful tool in AI systems. The interpretability of RL depends on its internal design. RL can use the Markov chain inside or adopt deep

learning. Hence, it is necessary to choose an intelligible internal algorithm when designing an explainable AI.

Currently, expert systems are good at explainability but are weak at performance, contrary to deep neural networks with good performances and lousy explainability. They are not perfect, so we still need a solution that has high performance and good interpretability. Reinforcement learning also has good performance and interpretability with appropriate designs. The effective white-box algorithm is what we desire, but such a system requires creating an extensive knowledge base. Although this work proposes using reinforcement learning as its core component, it is necessary to mention that this process is estimated to take a very long time to converge. Before it gains enough knowledge, we also use deep learning as a supplement. In summary, this method suggests gradually improving the white-box approach but keeping the black-box approach as a tool, which is a dual-track system.

3. Fine-Grained Sentiment Analysis and Dual-Track System

As mentioned in Section 2, scientists have not opened the black box of deep learning, and white-box approaches cannot replace deep learning. Thus, a trade-off is to use a dual-track system, which means we should use the white-box approach as much as possible, but still keep deep learning as a supplement. The following part of this paper will explain this principle in detail.

This work uses fine-grained product sentiment classification as an example to demonstrate such a dual-track system. Although deep learning performed well in sentiment classification, it can only give customers a simple score for each review. However, customers want details from the reviews, which requires a fine-grained level. This level is challenging for deep learning, so other approaches, such as reinforcement learning and expert systems, are needed.

3.1. Fine-Grained Product Sentiment Classification

Before starting the design of lifelong architecture and algorithms, it is necessary to clarify the demand for fine-grained sentiment analysis. From the customer side, they want to know whether the product satisfies their demands. Different customers have different demands for a product, so a rating score from one customer may be meaningless to another. Thus, they want to obtain evaluations from different aspects. The seller also wants to know which problems the product has and how to improve them. Sentiment classification can only provide a sentiment score for the whole product, which is insufficient for customers to purchase. So fine-grained sentiment analysis is still necessary.

When customers are shopping online, they need to read many reviews to evaluate the product quality, even if they already know the rating of the product (Figure A3 in Appendix B.1 is a product review example of renewed iPhone XR). This case shows that the rating score cannot provide enough information for purchase. If the website can summarize the product from different aspects, this will be helpful.

If we can show customers the rating of each feature, it would be more helpful. Table 1 demonstrates what a fine-grained sentiment classification looks like, such as for a product review in Figure A3 in Appendix B.1. Amazon only provides a total mark for a product, but fine-grained classification can show consumers in detail. With fine-grained analysis, customers can decide whether to purchase the product based on their demands.

The process of fine-grained sentiment has the following three steps. First, it requires an algorithm to recognize all product features and their values in a review, where values are either discrete or continuous. Second, the algorithm needs to determine the sentiment of each feature. Finally, knowing how each feature contributes to the overall sentiment score, the algorithm can calculate a total score based on the sentiment of each feature.

Three steps of fine-grained sentiment classification:

1 Recognize each feature in the review; the named-entity recognition task needs a list of features.

2 Determine the sentiment of each feature and the sentiment classification task.
3 Calculate the overall sentiment based on each feature.

Table 1. Fine-grained sentiment classification example.

Sentence	Feature	Score
I was really scared about obtaining a damaged product, I read the reviews every day until it came . . .	Not related to product	None
shipping was **fast** and the package came 1 day early. . .	Shipping	5
battery was at **84%** I was hoping 90 or higher but that is alright. . .	Battery life	4
for me.	Not related to product	None
I did have some **scuffs** at the top of the phone tho and near the charging port.	Appearance damage	3
the what bands on the Side also look a little **dingy**.	Appearance damage	4
My **seller** was CHUBBIESTECH. . .	Seller	None
and **they** gave me a free screen protector and case.	Seller	5
	Total	4

The first step from the above-mentioned steps is a named-entity recognition [50] (NER) task. This step requires a list of features and all of their values. The second step is a sentiment classification task. This step needs to learn the corresponding sentiment score of each value of features (to which deep learning is applicable). The final step needs a product structure and each feature's significance. This knowledge can be provided by a knowledge graph. It is easy to find the list in the first step, and the second step can use deep learning and implicit knowledge, followed by the third step, where explicit knowledge (in knowledge graphs) can be understood by humans. Thus, this work uses both implicit and explicit knowledge. Furthermore, this work will try to use reinforcement learning to mine explicit knowledge from implicit knowledge in the second step. Deep learning is still in charge of sentiment classification before reinforcement learning obtains enough explicit knowledge to conduct the second step independently. This work aims to only use explicit knowledge to solve the sentiment analysis task, but it needs time to collect explicit knowledge. Prior to that, implicit knowledge has a role in lifelong learning architecture. Thus, this work uses a dual-track approach, and both implicit and explicit knowledge are used.

3.2. Named-Entity Recognition for Lifelong Learning

Section 3.1 mentioned that the first step of fine-grained sentiment analysis is feature recognition. This detection needs NER, which is a common task in NLP and has many existing approaches.

In this work, a context-based approach is proposed for NER. This design is inspired by the concept of the "environment" of reinforcement learning. In NLP, the context of a word is its environment. When a word is mentioned, the NER algorithm should check the environment before giving a result. The environment can be set by humans or detected by AI. In most of the tasks, the background is known to determine the environment before NER. For instance, in the sentiment classification of an iPhone product review, the environment can be set as an electronic device, so "Apple" in this environment must be a company rather than a fruit. If the environment is unknown, AI can also read the context and determine it by itself. When it reads "battery", "screen", etc., it can also know the environment is irrelevant to fruit.

An environment contains plenty of entities, a knowledge graph (Figure A5 in Appendix B.2 provides an example) can provide information on the entities. Then when the machine is learning, it knows how to observe which entities. The entities in this knowledge graph also describe a close environment. The word "Apple" hardly refers to the fruit in the environment. In other words, the environment's setting can prevent ambiguity.

If the expert systems want to judge a product review, they should first know which product or feature is being referred to. Secondly, they need to build a relationship between the description and rating. NER can tell the machine which entities a sentence includes.

Then reinforcement learning can build the relationship between the entity's description and the rating.

Once the features are found from the review, the algorithm needs to determine their value. In the knowledge graph, there are possible values of the features. Thus, the algorithm can determine the feature values by rules and patterns.

Table 2 shows some patterns of reviews that describe battery life, where "Adj" stands for adjective and "Ratio" stands for the percentage of batter capacity. Thus, it is possible to create rules to match the reviews for battery life. The above patterns are insufficient to find, but expert systems can easily add more patterns.

Table 2. Pattern examples of batteries.

Pattern	Example
battery life|health is **+ Adj.***	The battery life is **perfect**
Adj.* + battery life|health	**terrible** battery health
battery capacity is|was at **+ Ratio****	battery capacity was at **84%**

For a mobile phone, the essential parts are the battery and screen [26]. For a renewed phone, the most common problems are low battery life and damage to the screen. Thus, this work creates rules to define the battery life and damage to the screen. Table 2 shows some patterns of reviews that describe battery life. Thus, it is possible to create rules to match the reviews for the battery life. The above patterns are insufficient to find all, but this approach is flexible and easily adds more patterns.

However, all of the rules are defined by humans in a traditional expert system, so it is impossible to let such an ES teach humans. Thus, this work proposes the use of reinforcement learning to mine knowledge and create rules.

3.3. Dual-Track Design for Fine-Grained Sentiment Analysis

Figure 2 shows how the dual-track system works. This system needs to find the comments about the product features, analyze their sentiments, and form an overall analysis. For sentiment classification, this work aims to use reinforcement learning and other white-box approaches as much as possible. However, when lacking the necessary knowledge, this system also uses deep learning as a backup. Thus, it uses white-box and black-box algorithms at the same time as a dual-track system.

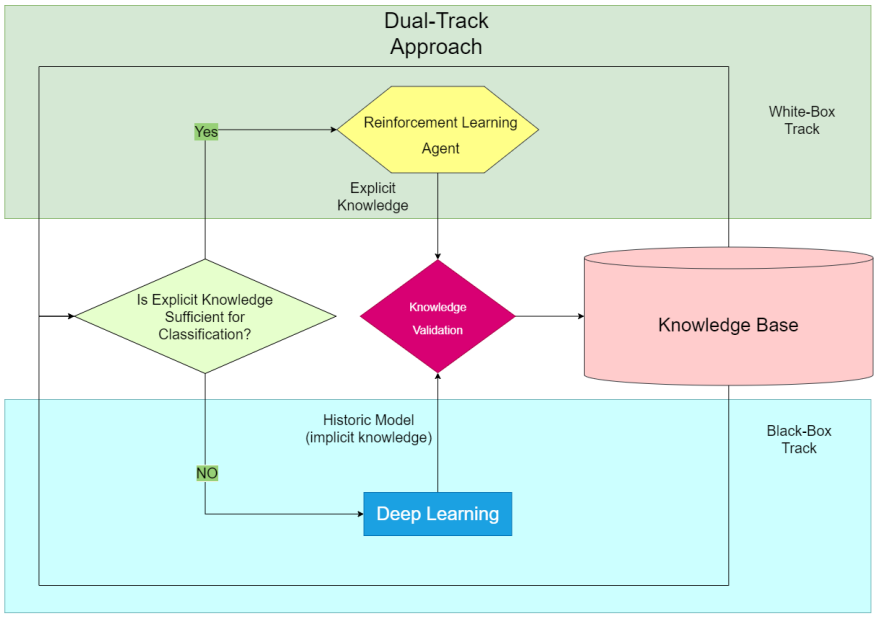

Figure 2. Dual-track fine-grained sentiment analysis.

4. Reinforcement Learning for Sentiment Analysis

Named-entity recognition can help the machine detect the entity in reviews. The next step is to build a path from the feature description of a product to the rating. Given a text description of a product to the machine, it needs to return a sentiment grade. How to determine the grade is a crucial point. The screen is essential to a phone, so customers always care about the damage to the screen when they purchase a renewed phone. They will describe that the screen has a "scratch", "crack", "scuff", or "chip". Thus, how does the machine understand the description?

The simplest way is to assign a score to each description manually. This method is possible but requires an expert to know well about the customer demands. However, e-commerce companies analyze product reviews because it is crucial for them to understand consumers. Thus, the machine needs to find answers by itself. Another possible way is to calculate the average score of the same description. This approach does not need human experts but it is not accurate. Whenever there is a connection between feature description and review grade, the average score is available. The average may have a bias when samples are not enough or when there are noises. More seriously, the machine does not ensure that the score is accurate. It does not have any confidence in it. Thus, a good design should let the machine generate a score; it also needs to give it a confidence level.

