An Unseen Features-Enriched Lifelong Machine Learning Framework

Nesar Ahmad Wasi(✉) and Muhammad Abulaish

Department of Computer Science, South Asian University, New Delhi 110068, India
nesarahmadwasi17@gmail.com, abulaish@sau.ac.in

Abstract. The dialect of a machine learning model is comprised of the features encountered during training. Nonetheless, as time passes, a deployed machine learning model may encounter certain features for the first time. In conventional machine learning approaches, newly observed features are typically discarded during testing data sample preprocessing. In lifelong machine learning, newly observed features may have appeared in the feature space of previously learned tasks; consequently, the knowledge associated with those features present in the knowledge base is incorporated to handle these features. However, there may be some features that have yet to appear in the knowledge base; lifelong machine learning also discards such features. Features that were not seen before are called *unseen features*. In this paper, we propose an enhanced lifelong machine learning framework for handling *unseen features* during the testing phase that incorporates *relative knowledge*. To extract *relative knowledge*, we retrieve semantically similar features using a language model. In addition, semantically similar features are examined in the knowledge base, and the knowledge of those present in the knowledge base is incorporated in order to deal with *unseen features*. Experiments conducted on the Amazon review dataset indicate that the proposed method outperforms three baselines and is competitive with state-of-the-art methods.

Keywords: Unseen Features · Lifelong Machine Learning · Continual Learning

1 Introduction

Data diversity is essential to generalize an unseen data sample using a machine learning model. The benefit of data diversity is that it enables the machine learning model to learn an accurate representation of the task's features. Lifelong Machine Learning (LML) is a process of continuous learning that retains knowledge acquired from previous learning tasks and uses this knowledge to learn incoming tasks. LML has a knowledge base that typically includes features that the model may not have seen for the current task, but there is a high likelihood that these features have appeared in previous tasks. Therefore, knowledge from the knowledge base is utilized to handle the unseen feature. However, there

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2023
O. Gervasi et al. (Eds.): ICCSA 2023, LNCS 13957, pp. 471–481, 2023.
https://doi.org/10.1007/978-3-031-36808-0_34

may be unseen features that neither the model nor the knowledge base has covered; such features are typically discarded during the preprocessing phase of the testing data sample. Because unseen features lack associated knowledge, it is difficult to incorporate them during classification.

In this paper, an unseen features-enriched lifelong machine learning (ULML) framework is proposed to address the problem of unseen features utilizing relative knowledge. We propose two methods for extracting *relative knowledge*: a synonym-based approach and a language model-based approach. In the synonym-based approach, the external dictionary's synonyms are extracted to extract *relative knowledge* for the unseen feature. The extracted synonyms are explored in the knowledge base, and the average weight associated with those features that are present in the knowledge base is assigned to the unseen feature. To handle unseen features in the language model-based approach, we identify semantically similar features to the unseen feature present in the language model itself [16].

This paper extends the approach proposed by [4] to the ULML framework. The method proposed in [4] extends Naïve Bayes to the LML setting. As regularization terms, two types of knowledge, namely domain-level and document-level knowledge, are incorporated. The training phase of our approach is identical to the one proposed by [4]. During the testing phase, however, unseen features are implemented based on *relative knowledge*. We have conducted extensive experiments on the Amazon review dataset. We compared the performance of the proposed approach to three baselines and one state-of-the-art method. Extensive empirical tests indicate that the proposed approach is better than three baseline methods and is comparable to the state-of-the-art method.

The rest of the paper is organized as follows. In Sect. 2, we have discussed related works. In Sect. 3, the proposed ULML framework is discussed. In Sect. 4, we discuss the experimental setup and results of the proposed approach. We also present performance, comparative analysis, and limitations of the proposed approach in Sect. 4. Finally, we conclude the paper in Sect. 5.

2 Related Works

The concept of Lifelong Machine Learning (LML) was first proposed by [21]. Though the concept of LML is similar to learning paradigms such as Transfer Learning (TL) [14], Multi-Task Learning (MTL) [27], and Online Learning [8], however, the core difference is that the knowledge transfer is not continues in the aforementioned paradigms. Thereafter, [20] used the idea of LML to formulate a binary classification problem for concept learning. [19] proposed an LML approach to extend the concept of MTL to the lifelong setting.

