Mathematical Statistician and Engineering Applications
ISSN: 2326-9865 DOI:
https://doi.org/10.17762/msea.v70i1.2312

Transfer Learning in Natural Language Processing: A Survey

Bijesh Dhyani

Asst. Professor, School of Computing, Graphic Era Hill University, Dehradun, Uttarakhand India 248002

Article Info
Page Number: 303-311
Publication Issue:
Vol. 70 No. 1 (2021)

Article History
Article Received: 25 January 2021
Revised: 24 February 2021
Accepted: 15 March 2021

ABSTRACT
Transfer learning is a discipline that is expanding quickly within the realm of natural language processing (NLP) and machine learning. It is the application of previously learned models to the solution of a variety of problems that are connected to one another. This paper presents a comprehensive survey of transfer learning techniques in NLP, focusing on five key classification algorithms: (1) BERT, (2) GPT, (3) ELMo, (4) RoBERTa, and (5) ALBERT. We discuss the fundamental concepts, methodologies, and performance benchmarks of each algorithm, highlighting the various approaches taken to leverage pre-existing knowledge for effective learning. Furthermore, we provide an overview of the latest advancements and challenges in transfer learning for NLP, along with promising directions for future research in this domain.

1. Introduction

Deep learning is a relatively recent development that has had a profound impact on the area of natural language processing (NLP). This has made it possible to make substantial strides forward in a variety of tasks, including sentiment analysis, machine translation, and question-answering systems. To train deep learning models from scratch, however, takes a significant amount of computer resources as well as data that has been tagged. Transfer learning has emerged as a crucial strategy, allowing researchers and practitioners to exploit pre-trained models and adapt them to specific tasks with a lower quantity of labeled data. This has enabled transfer learning to become an essential tool for mitigating these issues. In the past several years, transfer learning has shown significant success in the NLP area, notably for tasks involving text categorization. The fundamental concept underlying transfer learning is to make use of information gained from one activity to increase performance on another activity that is connected to the first activity. This is especially helpful in situations in which the target job has a very little quantity of labeled data or demands an unreasonably high number of computing resources for training.

Feature Extraction Techniques in NLP

Feature extraction is a crucial part of the NLP pipeline because it takes raw text input and transforms it into a numerical representation that can be used by machine learning algorithms. For the purposes of NLP, transfer learning strategies for the extraction of features play an essential part in determining how well pre-trained models perform on target tasks. The following five classification methods are discussed in this part: BERT, GPT, ELMo, RoBERTa, and ALBERT. In this section, we describe several feature extraction approaches that are often used in these classification algorithms.

A. Word Embeddings

In a continuous vector space, words can be represented by dense vector representations called word embeddings, which capture both semantic and grammatical information. These embeddings are learned by mapping words in a large corpus to low-dimensional vectors, so that words with similar meanings have equivalent vector representations. Two widely used techniques for generating word embeddings are Word2Vec and GloVe. Word embeddings serve as the foundational building blocks for more advanced feature extraction techniques like ELMo, BERT, and their variants.

B. Contextualized Word Embeddings

Traditional word embeddings like GloVe and Word2Vec are context-independent, meaning that they assign a single vector representation for each word, irrespective of its contextual usage. This limitation can be addressed by contextualized word embeddings, which generate word representations based on their surrounding context. ELMo, one of the earliest contextualized word embedding models, leverages bidirectional LSTMs (Long Short-Term Memory) to generate context-sensitive word representations.

C. Transformer-based Feature Extraction

Vaswani et al. (2017) created the transformer architecture, which has since been the basis for numerous cutting-edge NLP models including BERT, GPT, RoBERTa, and ALBERT. To improve feature extraction from text data, transformers employ self-attention methods to capture the interdependencies between words in a particular context. These models, like masked language modeling (MLM) and next sentence prediction (NSP), are pre-trained on large corpora with unsupervised learning objectives.