Reinforcement learning aims to find the best strategy to treat the current status in the environment. The environment (E) is a set of objects and features shown in product reviews. A certain environment can help to limit the range of possible entities in the NER. In the environment, the n_{th} review's m_{th} features f_m and their values v_k are the statuses of the environment (each feature only has one value at a time). Fine-grained sentiment analysis needs to predict the sentiment of each feature s_m and then calculate the product's overall score based on the weight of each feature w_m.

$$\hat{s_n} = \sum_{m=1}^{M} w_m * \hat{s_m} \tag{1}$$

As each feature only has one value in a review, the sentiment of a feature s_m is determined by the sentiment of its k_{th} value, s_m^k. For a feature's value v_k^m, the algorithm needs a sentiment score s_m^k. Assume the feature's value v_k^m is one of the states in the environment (E), and the sentiment score s_m^k is the action that the AI agent takes. The agent has five possible actions, which are sentiment scores from one to five. To help with the action choice, the reinforcement learning algorithm assigns a credit score c_v^a to each action. If an action leads to an accurate prediction, its credit will increase. Otherwise, it will be reduced. Thus, the action with the highest credit is the best choice.

$$a_v = \arg\max_a C(v, a), \qquad a \in A \tag{2}$$

After an action a_v is taken, the algorithm must update its credit based on the predicted result. The update is according to the reward of the action. The action obtains an enormous reward when the predicted sentiment is the same as the actual sentiment.

$$C(v, a)' = c_v^{a'} = (1 - \alpha)c_v^a + \alpha R(v, a_v) \tag{3}$$

In Formula (3), $C(v, a)'$ is the new credit of action a for value v, and $R(v, a_v)$ is the reward given for this action. The learning rate α controls the learning forgetting speed. A high learning rate lets the machine focus on the current case.

$$R(v, a_v) = R_0 - |s_m - c_v^a|, R_0 < max(A) \tag{4}$$

With labeled data of a feature's sentiment s_m, the approximation solution involves using the review's sentiment s_n to replace.

$$R(v, a_v) \approx R_0 - |s_n - c_v^a|, R_0 < max(A) \tag{5}$$

The reward is calculated based on the predicted error $s_m - c_v^a$. Assume R_0 is a constant. The reward is maximum when the predicted error is 0. If the error is larger than R_0, the reward is a negative value.

Based on the credits of a feature's value, we can calculate the suggested sentiment of a feature's value.

$$\hat{s_m} = \sum_{a=1}^{A} a \frac{c_v^a}{\sum_{a=1}^{A} C(v, a)}, \qquad a \in A \tag{6}$$

After learning, the predicted sentiment will be stable, and the entropy of the action credit will decrease. If we use the ratio of an action credit c_v^a to the total credit $\sum_{a=1}^{A} C(v, a)$ to represent the probability of action as the best action, then we can evaluate the entropy of a feature's value. Before learning, each action has the same probability, and the entropy is the largest. After learning, the entropy will decrease. Entropy shows how the machine is sure about its prediction.

$$P(v, a) = \frac{c_v^a}{\sum_{a=1}^{A} C(v, a)}, \qquad a \in A \tag{7}$$

$$H(v, A) = -\sum_{a=1}^{A} P(v, a) \log P(v, a), \qquad a \in A \tag{8}$$

The entropy of a feature's value $H(v, A)$ indicates the uncertainty level of the system responding to a specific situation. High entropy means the system tends to take different actions for the same state, which is unconverged and unreliable. A system with low entropy prefers specific responses to the same state. Hence, we can set a threshold θ_h to judge whether the machine can give a confident prediction. When a feature's entropy is higher than the threshold, further training is needed.

In addition, the change of the suggested prediction $\hat{s_m}$ can indicate whether the prediction is reliable. If a feature's value has a specific sentiment score, the suggested prediction $\hat{s_m}$ should be stable around it. Assuming that the average suggested prediction $\hat{s_m}$ in a period T is $\bar{\hat{s_m}}$, the difference to the average score should become lower during learning.

$$\bar{\hat{s_m}} = \frac{\sum_{t=1}^{T} \hat{s_m}}{T} \tag{9}$$

$$d_{\hat{s_m}} = \hat{s_m^t} - \bar{\hat{s_m}} \tag{10}$$

Feature Weight and the Overall Sentiment Score

The named-entity recognition and reinforcement learning have solved two steps of sentiment analysis—providing the sentiment of each feature and calculating the overall sentiment score. The steps need to know the weight of each feature. Humans can directly assign weights. The machine can also learn the weights by itself. As the predicted sentiment score can be calculated by Formula (1), the weights

$$\arg\min_{w_m} \sum_{1}^{N} s_n - \hat{s_n} = \sum_{1}^{N} s_n - \sum_{1}^{M} w_m * \hat{s_m} \tag{11}$$

To enable using gradient decent, the error in Formula (11) can be replaced by the mean squared error:

$$\arg\min_{w_m} \sum_{1}^{N} s_n - \hat{s_n} = \sum_{1}^{N} \frac{1}{2}[s_n - \sum_{1}^{M}(w_m * \hat{s_m})]^2 \tag{12}$$

It is a simple neural network. If replacing the $\hat{s_m}$ with Formulas (6) and (7), the whole learning process could be learned by a neural network.

$$\begin{aligned} \arg\min_{w_m} \sum_{1}^{N} s_n - \hat{s_n} &= \sum_{1}^{N} \frac{1}{2}[s_n - \sum_{1}^{M}(w_m * \sum_{a=1}^{A} a \frac{c_v^a}{\sum_{a=1}^{A} C(v,a)})]^2 \\ &= \sum_{1}^{N} \frac{1}{2}[s_n - \sum_{1}^{M} w_m * \sum_{a=1}^{A} aP(v,a)]^2 \\ s.t. \sum_{a=1}^{A} P(v,a) &= 1 \end{aligned} \tag{13}$$

If the data are sufficient, we can use a particular neural network to learn the feature weights and feature sentiment values. However, it requires the feature number to be fixed. If there is a new feature, the weights of the features need adjustments.

5. Knowledge Management in a Dual-Track System

Section 3 focuses on knowledge mining because knowledge is the core of a lifelong learning system. The reason why AlphaGo became so powerful is that it could practice the game over 30 million times, which human players can never do. Thus, AI can teach humans because AI can practice many times. Hence, the AI can teach humans its experience/knowledge obtained from the practice. Similarly, after reading thousands of product reviews, AI can tell consumers how this product is. It can even compare multiple similar products and provide professional advice. When a lifelong learning model is first created, it can be naive, but it should become wiser during learning. Thus, the design of lifelong learning should focus on setting learning goals and learning paradigms rather than the short time performance. During knowledge mining, the system will become powerful, but the knowledge must be correct.

From the fine-grained sentiment classification for a product review, the most immediate benefit involves generating a product report, which lists the score and comments of each feature of the product. Consumers can read this report and directly know the product rather than only see a simple total score. Furthermore, ES can extract consumer complaints and corresponding features as knowledge. This kind of knowledge is helpful when creating intelligent customer service. When the robot reads a complaint, it can quickly search the knowledge base to find the corresponding feature.

5.1. Knowledge Validation

Knowledge can be input by humans or learned by a machine. Whatever the source of knowledge, AI should be able to evaluate and update the knowledge after input. Thus, researchers should be aware that a lifelong learning knowledge system is much more complex than traditional knowledge systems. It not only needs knowledge mining but also requires knowledge validation and updating.

As illustrated in Figure 3, the knowledge system used for lifelong machine learning has a knowledge assessor. The traditional knowledge system only has a knowledge base and does not check whether knowledge is appropriate. The knowledge assessor [8] checks the incoming knowledge and the archived knowledge. If the knowledge is obsolete, it may become conflicted with new knowledge. Then a conflict solver is needed to judge what is correct and to merge conflicts.

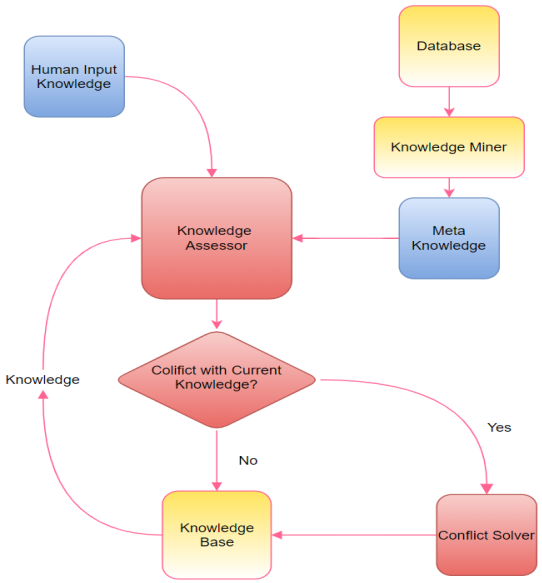

Figure 3. Knowledge system structure for lifelong machine learning.

5.2. Knowledge Updating

Explicit knowledge could be divided into static knowledge and dynamic knowledge. Static knowledge is always valid, such as "there are seven days in a week", also called common sense. Dynamic knowledge may need to adapt during the time, such as "the president of Oxford University". Knowledge includes dynamic knowledge, so the knowledge system must have the ability to adapt to change. This adaptation can be automatic or operated manually. For instance, when Joseph Robinette Biden became president of the United States, the knowledge "the president of USA" should be changed from "Donald Trump" to "Joseph Biden". Editors of Wikipedia can obtain the news from the internet and modify the knowledge manually. Machines, can also read the news and find the change. When it reads "the USA President Biden", it may be confused based on its knowledge. Thus, it may begin to doubt its previous knowledge and consider modifying it. After reading more news, when it finds all of the news that mentions "the USA President Biden" but never mentions "Donald Trump", it knows the change and updates the knowledge. This process is similar to the human learning process. Humans have very high confidence in "common sense" and low confidence in fresh discoveries. As time passes, if the discovery is always true, humans become more confident with it and regard it as "common sense". In machine learning, this process is similar to the activation of neural networks and the behavior of reinforcement learning. For fine-grained sentiment analysis, we can use entropy to identify useful knowledge. If the knowledge has low entropy, it is reliable. When entropy increases and becomes high, we can know its meaning changes, and new training is needed.

Under lifelong learning, the knowledge graph can grow over a lifetime. A new task can inherit knowledge generated from previous tasks. This is the main advantage of lifelong learning. As an iPhone is a phone, it has all components of the phone. Similarly, iPhone 7 and iPhone X are inherited from the iPhone, so the system only needs to define new components, such as "Touch ID" and "Face ID" (Figure A4 in Appendix B.2 demonstrates this development).