The approach proposed by [4] extended Naïve Bayes to the LML framework and utilized stochastic gradient descent to optimize domain-dependent and domain-independent knowledge in the Naïve Bayes. Further, they applied their approach to the sentiment classification task. Further, [24] proposed an LML approach to handle the difference between opinion and aspect words in

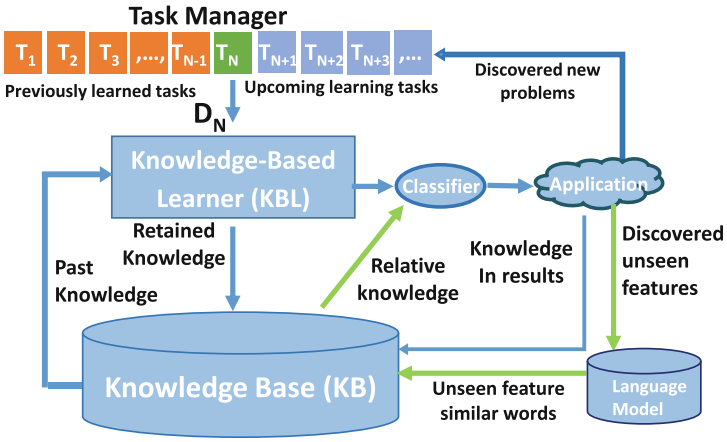

Fig. 1. The overall workflow of the unseen features enriched lifelong machine learning framework

an aspect-based sentiment classification task. In [22], the authors extended the work of [4] and proposed an LML approach for the sentiment classification task that can transfer knowledge to the future and previously learned tasks. In [23], a Bayes-enhanced deep learning approach is proposed that uses the generative parameters of Naïve Bayes to learn knowledge used in attention networks. Further, [23]'s approach is used for the task of sentiment classification.

In [10], the authors proposed a neural network-based continual learning approach for sentiment classification that is able to transfer knowledge learned from the previous tasks to the current task as well as it is able to enhance the performance of those tasks that are previously learned by incorporating knowledge learned from the current task. In [7], an iterative pruning approach is utilized for pruning the unwanted parameters in a deep learning network. By using pruning, it is able to free up space that can later be used to learn new tasks. They also adopted an uncertainty regularization based on the Bayesian framework while updating the weights associated with the previously learned tasks, which as a result, facilitate the learning in previously learned tasks to have positive knowledge transfer.

The significant difference between our approach and all aforementioned approaches is in utilizing unseen features. The idea of unseen features was first proposed by [25]. To the best of our knowledge, we are the first to incorporate unseen features in lifelong machine learning.

3 Unseen Features Enriched Lifelong Machine Learning Framework (ULML)

In this section, we propose a lifelong machine learning (LML) framework in which unseen features that appear for the first time during the testing phase

are utilized. LML is a continuous learning paradigm that aims to mimic how human beings learn. In the LML paradigm, a machine learning model has faced $n-1$ tasks; when it faces n^{th} task, it can leverage the knowledge learned from the previous $n-1$ tasks. An approach is called an LML approach if it is able to continuously learn, store, and extract new knowledge from the task it faces [3].

Figure 1 presents the overall workflow of the proposed framework. Starting from assigning Task T_N along with the data D_N, the task is assigned to the knowledge-based learner to learn new knowledge from data D_n and utilize knowledge in the knowledge base to train the classifier. The classifier is deployed to the retrospective application area. In LML, it can identify new problems, which can further be assigned as new tasks to be learned, and discover unseen features that can be learned with the help of a language model and knowledge base.

From Fig. 1, it can be observed that the ULML framework has four main components, namely, Task Manager (TM), Knowledge-Based-Learner (KBL), Knowledge Base (KB), and Language Model. TM manages the arrival of incoming tasks as well as manages the tasks previously learned by KBL. KBL learns and mines knowledge from the training data of the incoming tasks and stores the results in KB. When the classifier is employed in the application area, it may face some features that are not present in the features space of KB. For each unseen feature, the most similar features are extracted from language model. The extracted features are looked at in the KB to get knowledge of similar features from KB. Further, the average of the similar features found in KB is computed and assigned to the unseen feature, and the knowledge is termed as *relative knowledge*.