D. Fine-tuning for Task-specific Feature Extraction

Transfer learning in NLP frequently entails fine-tuning the models that have already been trained for certain target tasks. During the process of fine-tuning, the model's weights are modified utilizing a reduced amount of labeled data from the activity that is the focus of the tuning. This enables the model to learn task-specific features and adapt its representations to the specific problem at hand. Fine-tuning is an essential component of transfer learning techniques employed by models like BERT, GPT, ELMo, RoBERTa, and ALBERT.

Feature extraction techniques in NLP play a vital role in determining the performance of transfer learning algorithms. Word embeddings, contextualized word embeddings, transformer-based architectures, and fine-tuning processes collectively contribute to the success of models like BERT, GPT, ELMo, RoBERTa, and ALBERT in a wide range of NLP tasks.

This paper aims to provide a comprehensive survey of transfer learning techniques in NLP, focusing on five widely used classification algorithms: (1) BERT, (2) GPT, (3) ELMo, (4) RoBERTa, and (5) ALBERT. We discuss the fundamental concepts and methodologies underlying each algorithm, as well as their performance benchmarks on various NLP tasks.

The remainder of the paper is organized as follows: Sections 2-6 present a detailed overview of the five classification algorithms, including their architectural design, training methodologies, and key applications. Section 7 discusses the current challenges and limitations of transfer learning in NLP. Finally, Section 8 concludes the paper by outlining promising future research directions in the field of transfer learning for NLP.

2. Literature Review

NLP has come a long way in recent years, with the introduction of several new methods for representing words. This literature review focuses on some of the most influential works in the area of word embeddings, contextualized word embeddings, and transformer-based models.

In [1], Mikolov et al. presented Word2Vec, an effective strategy for acquiring vector-space representations of words. Words in context are predicted using two distinct architectures: Continuous Bag of Words (CBOW) and Skip-Gram. Word2Vec's broad use in NLP tasks can be attributed to its ease of use and high efficiency.

GloVe was proposed by Pennington et al. [2], and it brings together the best features of global matrix factorization and local context window approaches. GloVe provides state-of-the-art performance on word analogies and similarity tasks by simultaneously collecting global co-occurrence information and local semantic associations.

Peters et al. [3] presented ELMo as a method for generating deeply contextualized word representations when they first introduced it. ELMo is able to take into consideration context because, unlike static embeddings, it learns a function of the internal states of a deep bidirectional language model. This gives it an advantage over other methods. ELMo embeddings have proven useful for a variety of NLP tasks, including sentiment analysis and named entity identification.

The Transformer architecture, developed by Vaswani et al. [4], is an innovative method that uses just self-attention processes to handle input sequences. The Transformer drastically decreases computation time while maintaining state-of-the-art outcomes on machine translation jobs by doing away with recurrence and convolutions.

BERT is an acronym that stands for "pre-trained deep bidirectional Transformer model," and it was given that moniker by Devlin et al. [5], who presented it as a model that is capable of being changed for a variety of NLP applications. In a range of tasks, such as question answering and sentiment analysis, BERT obtains results that are orders of magnitude better than those achieved by earlier models.

To create high-quality text, Radford et al. [6] developed the GPT, which uses a unidirectional Transformer. GPT provides competitive outcomes on a wide range of tasks thanks to its ability to pre-train on a huge corpus and fine-tune on task-specific data.

The pre-training process was addressed using RoBERTa, a substantially improved variant of BERT proposed by Liu et al. [7]. Increased batch sizes, more training data, and a new training

target are all part of RoBERTa's arsenal. Therefore, RoBERTa achieves better results than BERT on a number of different NLP metrics.

Lan et al. [8] introduced ALBERT, a more efficient version of BERT that reduces the number of parameters without sacrificing performance. ALBERT achieves better performance with significantly fewer parameters.