With the support of the knowledge graph, NER can give a more accurate prediction-based context. It is also easier for AI to detect and solve ambiguities. Researchers can input a new entity into a knowledge graph manually or order the machine to discover if there is a new entity. The knowledge graph gives each entity a precise definition so that the machine can find new entities based on context. For instance, the most crucial function of a phone is communication, so the service of its carrier is also essential. Hence, the machine

must have the ability to detect the new carrier from the product reviews. According to the knowledge graph, it is easy to know that a carrier can provide SIM cards and cell signals. Hence, the machine can read reviews and find the nouns always shown with "SIM card", "cell signal", and other carriers.

Figure 4a gives an example of how a customer comments on the carrier. The carrier "AT&T" shows with "Sim card". Once "AT&T" is known as a carrier, it is easier to find carriers, such as "Cricket". A customer mentioned "AT&T" and "Cricket" at the same time in Figure 4b, which means "Cricket" may also be a carrier. Although "apple" is also shown in lowercase, the machine can know it refers to a company rather than fruit due to carriers also being shown.

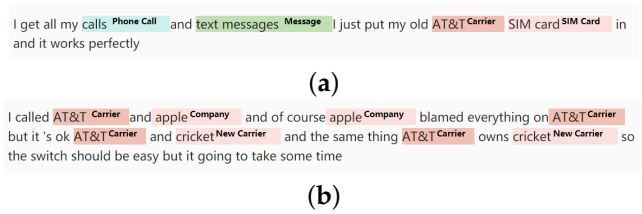

Figure 4. Visualization of BERT attention. (**a**) Carrier-named entity recognition example; (**b**) new carrier-named entity recognition discovery example.

6. Experiment

6.1. Dataset Generation

The advantage of lifelong learning is that it can continuously learn to be more potent during a lifetime. Thus, uninterrupted data providing is essential. In this work, we used a chrome plugin extension (i.e., instant data scraper) to obtain reviews from internet websites. Specifically, we crawled two lifelong fine-grained product review sentiment classification (LFSC) datasets from Amazon and Twitter. The two datasets included product reviews of five models of iPhone, including the main models of recent years. The data include phone models, customer reviews, and rates. As the customers are required to rate the products on Amazon, the Amazon dataset is a labeled dataset by default. In contrast, the Twitter dataset is unlabeled. The authors only annotate a small fraction of them.

This dataset has 12,740 samples, and sentiment grades vary from 1 star to 5 stars; 72.11% are positive samples (4 stars and 5 stars) according to Tables 3 and 4. It is an imbalanced dataset. A positive class is a major class, while neutral and negative classes are minor classes. Because of this imbalance, the machine learning model will classify samples from minor classes to positive classes by mistakes.

Table 3. Lifelong fine-grained product review sentiment classification Amazon (LFSC-A) dataset.

Product Number	Sample	Pos:Neutral:Neg
Apple iPhone 7, 32 GB, Black Fully Unlocked (Renewed)	1563	951:91:521
Apple iPhone 8, 64 GB, Space Gray Fully Unlocked (Renewed)	4444	3089:246:1109
Apple iPhone X, 64 GB, Space Gray Fully Unlocked (Renewed)	1822	1253:113:456
Apple iPhone XR, 64 GB, Black Fully Unlocked (Renewed)	4743	3780:192:771
Apple iPhone 11, 64 GB, Black Fully Unlocked (Renewed)	168	114:9:45
Total	12,740	9187:651:2902
Ratio	-	72%:5%:23%

Table 4. Lifelong Fine-grained product review sentiment classification—Twitter (LFSC-T) dataset.

Labeled	Positive:Neutral:Negative (Labeled)	Average Star	Unlabeled	Total
206	17:50:139	3.75	14,842	15,048

The LFSC-T dataset has 15,048 tweets, but the authors only annotated a few as the test dataset and retained others for further unsupervised learning research because the annotation was time-consuming. Although these tweets mentioned the iPhone, some of them were advertisements rather than product reviews. So if researchers want to analyze public opinion on social networks, they must detect and remove irrelevant tweets. Lacking labeled samples, it is difficult to train a new model based on this dataset. Thus, this work only uses it as a test dataset to investigate how can lifelong learning deal with unsupervised tasks. In this work, the authors removed the advertisements from datasets and then annotated the remaining from Twitter based on the overall product quality, such as the behavior of an Amazon customer. The author only annotated a tweet with a low (negative) rate if the customer had a negative sentiment toward an important feature. Thus, the author still annotated a tweet with a positive rating even though the tweet showed very negative emotion toward an unimportant feature.

6.2. Sentiment Prediction

RoBERTa [51] (robustly optimized BERT pretraining approach) builds on BERT's language-masking strategy and modifies key hyperparameters in BERT, including removing BERT's next-sentence pretraining objective and training with much larger mini-batches and learning rates. RoBERTa was also trained on data by an order of magnitude more than BERT for a longer time. This allowed RoBERTa representations to generalize even better to downstream tasks compared to BERT. This work used five cross-validations to test the performance of RoBERTa and our dual-track approach of the LFSC-A and LFSC-T datasets.

Table 5 indicates that the average accuracy is 90.8%, which is not low. In addition, the average F1 score of the negative class is 96%, which means the model can correctly classify most of the negative reviews. It is well known that negative reviews are more severe for both customers and sellers. Customers want to know the product's drawbacks, and sellers are afraid of the influence of negative reviews. Most merchants pay more attention to negative comments. If just looking at the F1 score, the sentiment classification problem seems similar to a solved task. However, these classification results have not provided enough information to readers.

Table 6 shows that the model trained from only the iPhone 7 dataset can reach a predicted accuracy of 90%, whereas iPhone 7 and iPhone X were tested from the rest of the models. Although the data distribution of the iPhone X was different from iPhone 7, it did not influence the prediction accuracy. Even using the whole dataset (five cross-validations) for training, the predicted accuracy increased to 90.8%. When training data are sufficient, RoBERTa cannot extra information from the training data. If meeting a bottleneck, it is necessary to use reinforcement learning for further knowledge mining.

As the LSFC-A dataset is labeled, RoBERTa can still provide an accurate sentiment classification even without NER. However, this result does not mean that RoBERTa fully understands how to evaluate an iPhone. It predicts based on the sentiment of a sentence rather than the product quality. In the Amazon dataset, customers tend to comment on the whole product. Thus, the sentiment of a review is approximately the sentiment of the product, and RoBERTa's prediction can reflect the product quality.

Though deep learning has achieved outstanding performance on the LFSC-A dataset, its mistakes are inevitable. For example, RoBERTa will classify "Does not charge" as a positive review. If the charger cannot work, it is a negative review. A dual-track approach can easily find and resolve these kinds of mistakes. Table 7 shows the performance comparison of RoBERTa and the proposed dual-track approach, where the superior figures are emboldened. It is observed that the dual-track approach sightly improves the classification accuracy from 88% to 92%, and significantly improves the F1 scores on positive classes and neutral classes, exhibiting consistent superiority over RoBERTa.

Table 5. Five cross-validation performances of RoBERTa on the LFSC-A dataset.

Cross	Acc	F1 Pos	F1 Neg	F1 Neutral	F1 Macro	F1 Micro
1	89%	88%	94%	28%	70%	88.14%
2	90%	86%	96%	14%	65%	89.77%
3	90%	86%	96%	31%	71%	89.76%
4	92%	87%	97%	24%	69%	91.67%
5	93%	83%	97%	18%	66%	95.08%
Ave	90.8%	86%	96%	23%	68.2%	90.92%

Table 6. Performance of LFSC-A via various train domain choices.

Train Domain Model	Accuracy	Positive F1	Negative F1	Neutral F1
iPhone 7	90%	84%	95%	3%
iPhone X	90%	85%	96%	12%
All models	90.8%	86%	96%	23%

Table 7. Dual-track sentiment classification for iPhone 11 (LFSC-A).

Method	Acc	F1 Pos	F1 Neg	F1 Neutral	F1 Macro	F1 Micro
RoBERTa	88%	83%	92%	0%	59%	84.77%
Dual-Track	**92%**	**91%**	**95%**	**36%**	**74%**	**91.67%**

Table 8 indicates the RoBERTa trained on the Amazon dataset (LSFC-A) performs terribly in the labeled part of the Twitter dataset (LSFC-T). If RoBERTa trained of the Amazon dataset learns how people analyze a product, it should have an outstanding performance in the Twitter dataset since both consist of product comments of the iPhone. However, if a tweet has negative sentiment about an unimportant feature, RoBERTa will classify it as a negative product review. This means RoBERTa has not learned how people evaluate iPhones. When the customers complain about an APP, RoBERTa thinks the users dislike the phone and classify reviews as negative. However, consumers can dislike the APP and still feel optimistic about the phone. RoBERTa, however, does not understand the difference between the APP and the phone, so it classifies many positive reviews as negative, and the F1 score of the positive class becomes low. To allow RoBERTa to perform better on the Twitter dataset, researchers need to annotate more samples and train a new model.

This experiment shows that the pre-trained models, such as RoBERTa, have not gained the capability to learn. It can only copy behaviors but it cannot understand their reasons. To allow AI to have a higher level of intelligence and achieve the goal of lifelong learning, we need to teach AI how humans think. Table 8 shows that the dual-track approach has remarkable improvement in the labeled part of the Twitter dataset. The average star increased to 3.69 from 2.81, much closer to the original star of 3.75. The results show that the dual-track approach has a significantly better understanding of product quality analysis. The dual-track approach knows the importance of each feature, so it exhibits a better understanding of how a feature contributes to the overall sentiment of a phone. When it reads a review, it knows whether the customer is talking about an important feature. When the consumer writes some unrelated comments, it will ignore them. It was noticed that the dual-track approach produces a better F1 score on positive and neural classes than RoBERTa.

Table 8. Dual-track sentiment classification of LFSC-T (labeled part).

Method	Acc	F1 Pos	F1 Neg	F1 Neu	F1 Macro	F1 Micro	Star
RoBERTa	47%	24%	71%	3%	33%	55%	2.81
Dual-Track	**72%**	**95%**	**75%**	**62%**	**77%**	**76%**	3.69

6.3. Fine-Grained Analysis for the Battery

Although iPhone 7 only has 1563 samples (12.27%) and its data distribution differs from others, the model trained on it still performs well in other domains (models). On the contrary, adding more training data does not significantly improve performance. Using all of the model's data results in only a slight improvement in accuracy. This situation indicates that deep learning cannot mine more information, even given more data. When algorithms cannot convert data to useful information, data become trash rather than gold mines.