In this paper, we extend the approach proposed by [4] to the ULML framework. In the approach presented by [4], the Naïve Bayes is extended to the lifelong machine learning setting. Naïve Bayes is a probabilistic generative model [13] that uses Bayes Rule [1]. Bayes theorem was first applied to text classification by [12].

$$c_{pred} = \underset{c_k \in C}{\operatorname{argmax}} P(c_k|d) \tag{1}$$

In Naïve Bayes, in order to classify a document d to the corresponding class label $c \in C$, Eq. (1) is used. In Eq. (1), each document d can also be represented using set of features $f_1, f_2, f_3, \cdots, f_n$. The base for Naïve Bayes is the Bayes theorem, which calculates the conditional probability of each feature given class c_k.

$$P(f|c_k) = \frac{F_{c_k,f} + \lambda}{\sum_{v=1}^{|V|} F_{c_k,f_v} + \lambda|V|} \tag{2}$$

In Eq. (2), $F_{c_k,f}$ is the frequency of feature f_i appeared in class c_k. The main parameter of Naïve Bayes is $F_{c_k,f}$. λ is the smoothing parameter. $|V|$ denotes the number of features present in the vocabulary. In Naïve Bayes, each feature f_i in a document d is assumed to be independent of each other. The Naïve Bayes classifier for a document d is defined as (3).

$$P(c_p|d) = \frac{P(c_p)\prod_{f_i \in |V|} P(f_i|c)}{\sum_{b=1}^{2} P(c_b)\prod_{f_i \in |V|} P(f_i|c)} \tag{3}$$

For correct classification of the document d in a binary classification setting, the $P(c_p|d) = 1$ and $P(c_q|d) = 0$, where c_p is the label of the positive class, and c_q is the label of the negative class.

$$\underset{c_k \in C}{\operatorname{argmax}}\, P(c_p|d) - P(c_q|d) \tag{4}$$

In order to solve the optimization problem in Eq. (4), Stochastic Gradient Descent (SGD) is used. SGD updates the expected frequency of feature f_i. To differentiate the expected frequency from the actual frequency of the feature f_i, i.e., $F_{c_k,f}$, the expected frequency of feature f_i is denoted by $X_{c_k,f}$. The starting point for SGD is $F^t_{c_k,f} + F^{KB}_{c_k,f}$, where $F^t_{c_k,f}$ is the frequency of feature f in target domain t, and $F^{KB}_{c_k,f}$ is the frequency of feature f in KB.

During the training phase of the proposed approach, two types of knowledge – domain-dependent and domain-level knowledge are incorporated in the form of regularization terms. For both types of knowledge, two vocabularies, V_t, and V_{KB} are constructed for both types of knowledge. Domain-dependent knowledge ensures that features appear in V_t are those features that are highly reliable in the target domain, i.e., $\frac{P(f_i|+)}{P(f_i|-)} \geq \sigma$ or $\frac{P(f_i|-)}{P(f_i|+)} \geq \sigma$, number of documents is denoted by σ.

$$\frac{1}{2}\sum_{f \in V_t}\left(\left(X_{+,f} - F^t_{+,f}\right)^2 + \left(X_{-,f} - F^t_{-,f}\right)^2\right) \tag{5}$$

Features that appeared in more number of previous tasks/domains compared to those features that are highly specific to some domains are more reliable. Domain frequency of each feature is recorded, and a list of features is constructed for those features that appear in a substantial number of domains, i.e., $R^{KB}_{+,f} \geq \tau$ or $R^{KB}_{-,f} \geq \tau$, the τ denoted number of domains. V_d denotes the list of domain-level knowledge.

$$\begin{aligned}&\tfrac{1}{2}\alpha\textstyle\sum_{f \in V_d}\left(\left(X_{+,f} - M_f \times \left(F^t_{+,f} + F^{KB}_{+,f}\right)\right)^2\right.\\&\left.+\left(X_{-,f} - M_f \times \left(F^t_{-,f} + F^{KB}_{-,f}\right)\right)^2\right)\end{aligned} \tag{6}$$

In Eq. (6), M_f is equal to $R^{KB}_{+,f} / \left(R^{KB}_{+,f} + R^{KB}_{-,f}\right)$. Equations (5) and (6) are incorporated along with Eq. (4) as penalty terms to leverage document-level and domain-level knowledge. Further, SGD is employed to train the machine learning model.