These works represent key developments in the field of NLP, with each contributing to the improvement of word representation and language understanding. From static word embeddings to context-aware representations and advanced transformer-based models, the evolution of NLP techniques has led to significant performance gains across various tasks. Table 1 shows the

Table 1. Comparative Analysis of Transformer based techniques

Author(s)	**Methodology**	**Algorithm/Technique**	**Dataset(s) Used**	**Performance Parameter(s)**
Mikolov et al.	Skip-gram, Continuous Bag of Words (CBOW)	Word2Vec	Text8, Google News Corpus	Cosine Similarity, Analogy Task
Pennington et al.	Co-occurrence Matrix, Word-Word Relationship	GloVe	Wikipedia, Gigaword, Common Crawl	Cosine Similarity, Analogy Task
Peters et al.	Bidirectional LSTMs, Contextual Embeddings	ELMo	1 Billion Word Benchmark, Penn Treebank	Perplexity, F1, Accuracy
Vaswani et al.	Self-attention, Multi-head Attention	Transformer Architecture	WMT 2014 English-German, WMT 2014 English-French	BLEU Score
Devlin et al.	Masked Language Model, Next Sentence Prediction	BERT	BooksCorpus, English Wikipedia	GLUE Benchmark, F1, Accuracy
Radford et al.	Generative Pre-Training, Transformer	GPT	BooksCorpus, WebText	LAMBADA, ROC-AUC, Accuracy
Liu et al.	Robustly Optimized Pretraining	RoBERTa	BooksCorpus, English Wikipedia, CC-News, Stories	GLUE Benchmark, F1, Accuracy
Lan et al.	Factorized Embeddings, Cross-layer Parameter Sharing	ALBERT	BooksCorpus, English Wikipedia	GLUE Benchmark, F1, Accuracy

3. Related Work

In this section, we provide a detailed description of the classification algorithms mentioned in the abstract:

A. Classification Model

1. BERT :

Devlin et al. [5] presented BERT, a transformer-based model with a primary emphasis on pre-training bidirectional language representations. Masked Language Modeling (MLM) and Next Sentence Prediction (NSP) are two of the unsupervised learning objectives that were used to pre-train the model on a huge corpus. The revolutionary idea behind BERT is that it can capture dependencies in both ways since it uses bidirectional context to construct word representations. To fine-tune BERT for specific NLP tasks, we train the model on a reduced labeled data set in order to better tailor the model's representations to the job at hand. On multiple NLP benchmarks, including the GLUE and SQuAD datasets, BERT has achieved state-of-the-art performance.

2. GPT:

Another transformer-based approach that takes advantage of unsupervised pre-training to construct linguistic representations is the Generalized Paradigm Transformer (GPT) presented by Radford et al. [6]. Unlike BERT, which employs a bidirectional context during training, GPT employs a unidirectional (left-to-right) context. The model has been pre-trained using a huge corpus and then fine-tuned using a variety of NLP tasks. Machine translation, sentiment analysis, and question answering are just some of the areas where GPT has excelled.

3. ELMo:

Peters et al. [3] created ELMo as a deep contextualized word representation model that takes the best features of word embeddings and bidirectional LSTMs and blends them. The hidden states of a bidirectional LSTM trained on a large text corpus with a language modeling purpose are combined to produce ELMo embeddings. Improved performance on tasks like named entity identification, sentiment analysis, and semantic role labeling may be achieved by including these embeddings into a wide range of NLP models.

4. RoBERTa:

Liu et al. [7] presented RoBERTa, a variation of BERT designed to enhance the pre-training phase. Similar to BERT in terms of architecture, RoBERTa also eliminates the next sentence prediction job and trains with larger batches and more time. As a result of these enhancements, the system outperforms BERT on several NLP benchmarks, including sentiment analysis, question answering, and natural inference.