When deep learning fails to mine from data, reinforcement learning can keep digging, showing the dual-track system's advantage. As expert systems use reinforcement learning to mine knowledge, the demand for data is also enormous. Although numerous samples are provided, samples of each feature are still insufficient.

Table 9 shows the learning results of the reinforcement of battery-related adjectives. The words were discovered by patterns, such as "battery (life|health) is+ Adj." and "Adj.+ battery (life|health)", as presented in Table 2. Although 2534 reviews mentioned battery, only 362 of them used adjectives to describe battery directly. Hence, there are more data needed, and lifelong learning is necessary.

Based on current data, entropy can indicate the machine's confidence in the learning result. High entropy means customers use words to describe the products but are given different scores. On the contrary, low entropy shows that consumers have the same understanding of a word. Thus, for a word with low entropy, the machine has high confidence in its correctness. For example, although 56 reviews mentioned "good", which is more than "excellent" (14 times), the entropy shows the confidence of "excellent" as being higher. These cases indicate that the customers have different understandings of the same adjective. If the entropy is high, the machine knows the knowledge is unreliable and needs more training. Moreover, when the knowledge frequency of the occurrence is low and unreliable. For example, "battery is outstanding" is a positive comment, but "outstanding" only obtains a score of 3.89 due to insufficient training samples.

Figure 5 is an example of how reinforcement learning learns the sentiment of a feature. It shows the sentiment score of each feature attribute (battery-related adjectives) over time. Some words, such as "great" and "good", become steady after a short time increase. This situation indicates that the algorithm obtains enough data and tends to be convergent. Meanwhile, the sentiment score of "new" is still increasing, which means that it needs more data and study. In other words, its current value is unreliable.

Knowing what knowledge is reliable is essential to lifelong learning. Although the value change in Figure 5 can help us evaluate the knowledge reliability, we still need a clearer indicator to measure it. In this work, we use entropy as an indicator.

Though the scores of "great" and "good" have become steady in Figure 5, their entropy values are quite different. From Figure 6, we can see, the entropy of "great" is much lower than the value of "good". This situation shows that the value of "great" has higher reliability. Meanwhile, we can see the entropy of "new" gradually decrease to become convergent. In addition, although words such as "excellent" and "amazing" have not converged, their entropies are low, so their sentiment scores also have high reliability. Entropy is a good tool that can help us evaluate the quality of knowledge and judge how to use knowledge.

Table 9. Sentiment score of battery-related adjective.

Word	Score	Frequency	Entropy	Word	Score	Frequency	Entropy
amazing	4.67	22	0.4	great	4.66	60	0.4
excellent	4.39	14	0.53	unhealthy	3.0	0	0.54
perfect	4.41	6	0.54	good	4.37	56	0.62
defective	1.77	4	0.65	dead	2.07	5	0.73
best	4.08	4	0.79	draining	1.93	5	0.79
damaged	2.08	4	0.79	wonderful	3.89	2	0.81
drained	3.44	2	0.81	fine	3.84	7	0.81
incredible	3.89	2	0.81	impeccable	3.89	2	0.81
awful	2.11	2	0.81	faulty	2.11	2	0.81
normal	3.89	2	0.81	outstanding	3.89	2	0.81
fantastic	3.89	2	0.81	failing	3.89	2	0.81
old	3.18	3	0.85	junk	2.64	3	0.85
horrible	2.52	12	0.87	weak	3.15	4	0.89
new	3.73	51	0.89	terrible	2.53	6	0.9
healthy	2.56	2	0.91	crappy	3.22	2	0.91
nice	3.44	2	0.91	choppy	3.29	1	0.92
critical	2.43	1	0.92	seriously	3.57	1	0.92
inflamed	2.43	1	0.92	spanking	3.57	1	0.92
kaput	2.71	1	0.92	questionable	2.43	1	0.92
okay	3.29	1	0.92	spectacular	3.57	1	0.92
wrong	3.57	1	0.92	pitiful	3.0	1	0.92
fake	3.0	1	0.92	awesome	3.29	1	0.92
magnificent	3.57	1	0.92	generic	2.71	1	0.92
dying	2.43	1	0.92	operational	3.57	1	0.92
diminished	2.71	1	0.92	ridiculously	2.71	1	0.92
leaking	2.71	1	0.92	reasonable	0.57	1	0.92
swollen	3.0	3	0.93	bad	2.65	26	0.93
poor	2.6	5	0.94	low	2.66	15	0.95

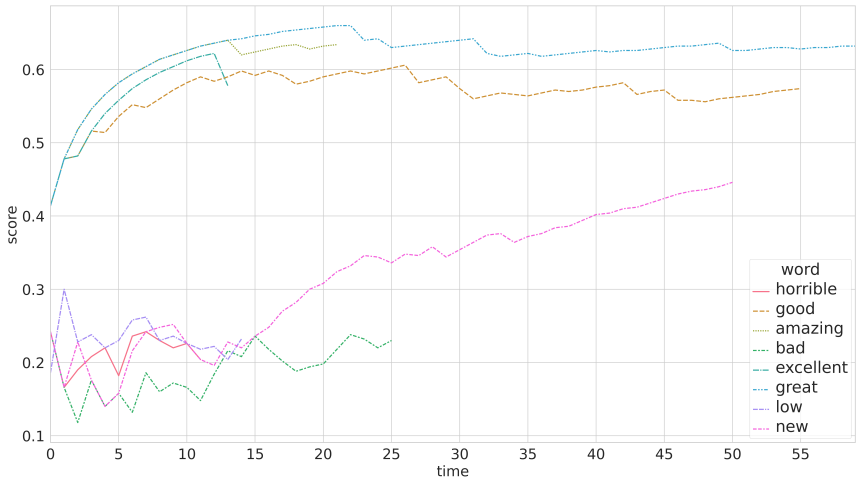

Figure 5. Sentiment score over time of battery-descriptive words.

When people describe battery life, they can use a numeric method, such as a percentage. So, it is necessary to learn the relationship between the percentage of the battery life and sentiment. Although the battery life is from 1% to 100%, reinforcement learning can also work if data are sufficient. When data are insufficient, merging input range into bins is reasonable, such as under 80%, 80–90%, over 90%, etc.

Ideally, the percentage of battery life should be positively correlated with the sentiment score. However, it is not a satisfactory positive relationship according to Table 10. A lower percentage may result in a higher sentiment score, but a high percentage has a higher possibility of receiving a good score. When the percentage of battery life is over 90%, the sentiment score is over 4.50 except "96%". Once the percentage is lower than 90%, the score is much harder to reach 4.50. The quality requirement of the renewed phone battery life is at least 80%, so customers can accept a phone battery life of over 80%. However, when the battery life is at 81% and 80%, they become nervous. The average score of the battery is 3.96 according to Table A3 and the scores of 81% and 80% are 3.70 and 3.28, which are lower than average. When the percentage drops to 79%, the score quickly drops to 2.50, which equals a serious complaint. To be noticed, when the percentage

drops to 89%, customers are also unsatisfied. This case shows that customers feel unhappy when the quality drops from high to low. This result can also help people understand the customers' emotions.

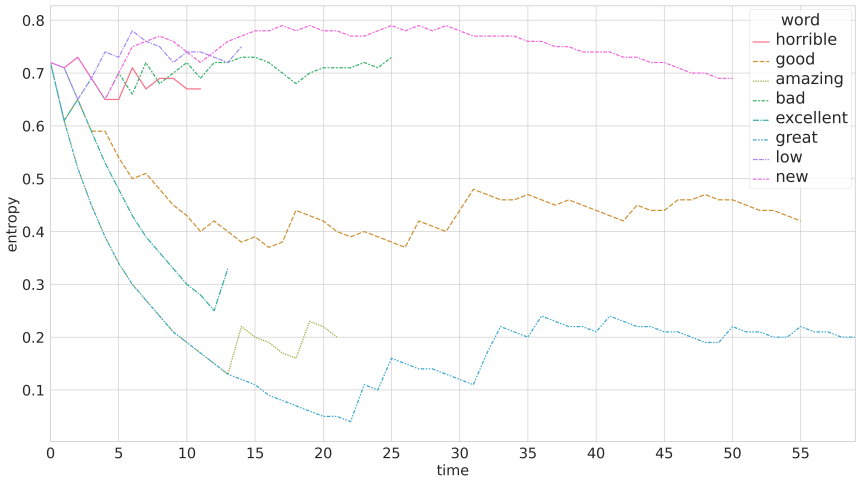

Figure 6. Entropy value over time of the battery-descriptive words.

Although the relationship between the battery life percentage and sentiment score is not perfectly positive, it can still help the machine understand the described words. Once using the descriptive words in Table 9 to replace the score in Table 10, the machine can understand the relationship between the battery life percentage and descriptive words. A descriptive word can comment on a close score. For instance, "excellent" means a score of 4.80, so the machine can comment on all scores higher than 4.80 as "excellent".

Table 10. Sentiment score and description of the battery life.

Battery Life	Score	Freq	Entropy	Battery Life	Score	Freq	Entropy
100	4.88	118	0.26	95–99	4.73	95	0.46
90–94	4.64	167	0.55	85–89	4.26	104	0.87
80–84	4.05	76	0.95	Unqualified	2.59	31	1.48

Table 11 uses a describing word to comment on each battery life percentage. In this table, we can see that customers are satisfied when the battery life is over 90% and tend to decrease their rating when it is between 80% and 90%. Consumers will give negative ratings when the battery life is below 80% and is unqualified.

Table 11. Sentiment score of the battery life.

Battery Life	Score	Description	Battery Life	Score	Description
100	4.88	Amazing	95–99	4.73	Great
90–94	4.64	Great	85–89	4.26	Good
80–84	4.05	Good	Unqualified	2.59	horrible

6.4. Fine-Grained Analysis for Screen

Similar to the battery, customers also care about the screen. The most common problem with the renewed phone is screen damage. Consumers use words, such as "scratch", "crack", and "chip" to describe the damage. However, it is difficult for the system to understand these kinds of descriptions. This work also uses reinforcement learning to learn the sentiment score of the descriptive words. Sentiment scores of screen-related words are shown in Table 12, where only the stage differences after ten times shown in the review are listed. In this table, besides "scuff", all other words are lower than 4. This indicates that customers cannot normally stand the damage to the screen. The most common problem is "scratch", with a score of 3.4. Moreover, the scores of "chip" and "crack" are 3.27 and 2.63.