Algorithm 1: ULML for a document d_p of target domain in testing phase.

Input : Document d_p having features $\{f_1, f_2, \ldots, f_n\}$, $\mathcal{F}_{kb}$ vocabulary of features of all domains, parametric weight $X_{\pm,f_i}$ of all features.
Output: Predicted sentiment label of the document d_p.

for *each feature* $f_i \in d_p$ **do**
  if f_i *belongs to* $\mathcal{F}_{kb}$ **then**
    $F_{+,f_i} \leftarrow X_{+,f_i}$
    $F_{-,f_i} \leftarrow X_{-,f_i}$
  else
    $F_{+,f_i} \leftarrow \hat{P}(f_i|+)$ #Equation (8)
    $F_{-,f_i} \leftarrow \hat{P}(f_i|-)$ #Equation (8)
  end
end
return $\text{argmax}_{c_k \in \{+,-\}} P(c_k|d_p)$ #Equation (1)

3.1 Relative Knowledge Extraction

During the testing phase, when document $d_p = \{f_1, f_2, f_3, \ldots, f_{|d_p|}\}$ is fed to a classifier to predict its class label, there may exist some features which are not present in the features space of all previously learned domains as well as features space of the target domain stored in the knowledge base, i.e.,

$$\mathcal{F}_u = \forall f_i \notin \mathcal{F}_{kb} \quad where,\ 1 \leq i \leq |d_p| \tag{7}$$

In Eq. (7), $\mathcal{F}_{kb}$ denotes features space of the knowledge base, and $\mathcal{F}_u$ denotes unseen features. In order to detect $\mathcal{F}_u$, Eq. (7), is used. As $\mathcal{F}_u$ are those features that the classier has not seen before and do not carry direct knowledge with them; therefore, it is hard to utilize. In order to utilize such features, language models, such as Word2Vec [11], GloVe [16], BERT [6], Fastext [9], ELMO [17], XLNet [26], GPT [18] can be instrumental. Language models can be used to extract relative knowledge of $\mathcal{F}_u$ features, as language models are vector-spaced representations of words that preserve contexts and semantics.

To extract relative knowledge associated with the features identified as unseen features $\mathcal{F}_u$, those features that are semantically similar to the unseen feature are extracted from the language model. Because language models are high-dimensional vector representations, a multi-dimensional data structure is used to index all features present in the language model. K-dimensional tree [2], is used as the multi-dimensional data structure. In order to extract semantically similar features for the unseen feature, the nearest neighbor approach [5] is used. Further, the extracted features denoted by $\mathcal{S}_u$ are looked in KB, i.e., $\exists \mathcal{S}_u \in KB$. The list of semantically similar features discovered in KB is denoted by $\mathcal{R}_u$.

The average weight associated with features $\mathcal{R}_u$ is calculated and assigned to the unseen feature using Eq. (8).

$$\hat{P}(F_u|c_k) = \frac{\sum_{r=1}^{|\mathcal{R}_u|} P(\mathcal{R}_u^r|c_k)}{|\mathcal{R}_u|} \tag{8}$$

Algorithm 1 presents the procedure of ULML to handle unseen features during the testing phase. In Algorithm 1, in order to detect the polarity of the document d_p during the testing phase, we need $\mathcal{F}_{kb}$, i.e., the vocabulary of features that appeared in all previous domains to check whether the feature $f_i \in d_p$ is an unseen feature. If feature f_i already appears in previous domains, the parametric weight $X_{\pm,f_i}$ is assigned to the feature f_i. Else f_i will be assigned relative knowledge extract using Eq. (8).