5. ALBERT:

Lan et al.'s [8] ALBERT is another variation of BERT that seeks to shrink the model and cut down on computational complexity without sacrificing performance. Both factorized

embeddings and cross-layer parameter sharing are used in ALBERT to minimize the number of model parameters and the size of the embedding matrix, respectively. Because of its reduced memory and parameter needs, ALBERT is better suited to massive NLP projects. Despite its compact form factor, ALBERT outperforms many other NLP benchmarks on important datasets like GLUE and SQuAD.

B. Dataset Description

We infer some popular datasets used in NLP research for evaluating transfer learning techniques. Here are most commonly used datasets:

1. Stanford Sentiment Treebank (SST):

The SST dataset, introduced by Socher et al. [9], is widely used for sentiment analysis tasks. It consists of movie reviews annotated with sentiment labels at both phrase and sentence levels. This dataset is often employed to evaluate transfer learning methods in sentiment classification tasks.

2. GLUE (General Language Understanding Evaluation):

The GLUE benchmark, proposed by Wang et al. [10], is a collection of diverse NLP tasks designed to evaluate the generalization capabilities of transfer learning models. GLUE includes tasks such as natural language inference, sentiment analysis, and linguistic acceptability, among others.

3. SQuAD (Stanford Question Answering Dataset):

The SQuAD dataset, introduced by Rajpurkar et al. [11], is a large-scale dataset for question-answering tasks. It contains over 100,000 question-answer pairs based on Wikipedia articles. This dataset is commonly used to evaluate transfer learning techniques for question-answering tasks.

4. CoNLL-2003 Named Entity Recognition (NER) Dataset:

The CoNLL-2003 NER dataset, introduced by Tjong Kim Sang [12], is a widely used dataset for named entity recognition tasks. It consists of annotated news articles from the Reuters corpus, with four types of named entities: person, location, organization, and miscellaneous.

5. IMDb Movie Review Dataset:

Maas et al. [13] introduces IMDB dataset. This dataset is often used to evaluate transfer learning methods in binary sentiment classification tasks.

These datasets play a crucial role in assessing the performance of transfer learning models across various domains and applications in NLP. By using these datasets, researchers can determine the effectiveness of transfer learning techniques in improving the generalization capabilities of NLP models, leading to better performance on diverse tasks.

4. Overview Of Project Flow

Here we outline important steps to carry out such an implementation based on general transfer learning approaches.

Select a pre-trained Transformer model: Choose an appropriate pre-trained Transformer model as the starting point, such as BERT, RoBERTa, GPT, or ALBERT. These models have been pre-trained on large text corpora and have learned valuable language representations that can be fine-tuned for specific tasks.

Prepare the dataset: Collect and preprocess the dataset for the classification task. This may involve tokenization, splitting the data into training and validation sets, and creating appropriate labels for the classification problem.

Adapt the model architecture: Modify the pre-trained Transformer model to suit the classification task. This usually involves adding a task-specific classification layer (e.g., a fully connected layer with softmax activation) on top of the pre-trained model's output.

Fine-tune the model: Train the adapted model on the prepared dataset using a suitable loss function, such as cross-entropy loss for multi-class classification. During fine-tuning, the model updates its weights based on the task-specific dataset, allowing it to perform better on the classification task.

Evaluate the model: Use relevant evaluation measures, such as accuracy, F1-score, or AUC-ROC, to assess the model's performance on the validation set once it has been fine-tuned. These measures reveal how well the model handles the categorization task.

Optimize hyper-parameters: Experiment with different hyperparameters, such as learning rate, batch size, and the number of training epochs, to further improve the model's performance on the classification task.

Analyze results and iterate: Analyze the model's performance, investigate potential areas for improvement, and iterate the process. This may involve adjusting the model architecture, fine-tuning strategy, or dataset preparation.