From reinforcement learning, the machine can know that "scuff" means slight damage, and the customer can tolerate it. However, "chip" and "crack" are more severe problems and are unacceptable. Although human experts can directly provide scores, reinforcement learning can save the labor force and achieve the same outcomes.

Table 12. Sentiment scores of screen-related words; "En" refers to "Entropy".

Word	Score	Freq	En	Word	Score	Freq	En
perfect	4.71	22	0.32	flawless	4.6	8	0.41
great	4.57	7	0.44	protective	4.38	5	0.56
amazing	4.33	3	0.59	bad	1.71	4	0.62
clear	4.29	3	0.62	scuff	4.29	4	0.62
big	4.24	5	0.63	sensitive	4.23	3	0.65
replacement	2.38	2	0.65	large	4.23	2	0.65
good	4.07	11	0.68	beautiful	4.17	3	0.68
damaged	1.83	3	0.68	smooth	4.17	2	0.68
freezing	4.17	2	0.68	intermittent	4.0	2	0.76
flickering	2.0	2	0.76	small	4.0	2	0.76
chip	3.27	6	0.76	defective	2.14	6	0.79
crack	2.63	53	0.79	scratch	3.4	350	0.82
delayed	3.67	2	0.84	faulty	3.38	2	0.86
replaced	2.62	5	0.86	unresponsive	2.73	3	0.86
popped	3.38	2	0.86	poor	3.0	2	0.86
original	3.75	3	0.86	broken	2.67	11	0.89

6.5. Named-Entity Recognition and Fine-Grained Analysis for the Carrier

Discovering new entities is also essential during knowledge mining. For instance, the phone is related to carriers, so finding the carriers from reviews is also crucial. Creating a list of carriers is one possible way, but there are always new carriers. Thus, it is necessary to discover carriers from reviews. Although new carriers are unknown, they have the same features as old carriers. When customers mention a carrier, they may complain about the SIM card or cell signal. One possible way to discover the carriers is to find the organization in the review about SIM cards and signals.

Stanford CoreNLP [52] is used to annotate the organization. CoreNLP has a named-entity recognition function and it can annotate the organization. However, it can only recognize the organization name starting with the up case, such as "AT&T". If the input changes to lower case, such as "AT&T", CoreNLP cannot work.

Table 13 lists the organizations recognized by CoreNLP. Although it has some mistakes, such as "All", "Simple", etc., it narrows the search scope. There are many correct carriers in the list, including "AT&T", "Verizon", "Sprint", "MetroPCS", "Tracfone", "Comcast", "T-Mobile", and "Vodafone", "BT". There are many typos or abbreviations for a name. For example, "AT&T" also called "ATT", and "T-Mobile" was written as "TMobile", and "TMobil".

Table 13. Named-entity recognition result by CoreNLP.

Word	Frequency	Word	Frequency	Word	Frequency
AT&T	123	Verizon	58	Apple	14
SIM	10	ATT	6	Sprint	5
Amazon	4	MetroPCS	3	Tracfone	2
Metropcs	2	Comcast	1	Metro	1
PCS	1	Motorola	1	Mobil	1
Kroger	1	Vodafone	1	S7	1
Lightning	1	TMobil	1	Service	1
All	1	Carriers	1	Local	1
Simple	1	Mobile	1	BT	1

Using existing NER tools, such as CoreNLP, is efficient, but it only works for the apparent entity. If the entity is written in lowercase, CoreNLP cannot recognize it. Thus, this work also uses coordinating relationships to find new entities, if two entities belong to the same class and coordinating conjunction can connect them. Based on this idea, this work annotates sentences and records which entities are shown together with known carriers.

There are two carriers found by the coordinating relationship in Table 14. "Cricket" refers to "Cricket Wireless", but CoreNLP may regard it as a kind of sport. Although Google is an internet company, it has also "Google Fi", and many customers use it as a carrier. In addition, based on Table 15, "TMobile" and "Version" are "T-Mobile" and a typo of "Verizon". "Transferred" is a verb, but CoreNLP recognizes it as a noun, so CoreNLP can also involve mistakes.

Table 14. Named-entity recognition results from the coordinating relationship.

Word	Frequency	Word	Frequency	Word	Frequency
Apple	1	Cricket	2	TMobile	3
Version	2	Google	1	Transferred	1

Table 15. Abbreviation and Typo of Carriers.

Carrier	Abbreviation and Typo	Carrier	Abbreviation and Typo.
AT&T	ATT	Cricket	
T-Mobile	TMobile, T Mobile	Comcast	
Verizon	Version	Sprint	
MetroPCS	Metro, PCS	Tracfone	
Google Fi	Google	Vodafone	
BT Mobile	BT	Boost Mobile	Boost

Once obtaining the list of carriers, it is possible to analyze how to choose carriers before a purchase. This work also used reinforcement learning to evaluate the carriers.

"Sprint" obtained the worst rating in Table 16. The single score can indicate that Sprint has some problems, but it is still unclear. There were 66 of 88 reviews rating Sprint lower than four stars.

Table 16. Sentiment scores of carriers.

Word	Score	Freq	Entropy	Word	Score	Freq	Entropy
Metro	4.35	64	0.87	Google Fi	4.1	12	1.26
Tracfone	4.05	7	1.02	Cricket	4.02	30	0.92
BT	3.57	1	1.48	Comcast	3.57	1	1.48
Vodafone	3.57	1	1.48	Verzion	3.47	27	1.41
AT&T	3.13	119	1.17	T-mobile	3.12	79	1.29
Boost	3.1	28	1.47	Sprint	2.24	88	1.29

According to Table 17, the main problem with "Sprint" involves incompatibility, which means the renewed phone cannot use the SIM card from it. It is incompatible for two reasons, i.e., the phone model or carrier lock. A few iPhone models do not support Sprint, so this may lead to incompatibility. Most problems are due to the carrier lock. If "AT&T" sells a phone, it may have a carrier lock to prevent the users from changing carriers. In addition, if a phone was bought via a loan or was stolen, the carrier will lock the phone. In summary, the main problem with "Sprint" is the carrier lock. This problem also exists in complaints to other carriers. Thus, customers can know that the phone is not really fully unlocked as the advertising says.

Table 17. Complaints to Sprint.

Reason	Frequency	Reason	Frequency	Reason	Frequency
Incompatible	36	Carrier Lock	20	Loaned Phone	4
Customer Service	3	Illicit Phone	2	No SIM Card	1

Telling the customers what problems the carriers have is more helpful than providing them with sentiment scores. Customers will know that the phone is not fully unlocked, and the sellers will know that their products have problems.

7. Discussion and Future Work

This work proposes a lifelong dual-track approach for fine-grained sentiment classification. The authors designed a reinforcement learning-based white-box algorithm for fine-grained sentiment analysis. Reinforcement learning can mine knowledge from product reviews and evaluate knowledge reliability. Compared with the deep learning method, the dual-track approach can handle the fine-grained sentiment analysis and improve classification performance. It reaches the sentiment classification macro F1 of 74% on the Amazon dataset (iPhone 11) and 77% on the Twitter dataset. Compared with the transformer-based approach (RoBERTa), it achieves a 25.42% promotion on the Amazon dataset and 133% on the Twitter dataset. In summary, the dual-track approach has both better explainability and classification performance than RoBERTa. The proposed system has the potential to be applied to other areas, such as auxiliary medical diagnosis systems and auxiliary financial market decision systems.

Previous lifelong learning mainly focused on performance promotion or knowledge mining and did not investigate knowledge reliability. This work proposes using reinforcement learning as a tool to mine and validate knowledge. The entropy of knowledge can indicate its reliability. Humans and machines can use entropy values to determine whether to trust knowledge. As this is early research, the data amounts are insufficient. We did not build a systemic method to judge which knowledge was reliable. For example, we used entropy to evaluate knowledge, but a threshold was missing. More data and experiments are needed to figure out the threshold.

As this work aims to implement a lifelong machine learning algorithm, we should collect and transfer knowledge into different tasks. However, this work only collects knowledge about the phone. Although we successfully transferred knowledge from the Amazon dataset to the Twitter dataset, both datasets were about the phone. In other words, they involved the same tasks, but the sources were different. Next, the authors need to collect more datasets on various products, such as laptops and PCs. Humans can learn knowledge from one task and use them in different tasks, so a lifelong learning algorithm also needs such ability.

For the fine-grained sentiment analysis, this work lacks adequate standards to evaluate the performance. The authors only provide a series of examples, which are hard to judge. Thus, the authors will attempt to build a new fine-grained sentiment analysis dataset to test its performance.

The dual-track system can accumulate knowledge for reusing, and the knowledge system needs a standard. If two machines are mining and saving knowledge with different standards, sharing knowledge will result in challenges. For example, there are various reinforcement learning algorithms. If two machines use different algorithms, how do they treat each other's outcomes? Hence, the dual-track system is only the first step to lifelong learning. An actual lifelong learning system needs the contribution of the whole AI community. How to create such a standard for knowledge mining and sharing should be investigated.

8. Conclusions

This paper investigated how to enable human–AI collaborations and how AI can teach humans after AI becomes more intelligent than humans. One reasonable solution is to make sure the AI that humans develop is explainable AI. Only if AI has high explainability can humans understand AI and learn from it. Moreover, considering the difficulty in creating a high-performance and explainable AI; the authors suggest using a dual-track approach. Under the dual-track design, expert systems (ES) are responsible for interacting with humans. Humans can directly teach expert systems, and ES can mine knowledge by themselves. When ES practice numerous times, they know more than humans and can even teach humans. In this new design, a key component is a knowledge assessor. It is responsible for validating and updating knowledge. The authors demonstrate that the proposed approach can bring 133% promotion of the Macro-F1 score in the Twitter

sentiment classification task and 27.12% promotion of the Macro-F1 score in the Amazon iPhone 11 sentiment classification task, respectively, by conducting a series of experiments and suggesting that the AI community pay more attention to it in the near future.

Author Contributions: Conceptualization, S.-U.G. and X.H.; methodology, S.-U.G. and X.H.; software, X.H.; validation, X.H.; formal analysis, X.H.; investigation, S.-U.G. and X.H.; resources, X.H.; data curation, X.H.; writing—original draft preparation, X.H.; writing–review and editing, X.H., N.X. and Z.L.; visualisation, X.H.; supervision, S.-U.G., K.L.M., P.W.H.W. and D.L.; project administration, S.-U.G., K.L.M., P.W.H.W. and D.L.; funding acquisition, S.-U.G., K.L.M., Z.L. and D.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research is partially funded by the research funding: XJTLU-REF-21-01-002.

Data Availability Statement: The LSFC dataset is available at https://github.com/DerekGrant/LSFC, accessed on 11 January 2023.