4 Experimental Setup and Results

In order to perform experiments, we use the same dataset[1] as used in the work of [4]. This dataset contains reviews from 20 different types of products. It is extracted from Amazon.com. This dataset has 1000 reviews for each type of product. Each review present in the dataset is labeled as *positive*, *negative*, or *neutral*. Those reviews with a rating greater than 3 are labeled as *positive*. Those reviews with a rating less than 3 are labeled as *negative*, and those reviews with a rating of exactly 3 are labeled as *neutral*. As the setting of the problem is a binary classification problem, the neutral reviews are discarded from the experiments. The complete statistics of the dataset are presented in Table 1. As per Table 1, the dataset is skewed towards the positive class. Therefore, the minority class is the negative class which is very hard to classify.

In each domain, we randomly partition data into train and test partitions, and the ratio of both train and test in each domain is set to 80% : 20%, respectively. We have used the 5-fold cross-validation strategy for performance evaluation. In our experiments, we have used uni-gram features. In order to handle negation words, we followed [15]'s approach. To handle negation during the preprocessing phase, we prefix the token *"Not_"* to each word that appears after a logical negation word, i.e., *n't*, *not*, *no*, and *never* in the document until the next punctuation mark appears. We have used the default parameters for all baselines or as specified in the original paper. In order to extract relative knowledge in ULML-G, we have used the pre-trained GloVe [16] language model. We extract the top 35 similar features from the language model. We experimented with different numbers of top similar features. We got the best results with the top 35. In ULML-S, we have used the synonyms of the unseen feature. To extract synonyms, we have used *GroupDocs.search*[2] API. Further, knowledge associated with those synonyms that appeared in the knowledge base is incorporated as relative knowledge.

[1] https://www.cs.uic.edu/~zchen/downloads/ACL2015-Chen-Datasets.zip.
[2] https://docs.groupdocs.com/search/java/synonym-search.

Table 1. The proportion of negative instances in each domain of the dataset.

Domain	Proportion	Domain	Proportion
Alarm Clock	30.51	Baby	16.45
Bag	11.97	Cable Modem	12.53
Dumbbell	16.04	Flashlight	11.69
Gloves	19.50	GPS	13.76
Graphics Card	14.58	Headphone	20.99
Home Theater System	28.84	Jewelry	12.21
Keyboard	22.66	Magazine Subscriptions	26.88
Movies TV	10.86	Projector	20.24
Rice Cooker	18.64	Sandal	12.11
Vacuum	22.07	Video Games	20.93

Table 2. F1-Score: Performance evaluation results of the proposed approach and all baselines to identify the negative (minority) class which is harder to identify.

Domain	NB-S	NB-T	NB-ST	LSC	ULML-S	ULML-G
Alarm Clock	45.722	64.71	64.71	78.56	79.48	78.45
Baby	39.34	46.51	41.03	62.76	61.95	62.47
Bag	34.29	62.22	63.16	66.28	66.26	66.59
Gloves	30.30	57.14	57.78	74.54	50.70	74.66
Headphone	52.27	56.66	52.78	66.14	65.93	65.48
Home Theater System	76.09	71.73	82.00	81.64	81.73	81.27
Magazine Subscriptions	48.65	79.23	64.71	67.95	67.98	69.05
Projector	65.57	61.54	71.43	74.63	74.08	74.06
Rice Cooker	60.71	69.84	69.84	69.82	69.82	69.60
Sandal	50.00	45.16	50.00	54.01	53.75	53.75
Average over top 10 domains	50.29	61.47	61.74	69.63	67.17	69.54

4.1 Performance and Comparative Analysis

In order to evaluate the performance of the proposed approach, we compare it with three variants of Naïve Bayes (NB) and one state-of-the-art approach, i.e., lifelong learning for sentiment classification (LSC) [4]. While performing experiments, each domain is considered the target domain, while the rest 19 domains are considered non-target domains. As NB is a classification approach that works on a single domain at once, it is fed with three types of data to have a fair comparison. *NB-S* is trained using data from non-target domains. *NB-T* is a traditional supervised learning model. *NB-ST* is trained using data from both target and non-target domains. All the above approaches are tested using data from the target domain. *LSC* is the state-of-the-art approach that we used as our primary baseline. *NB-T* do not use data from other domains(tasks); therefore,

it can be regarded as a non-lifelong machine learning approach. Since *NB-S* and *NB-ST* incorporate data from other domains (tasks), these approaches can be regarded as basic lifelong machine learning approaches.