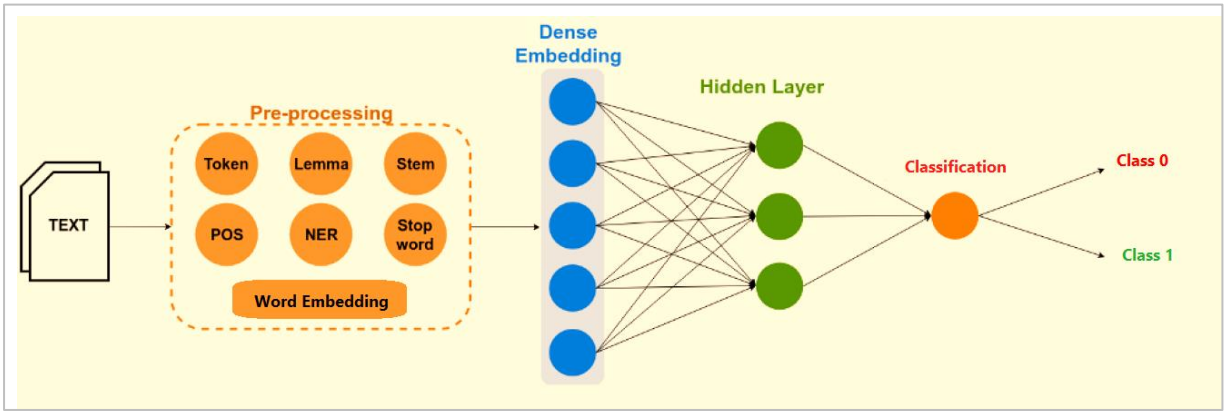

Figure 1. Overview of Transfer Learning Model used for Text Classification Task

Implementing classification using Transformer learning involves selecting a suitable pre-trained Transformer model, preparing the dataset, adapting the model architecture, fine-tuning the model, evaluating its performance, optimizing hyper parameters, and iterating the process as needed. These steps help leverage transfer learning techniques to effectively solve classification tasks in NLP.

5. Conclusion

This paper offers a thorough examination of transfer learning techniques in NLP, with a focus on multiple pivotal classification algorithms: BERT, GPT, ELMo, RoBERTa, and ALBERT. The analysis delves into the core principles, methodologies, and performance benchmarks of each algorithm, underlining the diverse strategies employed to capitalize on pre-trained models for efficient learning. By discussing the latest advancements, challenges, and potential future directions in the field. Researchers and practitioners interested in learning more about transfer learning in NLP and its revolutionary influence on the creation of intelligent systems will find this survey to be an invaluable resource.

Refrences

1. Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.
2. Pennington, J., Socher, R., & Manning, C. (2014). GloVe: Global Vectors for Word Representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532-1543.
3. Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep contextualized word representations. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT), Volume 1 (Long Papers), pp. 2227-2237.
4. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. In Advances in neural information processing systems (NeurIPS), pp. 5998-6008.
5. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). 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), Volume 1 (Long and Short Papers), pp. 4171-4186.
6. Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). Improving Language Understanding by Generative Pre-Training.
7. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. (2019). RoBERTa: A robustly optimized BERT pretraining approach. arXiv preprint arXiv:1907.11692.
8. Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., & Soricut, R. (2019). ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. In International Conference on Learning Representations (ICLR).

9. Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A., & Potts, C. (2013). Recursive deep models for semantic compositionality over a sentiment treebank. In Proceedings of the 2013 conference on empirical methods in natural language processing (EMNLP), pp. 1631-1642.
10. Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., & Bowman, S. R. (2018). GLUE: A multi-task benchmark and analysis platform for natural language understanding. arXiv preprint arXiv:1804.07461.
11. Rajpurkar, P., Zhang, J., Lopyrev, K., & Liang, P. (2016). SQuAD: 100,000+ questions for machine comprehension of text. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 2383-2392.
12. Tjong Kim Sang, E. F., & De Meulder, F. (2003). Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003 (CoNLL-2003), pp. 142-147.
13. Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., & Potts, C. (2011). Learning word vectors for sentiment analysis. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT), pp. 142-150.