Acknowledgments: This research is partially supported by the Department of Computing, School of Advanced Technology and AI University Research Centre, Xi'an Jiaotong-Liverpool University (XJTLU), Jiangsu (Provincial) Data Science and Cognitive Computational Engineering Research Centre at XJTLU; and research funding: XJTLU-REF-21-01-002. This paper is extended from a conference paper, "Can AI Teach Humans? Humans AI Collaboration for Lifelong Machine Learning"; 2021 4th International Conference on Data Science and Information Technology.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AI	artificial intelligence
BERT	bidirectional encoder representations from transformers
ES	expert system
LML	lifelong machine learning
LFSC	lifelong fine-grained product review sentiment classification

Appendix A

Appendix A.1. Different Type Expert System

- Rule-based systems
  A rule-based ES performance is based on rules generated by experts, such as the IF—THEN algorithm. Once the data are inputted, the ES performance is based on rules and produces an appropriate decision.
- Knowledge-based systems
  Knowledge-based systems [30] include the knowledge base, inference engine, and knowledge engineering manager. Unlike rule-based systems, knowledge-based systems can make inferences based on knowledge to generate results or produce knowledge by themselves.
- Case-based systems
  Case-based systems judged by previous similar cases. Thus, there should be a case base to store cases, and a search engine to search for similar cases.

Appendix A.2. Example of Decision Tree

Appendix A.2 is an example of the decision tree for sentiment analysis of phone product reviews. Sellers can input customer comments to analyze their sentiments.

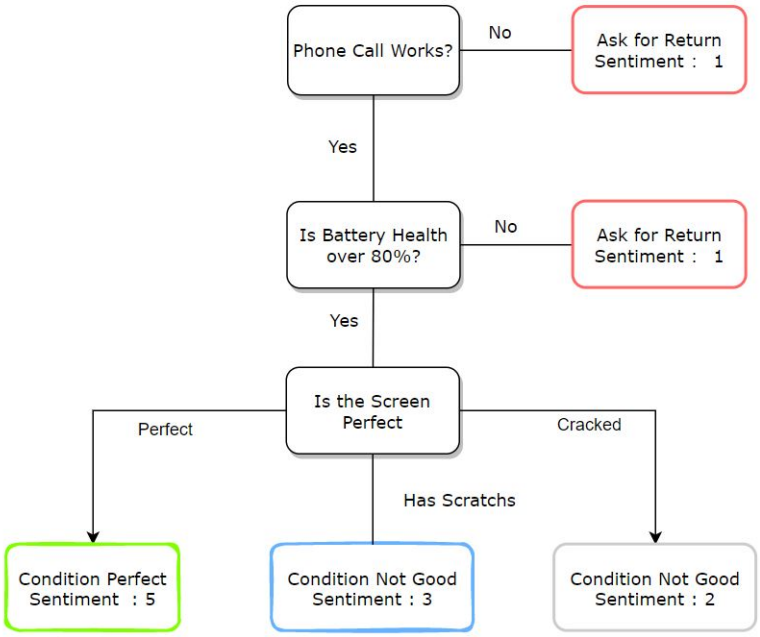

Figure A1. Decision tree for phone sentiment classification.

Appendix A.3. Visualization of Feature Maps

It is too difficult to explain what is behind the deep neural network using current technology. Thus, scientists mainly attempt to visualize the network parameters. Convolutional neural networks (CNNs) [53] use convolutional maps to extract features of pictures. Researchers visualize the feature maps with heatmaps to see how CNNs make decisions. For instance, Figure A2 shows three feature maps from the "Conv2D:0" layer of the YOLOv4. Red pixels mean activated units, and blue pixels decrease the activation [54]. From the heatmaps, researchers can understand how the network inferences and evaluates the correctness of this decision. With visualization, CNN is not entirely a "black box". Although researchers can partially understand what CNN focuses on, they still cannot understand all heatmaps. It is also impossible to directly tell the network to pay more attention to a specific area.

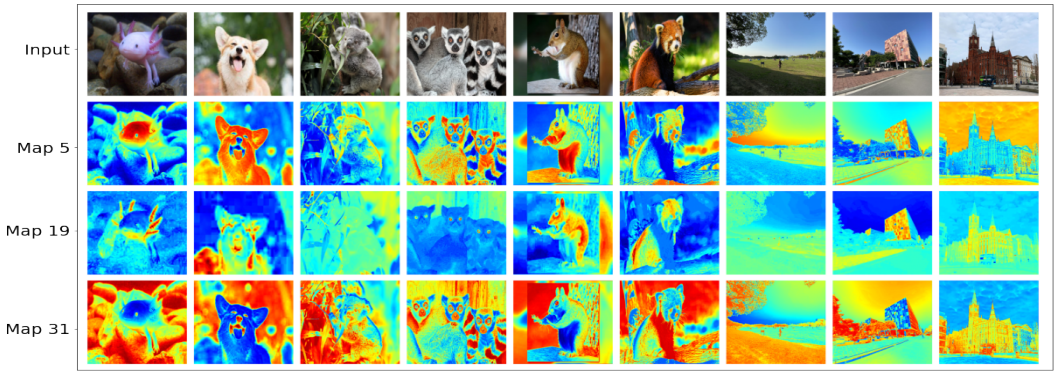

Figure A2. Visualization of three different feature maps in YOLOv4.

Appendix B. Product Review Examples

Appendix B.1. Amazon Product Review Example

An entity-inherit example of an iPhone is shown in Figure A4 and an entity example is illustrated in Figure A5.

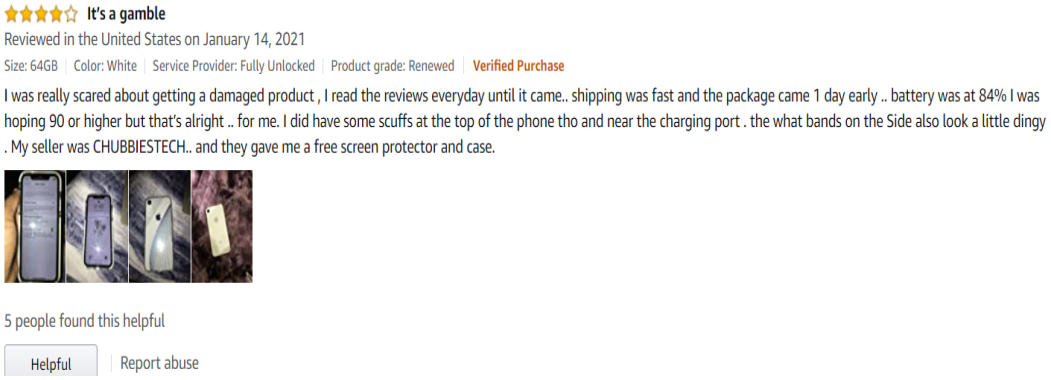

Figure A3. Amazon product review example for a renewed iPhone XR.

Appendix B.2. Entity Example for Phone

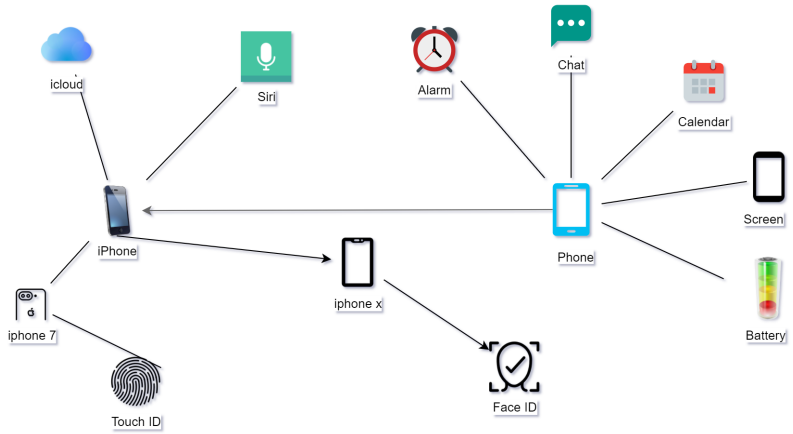

Figure A4. Entity Inherit Example for iPhone.

Figure A4 shows an example of the phone entity relationship. A phone has multiple functions. iPhone inherits the phone and has some new attributes.

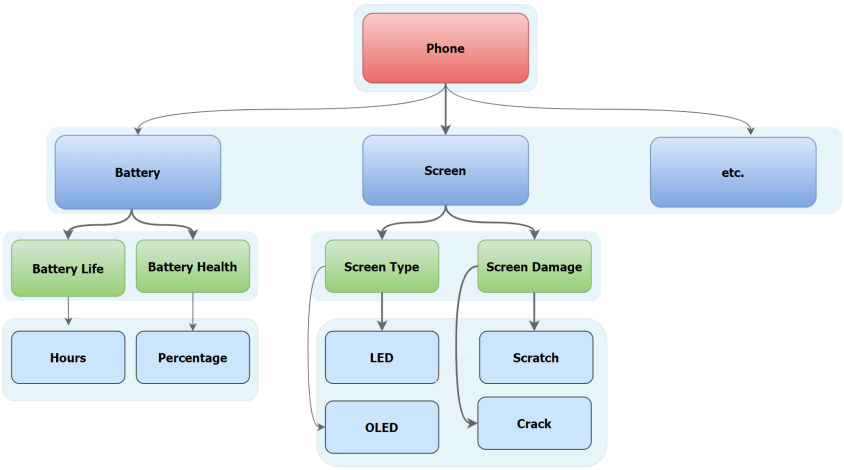

Figure A5. Entity Example for Phone.

Appendix B.3. Explanation for Battery Maximum Capacity

Table A1 shows how to evaluate the battery life via maximum capacity. It is not easy to directly tell (or teach) deep learning to understand, but expert systems can simply create new rules to match.

Table A1. Explanation of the battery's maximum capacity.

Capacity	Score	Explanation
over 90% can support normal peak performance	5	good battery life
80–90% can support normal peak performance	4	acceptable for renewed phones
under 80% cannot support normal peak performance	1–2	disqualification

Appendix C. Dataset Statistics

Appendix C.1. Annotation Standard of the Twitter Dataset (LFSC-T)

Table A2 presents a brief annotation standard of the Twitter dataset, where the full mark is 5. The author annotates the tweets according to the overall product quality rather than the sentence sentiment. This is because consumers tend to rate a product based on its overall quality rather than a single feature unless the feature is very important. Twitter users only write negative comments in a tweet, so a tweet's sentiment cannot be the attitude of the consumer to the product.

Table A2. Annotation standard of the twitter dataset (LFSC-T).