From Table 2, it can be observed that in *NB-ST*, simply incorporating data from other domains during training is an advantageous task. In Table 2, it can be seen that *NB-S* and *NB-T* are inferior to *NB-ST*. However, *NB-ST* is inferior to our proposed approaches, i.e., *ULML-G* and *ULML-S*. In *ULML-G*, we have used the knowledge extracted from the language model for the unseen feature. In *ULML-S*, we have used the knowledge extracted from synonyms of unseen features present in KB. In both proposed approaches (ULML-G and ULML-S), when we used relative knowledge associated with the unseen feature from GloVe, it performed better compared to the synonym-based approach. The prime reason that the proposed approach is not able to get competitive results in some domains compared to the state-of-the-art approach is the lack of correct relative knowledge in the knowledge base. Because unseen words appear for the first time during the testing phase, therefore, are very hard to handle.

4.2 Limitations

When a machine-learning model is deployed in an application domain, a testing data sample cannot be directly fed to the machine-learning model (trained model) for prediction. To prepare the testing sample for the trained model, it must go through a preprocessing phase. A testing data sample may contain features that the model has never seen before (traditionally, the unseen features are discarded during this phase); our approach is able to extract relative knowledge with the assistance of a language model and knowledge base, allowing us to incorporate the unseen features. As these features emerge for the first time, a lack of accurate knowledge can hinder prediction. Suppose there are insufficient semantically similar features to an unseen feature in the knowledge base. In that case, the relative knowledge assigned to the unseen feature will be incorrect and detrimental to the classification task. A further limitation of the proposed method is that we discard unseen features absent from the language model. However, such features can be managed more efficiently. We believe that addressing the aforementioned issue is beyond the scope of this paper because it involves the concept of the language model's out-of-vocabulary problem.

5 Conclusion

In this paper, we proposed a framework for lifelong machine learning that incorporates features that emerge for the first time when a machine learning model is deployed. First, unseen features are identified by comparing the features space of the incoming document to the features space of tasks contained within the knowledge base. We proposed two approaches, ULML-S and ULML-G, for handling unseen features. In ULML-S, we have utilized knowledge associated with synonyms of unseen features. In ULML-G, similar words are retrieved from the

language model for an unseen feature. In addition, knowledge associated with similar words is extracted from the knowledge base. The retrieved knowledge is assigned to the unseen features. We conducted exhaustive experiments on the Amazon review dataset. We compared the performance of the proposed method to three baselines and one state-of-the-art method. The performance evaluation results indicate that the proposed method outperforms the three baseline methods and is competitive with the state-of-the-art method.

References

1. Bayes, T.: An essay towards solving a problem in the doctrine of chances. Philos. Trans. R. Soc. Lond. (53), 370–418 (1763)
2. Bentley, J.L.: Multidimensional binary search trees used for associative searching. Commun. ACM **18**(9), 509–517 (1975). https://doi.org/10.1145/361002.361007
3. Chen, Z., Liu, B.: Lifelong Machine Learning, 2nd edn. Morgan & Claypool Publishers, San Rafael (2018)
4. 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), pp. 750–756. Association for Computational Linguistics (2015). https://doi.org/10.3115/v1/P15-2123
5. Cover, T.: Estimation by the nearest neighbor rule. IEEE Trans. Inf. Theory **14**(1), 50–55 (1968). https://doi.org/10.1109/TIT.1968.1054098
6. Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, Minneapolis, MN, USA, 2–7 June, pp. 4171–4186. Association for Computational Linguistics (2019). https://doi.org/10.18653/v1/n19-1423
7. Geng, B., Yang, M., Yuan, F., Wang, S., Ao, X., Xu, R.: Iterative network pruning with uncertainty regularization for lifelong sentiment classification. In: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2021, pp. 1229–1238. Association for Computing Machinery, New York (2021). https://doi.org/10.1145/3404835.3462902
8. Hoi, S.C., Sahoo, D., Lu, J., Zhao, P.: Online learning: a comprehensive survey. Neurocomputing **459**, 249–289 (2021). https://doi.org/10.1016/j.neucom.2021.04.112
9. Joulin, A., Grave, E., Bojanowski, P., Douze, M., Jégou, H., Mikolov, T.: Fasttext.zip: compressing text classification models. CoRR abs/1612.03651 (2016). https://doi.org/10.48550/arXiv.1612.03651
10. Ke, Z., Liu, B., Wang, H., Shu, L.: Continual learning with knowledge transfer for sentiment classification. In: Hutter, F., Kersting, K., Lijffijt, J., Valera, I. (eds.) ECML PKDD 2020. LNCS (LNAI), vol. 12459, pp. 683–698. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-67664-3_41
11. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. In: Proceedings of the 1st International Conference on Learning Representations, ICLR, Scottsdale, Arizona, USA, 2–4 May, Workshop Track Proceedings (2013). https://doi.org/10.48550/arXiv.1301.3781
12. Mosteller, F., Wallace, D.L.: The Federalist: Inference and Disputed Authorship. Addison-Wesley, Boston (1964)