Major Feature	Unimportant Feature	Overall Rate
Positive (4–5)	Positive (4–5)	Positive (4–5)
Positive (4–5)	Negative (1–2)	Neutral (3), Positive (4)
Negative (1–2)	Positive (4–5)	Negative (1–2)
Negative (1–2)	Negative (1–2)	Negative (1–2)

Appendix C.2. Statistic of LFSC-A

Table A3 provides the sentiment analysis example of LFSC-A. The iPhone XR obtained the highest score of 4.21 and the lowest score of 3.5. This score tells the customers that the iPhone 7 is not popular but it does not explain why. Looking at the scores of the battery and screen, the reason becomes clearer. The iPhone 7 only obtained a 3.04 on the battery and 2.91 on the screen, which are lower than the average scores. This case means that the customers are unsatisfied with the battery and screen conditions of the iPhone 7. This external information warns customers that the renewed iPhone 7 is quite old and not in good condition.

Table A3. Fine-grained sentiment analysis of LFSC-A.

Product (Renewed)	Total Score	Battery	Mention Rate of Battery	Screen	Mention Rate of Screen
Apple iPhone 7	3.50	3.04	16.95%	2.91	6.21%
Apple iPhone 8	3.84	3.73	20.99%	3.31	10.96%
Apple iPhone X	3.84	4.11	23.16%	3.07	22.61%
Apple iPhone XR	4.21	4.39	18.62%	3.80	15.86%
Apple iPhone 11	3.79	4.13	18.45%	3.56	13.69%
Average	3.94	3.96	19.89%	3.45	13.9%

Appendix C.3. Statistic of LFSC-A

The mentioned rate of a feature shows the attention each feature receives. From Table A4, it is clear that customers care about the battery more than the screen. The iPhone 7's mention rates of both the battery and screen are significantly lower than other models, and the average review length is shorter. This situation warns that the data distribution of the iPhone 7 is different from other models. When data distributions are different, sampling data from the target domain is always helpful. However, this domain adaptation does not work for LFSC-A.

Table A4. Fine-Grained Sentiment Analysis upon LFSC-A.

Product (Renewed)	Mention Rate of Battery	Mention Rate of Screen	Mention Rate of Review
Apple iPhone 7	16.95%	6.21%	84
Apple iPhone 8	20.99%	10.96%	164
Apple iPhone X	23.16%	22.61%	180
Apple iPhone XR	18.62%	15.86%	163
Apple iPhone 11	18.45%	13.69%	161

Appendix D. Case Analysis

To make the Amazon review more intuitive, the Amazon product page of a renewed iPhone X is shown in Figure A6.

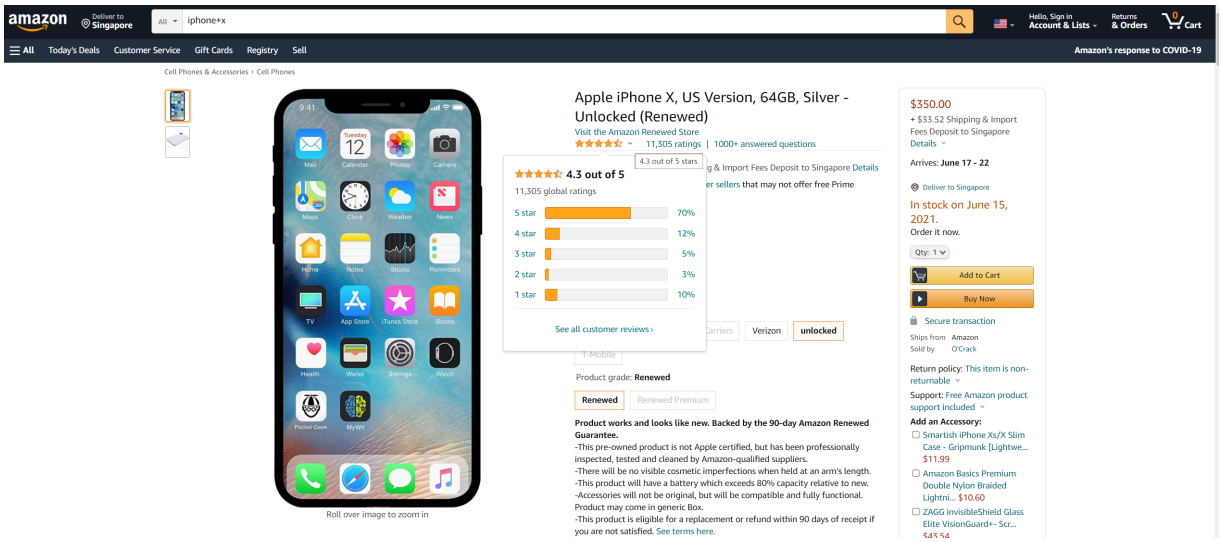

Figure A6. Amazon product page of a renewed iPhone X.

When customers purchase a phone, they cannot read all reviews. What they can do is check the overall rating and read a few comments. However, even most customers satisfied with the product cannot guarantee the product is suitable for everyone.

For example, the renewed iPhone X has 4.3 stars (Figure A6), and only 13% of customers gave negative reviews (under three stars). However, is it a good product? Customers cannot make correct decisions because they do not know why 13% of customers dislike it and whether they have the same problems.

One customer (Figure A7) purchased an iPhone X because he thought the rating was high. However, he then found it was incompatible with his carrier (Sprint), so he then gave it a score of 1 star. If he was able to obtain the information from Tables 16 and 17, he would know that Sprint received many complaints due to incompatibilities. Then he could have given up this purchase or changed carriers.

From Table 10, customers could know about the distribution of the battery life. Although the renewed phone only guarantees a battery life of at least 80%, many customers expect a high battery life, even 100%. If they know the real battery life distribution, they may consider it again.

Table A5 lists four negative review cases. The first and third reviews mentioned that the battery life was 80% and 78%, which are lower than the promotion in the advertisement. In such cases, customers normally give stars of 2 or 3. The first customer wanted a "refurbished" phone, in which battery life was at 100%. However, the "renewed" phone only promised a battery life of over 80%. Thus, the customer had a misunderstanding about the renewed phones. If the truths shown in Figure A8 are shown to everyone, misunderstandings could be avoided.

This iPhone X advertised as (Renewed) for Sprint [Carrier] seemed to have very positive ratings (4.5 stars from over 7K reviews); and since the price was very good, I jumped on it.Buyer Beware:Phone came with a charging cord and cube and looked in Excellent shape.Unfortunately upon taking it to the Sprint [Carrier] /T-Mobile [Carrier] store, my dejected daughter (it was an 18th birthday present) found out the phone was INCOMPATIBLE [Incompatible]. I phoned Sprint Tech Support and after providing the IMEI number verified the phone had been used with BOOST Mobile [Carrier] and for whatever reason, made the device incompatible with Sprint [Carrier] now. Phone is unusable despite being icloud and Blacklist CLEAN.Shame on this seller. There were other reviews claiming the seller was possibly selling 'stolen phones [IllicitPhone]'. I don't know if that's true but I'm going elsewhere for anther.

Figure A7. Customer review case of a renewed iPhone X.

Table A5. Customer review cases of a renewed iPhone.

Case	Star	Review	Reason	Suggestion
I	1	Had to order a replacement because the **battery life** was at **80%**. That is a used not a refurbished phone and it definitely not worth what I paid.	Low battery life, 80%	2 stars
II	1	**Battery life 81%**, 1 burned pixel, every side of the phone with **scratches**. I would not recommend buying this product.	Low Battery Life, 81% Screen damage Scratches	3 star
III	2	The phone I received was in OK condition appearance-wise, but the phone's **battery health is 78%** relative to new, which is below the 80% mentioned in the advertisement.	Low battery life, 78%	2 star
IV	1	The phone came with the front speaker dirty, scratches on the back and the **battery health is only 85%**	Low battery life, 85% Scratches, dirty speaker	4 stars

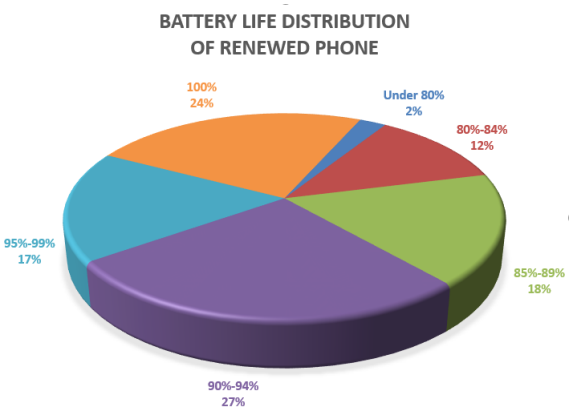

Figure A8. Battery Life Distribution of Renewed iPhone.

The second customer gave 1 star due to the battery life being 81%, screen damage, and scratches. The battery life is over 80%, which is qualified. Burned pixels and scratches are not crucial flaws, such a product should get around three stars. The fourth case is similar, i.e., the battery life is 85% and there is no significant damage to the phone. This kind of phone commonly obtains four stars, but this customer only gave it 1 star, which is not objective.

Although four customers gave negative reviews, only cases one and three could apply to customer service, returns, or replacements. With the knowledge learned by reinforcement learning, the machine can know whether a customer gave a fair rating and take suitable actions.

References

1. Adiwardana, D.; Luong, M.-T.; So, D.R.; Hall, J.; Fiedel, N.; Thoppilan, R.; Yang, Z.; Kulshreshtha, A.; Nemade, G.; Lu, Y.; et al. Towards a human-like open-domain chatbot. *arXiv* **2020**, arXiv:2001.09977.
2. Eo, S.; Park, C.; Moon, H.; Seo, J.; Lim, H. Comparative analysis of current approaches to quality estimation for neural machine translation. *Appl. Sci.* **2021**, *11*, 6584. [CrossRef]
3. He, P.; Liu, X.; Gao, J.; Chen, W. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv* **2020**, arXiv:2006.03654.
4. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.* **2015**, *115*, 211–252. [CrossRef]
5. Zhu, X.; Zhang, L.; Du, J.; Xiao, Z. Full-abstract biomedical relation extraction with keyword-attentive domain knowledge infusion. *Appl. Sci.* **2021**, *11*, 7318. [CrossRef]
6. Silver, D.; Schrittwieser, J.; Simonyan, K.; Antonoglou, I.; Huang, A.; Guez, A.; Hubert, T.; Baker, L.; Lai, M.; Bolton, A.; et al. Mastering the game of go without human knowledge. *Nature* **2017**, *550*, 354–359. [CrossRef]
7. Chen, Z.; Liu, B. Lifelong machine learning. *Synth. Lect. Artif. Intell. Mach. Learn.* **2016**, *10*, 1–145.
8. Hong, X.; Guan, S.-U.; Man, K.L.; Wong, P.W. Lifelong machine learning architecture for classification. *Symmetry* **2020**, *12*, 852. [CrossRef]
9. Hong, X.; Guan, S.-U.; Wong, P.; Nian, X.; Man, K.L.; Liu, D. Can ai teach humans? Humans AI collaboration for lifelong machine learning. In Proceedings of the 2021 4th International Conference on Data Science and Information, Shanghai, China, 23–25 July 2021.
10. 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, Weihai, China, 27–29 October 2018; pp. 76–79.
11. Thrun, S.; Mitchell, T.M. Lifelong robot learning. *Robot. Auton. Syst.* **1995**, *15*, 25–46. [CrossRef]
12. Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; Liu, P.J. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv* **2019**, arXiv:1910.10683.
13. Hong, X.; Zhang, J.; Guan, S.-U.; Yao, D.; Xue, N.; Zhao, X.; Huang, X. Incremental maximum gaussian mixture partition for classification. In *Proceedings of the 2017 2nd Joint International Information Technology, Mechanical and Electronic Engineering Conference*; Atlantis Press: Dordrecht, The Netherlands, 2017.
14. Pal, G.; Hong, X.; Wang, Z.; Wu, H.; Li, G.; Atkinson, K. Lifelong machine learning and root cause analysis for large-scale cancer patient data. *J. Big Data* **2019**, *6*, 1–29. [CrossRef]
15. Zhang, J.; Hong, X.; Guan, S.-U.; Zhao, X.; Xin, H.; Xue, N. Maximum Gaussian mixture model for classification. In Proceedings of the 2016 8th International Conference on Information Technology in Medicine and Education (ITME), Fuzhou, China, 23–25 December 2016; pp. 587–591.
16. Xu, F.; Uszkoreit, H.; Du, Y.; Fan, W.; Zhao, D.; Zhu, J. Explainable AI: A brief survey on history, research areas, approaches and challenges. In *CCF International Conference on Natural Language Processing and Chinese Computing*; Springer: Berlin, Germany, 2019; pp. 563–574.
17. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [CrossRef] [PubMed]
18. 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), Beijing, China, 26–31 July 2015; Volume 2, pp. 750–756.
19. Yuan, Y.; Lam, W. Sentiment Analysis of Fashion Related Posts in Social Media. In Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining, Tempe, AZ, USA, 21–25 February 2022; pp. 1310–1318.
20. Fan, C.; Gao, Q.; Du, J.; Gui, L.; Xu, R.; Wong, K. Convolution-based memory network for aspect-based sentiment analysis. In Proceedings of the 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, Ann Arbor, MI, USA, 8–12 July 2018; pp. 1161–1164.
21. Liu, Q.; Zhang, H.; Zeng, Y.; Huang, Z.; Wu, Z. Content attention model for aspect based sentiment analysis. In Proceedings of the 2018 World Wide Web Conference, Lyon, France, 23–27 April 2018; pp. 1023–1032.
22. Cambria, E.; Li, Y.; Xing, F.; Poria, S.; Kwok, K. SenticNet 6: Ensemble application of symbolic and subsymbolic AI for sentiment analysis. In Proceedings of the 29th ACM International Conference on Information & Knowledge Management, Virtual, 19 October 2017; pp. 105–114.
23. Guan, X.; Cheng, Z.; He, X.; Zhang, Y.; Zhu, Z.; Peng, Q.; Chua, T. Attentive aspect modeling for review-aware recommendation. *ACM Trans. Inf. Syst. (TOIS)* **2019**, *37*, 1–27. [CrossRef]
24. Li, S.; Li, Y.; Zhao, W.; Ding, B.; Wen, J. Interpretable Aspect-Aware Capsule Network for Peer Review Based Citation Count Prediction. *ACM Trans. Inf. Syst. (TOIS)* **2021**, *40*, 1–29. [CrossRef]
25. Hong, X.; Guan, S.; Wong, P.; Xue, N.; Man, K.; Liu, D.; Li, Z. Lifelong machine learning-based quality analysis for product review. In Proceedings of the 2021 3rd International Conference on Advanced Information Science And System (AISS 2021), Sanya, China, 26–28 November 2021; pp. 1–5.
26. Jin, J.; Ji, P.; Kwong, C.K. What makes consumers unsatisfied with your products: Review analysis at a fine-grained level. *Eng. Appl. Artif. Intell.* **2016**, *47*, 38–48. [CrossRef]
27. Zirn, C.; Niepert, M.; Stuckenschmidt, H.; Strube, M. Fine-grained sentiment analysis with structural features. In Proceedings of the 5th International Joint Conference on Natural Language Processing, Chiang Mai, Thailand, 8–13 November 2011; pp. 336–344.

28. Scott, A.C.; Clancey, W.J.; Davis, R.; Shortliffe, E.H. *Explanation Capabilities of Production-Based Consultation Systems*; Technical Report; Stanford University, Department of Computer Science: Stanford, CA, USA, 1977.
29. Swartout, W.R. Explaining and justifying expert consulting programs. In *Computer-Assisted Medical Decision Making*; Springer: New York, NY, USA, 1985; pp. 254–271.
30. Liao, S.-H. Expert system methodologies and applications-a decade review from 1995 to 2004. *Expert Syst. Appl.* **2005**, *28*, 93–103. [CrossRef]
31. Guidotti, R.; Monreale, A.; Giannotti, F.; Pedreschi, D.; Ruggieri, S.; Turini, F. Factual and counterfactual explanations for black box decision making. *IEEE Intell. Syst.* **2019**, *34*, 14–23. [CrossRef]
32. Stepin, I.; Alonso, J.M.; Catala, A.; Pereira-Fariña, M. Generation and evaluation of factual and counterfactual explanations for decision trees and fuzzy rule-based classifiers. In Proceedings of the 2020 IEEE International Conference on Fuzzy Systems, Glasgow, UK, 19–24 July 2020; pp. 1–8.
33. Hendricks, L.A.; Hu, R.; Darrell, T.; Akata, Z. Generating counterfactual explanations with natural language. *arXiv* **2018**, arXiv:1806.09809.
34. Kenny, E.M.; Keane, M.T. On generating plausible counterfactual and semi-factual explanations for deep learning. *arXiv* **2020**, arXiv:2009.06399.
35. Lawrence, J.; Reed, C. Argument mining: A survey. *Comput. Linguist.* **2020**, *45*, 765–818. [CrossRef]
36. AlKhatib, K.; Ghosal, T.; Hou, Y.; de Waard, A.; Freitag, D. Argument mining for scholarly document processing: Taking stock and looking ahead. In Proceedings of the Second Workshop on Scholarly Document Processing, Mexico City, Mexico, 10 June 2021; pp. 56–65.
37. Galassi, A.; Lippi, M.; Torroni, P. Multi-task attentive residual networks for argument mining. *arXiv* **2021**, arXiv:2102.12227.
38. Trautmann, D.; Fromm, M.; Tresp, V.; Seidl, T.; Schütze, H. Relational and fine-grained argument mining. *Datenbank-Spektrum* **2020**, *20*, 99–105. [CrossRef]
39. Cortes, C.; Vapnik, V. Support-vector networks. *Mach. Learn.* **1995**, *20*, 273–297. [CrossRef]
40. Friedman, N.; Geiger, D.; Goldszmidt, M. Bayesian network classifiers. *Mach. Learn.* **1997**, *29*, 131–163. [CrossRef]
41. Clinchant, S.; Jung, K.W.; Nikoulina, V. On the use of bert for neural machine translation. *arXiv* **2019**, arXiv:1909.12744.
42. Yoo, S.; Jeong, O. An intelligent chatbot utilising bert model and knowledge graph. *J. Soc. e-Bus. Stud.* **2020**, *24*, 87–98.
43. Xue, N.; Niu, L.; Hong, X.; Li, Z.; Hoffaeller, L.; Popper, C. Deepsim: GPS spoofing detection on UAVs using satellite imagery matching. In Proceedings of the Annual Computer Security Applications Conference, Austin, TX, USA, 7–11 December 2020; pp. 304–319.
44. Li, Z.; Shao, H.; Niu, L.; Xue, N. Progressive learning algorithm for efficient person re-identification. In Proceedings of the 2020 25th International Conference on Pattern Recognition (ICPR), Milan, Italy, 10–15 January 2021; pp. 16–23.
45. Li, Z.; Shao, H.; Niu, L.; Xue, N. PLA: Progressive learning algorithm for efficient person re-identification. In *Multimedia Tools and Applications 2022*; Springer: Berlin, Germany, 2022; pp. 1–21.
46. Li, Z.; Cai, S.; Wang, X.; Niu, L.; Xue, N. Multiple object tracking with GRU association and kalman prediction. In Proceedings of the 2021 International Joint Conference on Neural Networks (IJCNN), Shenzhen, China, 18–22 July 2021.
47. Yin, X.; Li, W.; Li, Z.; Yi, L. Recognition of grape leaf diseases using MobileNetV3 and deep transfer learning. *Int. J. Agric. Biol. Eng.* **2022**, *15*, 184–194. [CrossRef]
48. Xue, N.; Niu, L.; Li, Z. Pedestrian detection with modified R-FCN. In Proceedings of the UAE Graduate Students Research Conference 2021, Abu Dhabi, United Arab Emirates, 27 June 2021.
49. Gai, J.; Xue, X.; Li, Z.; Zhang, L. Spectrum Sensing Method Based on Residual Cellular Network. *IEEE Access* **2022**, *10*, 61354–61365. [CrossRef]
50. Nadeau, D.; Sekine, S. A survey of named entity recognition and classification. *Lingvisticae Investig.* **2007**, *30*, 3–26. [CrossRef]
51. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. Roberta: A robustly optimised bert pretraining approach. *arXiv* **2019**, arXiv:1907.11692.
52. Manning, C.D.; Surdeanu, M.; Bauer, J.; Finkel, J.R.; Bethard, S.; McClosky, D. The stanford corenlp natural language processing toolkit. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, Baltimore, MD, USA, 23–24 June 2014.
53. LeCun, Y.; Bengio, Y. Convolutional networks for images, speech, and time series. *Handb. Brain Theory Neural Netw.* **1995**, *3361*, 255–258.
54. Zintgraf, L.M.; Cohen, T.S.; Adel, T.; Welling, M. Visualising deep neural network decisions: Prediction difference analysis. *arXiv* **2017**, arXiv:1702.04595.