13. Nigam, K., McCallum, A., Thrun, S., Mitchell, T.: Learning to classify text from labeled and unlabeled documents. In: Proceedings of the 15th AAAI Conference on Artificial Intelligence, pp. 792–799. AAAI Press (1998). https://doi.org/10.5555/295240.295806
14. Pan, S.J., Yang, Q.: A survey on transfer learning. IEEE Trans. Knowl. Data Eng. **22**(10), 1345–1359 (2010). https://doi.org/10.1109/TKDE.2009.191
15. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up? Sentiment classification using machine learning techniques. In: Proceedings of the 7th Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 79–86. Association for Computational Linguistics (2002). https://doi.org/10.3115/1118693.1118704
16. Pennington, J., Socher, R., Manning, C.D.: GloVe: global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, 25–29 October 2014, Doha, Qatar, pp. 1532–1543. ACL (2014). https://doi.org/10.3115/v1/D14-1162
17. Peters, M.E., et al.: Deep contextualized word representations. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), New Orleans, Louisiana, pp. 2227–2237. Association for Computational Linguistics (2018). https://doi.org/10.18653/v1/N18-1202
18. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I.: Improving language understanding by generative pre-training (2018)
19. Ruvolo, P., Eaton, E.: Ella: an efficient lifelong learning algorithm. In: Proceedings of the 30th International Conference on Machine Learning, pp. 507–515. JMLR.org (2013)
20. Thrun, S.: Is learning the n-th thing any easier than learning the first? In: Proceedings of the Conference on Advances in Neural Information Processing Systems, pp. 640–646. The MIT Press (1996)
21. Thrun, S., Mitchell, T.M.: Lifelong robot learning. Robot. Auton. Syst. **15**(1), 25–46 (1995)
22. Wang, H., Liu, B., Wang, S., Ma, N., Yang, Y.: Forward and backward knowledge transfer for sentiment classification. In: Proceedings of The 11th Asian Conference on Machine Learning, ACML, pp. 457–472. PMLR (2019). https://doi.org/10.48550/arXiv.1906.0350
23. Wang, H., Wang, S., Mazumder, S., Liu, B., Yang, Y., Li, T.: Bayes-enhanced lifelong attention networks for sentiment classification. In: Proceedings of the 28th International Conference on Computational Linguistics, pp. 580–591. International Committee on Computational Linguistics (2020). https://doi.org/10.18653/v1/2020.coling-main.50
24. Wang, S., Zhou, M., Mazumder, S., Liu, B., Chang, Y.: Disentangling aspect and opinion words in target-based sentiment analysis using lifelong learning. CoRR abs/1802.05818, pp. 1–7 (2018). https://doi.org/10.48550/arXiv.1802.05818
25. Wasi, N.A., Abulaish, M.: An unseen features enhanced text classification approach. In: The Procedings of the International Joint Conference on Neural Networks (IJCNN), Queensland, Australia, p. 8 (2023)
26. Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R.R., Le, Q.V.: Xlnet: generalized autoregressive pretraining for language understanding. In: Proceedings of the 32th Advances in Neural Information Processing Systems, vol. 32. Curran Associates, Inc. (2019). https://doi.org/10.5555/3454287.3454804
27. Zhang, Y., Yang, Q.: A survey on multi-task learning. IEEE Trans. Knowl. Data Eng. **34**(12), 5586–5609 (2022). https://doi.org/10.1109/TKDE.2021.3070203