Generalised Zero-shot Learning for Entailment-based Text Classification with External Knowledge

Yuqi Wang[1], Wei Wang[1], Qi Chen[2], Kaizhu Huang[1], Anh Nguyen[3], and Suparna De[4]

[1]*School of Advanced Technology*, *Xi'an Jiaotong Liverpool University*, Suzhou, China
[2]*School of AI and Advanced Computing*, *Xi'an Jiaotong Liverpool University*, Suzhou, China
[3]*Department of Computer Science*, *University of Liverpool*, Liverpool, United Kingdom
[4]*Department of Computer Science*, *University of Surrey*, Surrey, United Kingdom

yuqi.wang17@student.xjtlu.edu.cn,
{wei.wang03, qi.chen02, kaizhu.huang}@xjtlu.edu.cn,
anh.nguyen@liverpool.ac.uk, s.de@surrey.ac.uk

***Abstract*—Text classification techniques have been substantially important to many smart computing applications, e.g. topic extraction and event detection. However, classification is always challenging when only insufficient amount of labelled data for model training is available. To mitigate this issue, zero-shot learning (ZSL) has been introduced for models to recognise new classes that have not been observed during the training stage. We propose an entailment-based zero-shot text classification model, named as S-BERT-CAM, to better capture the relationship between the premise and hypothesis in the BERT embedding space. Two widely used textual datasets are utilised to conduct the experiments. We fine-tune our model using 50% of the labels for each dataset and evaluate it on the label space containing all labels (including both seen and unseen labels). The experimental results demonstrate that our model is more robust to the generalised ZSL and significantly improves the overall performance against baselines.**

***Index Terms*—Zero-shot learning, natural language processing, deep learning, BERT**

I. Introduction

Classification is at the foundation of smart computing to analyse data at scale in multiple fields, which has been continuously studied in the past few decades and further advanced by deep learning techniques in recent years. As for the conventional classification, deep learning usually requires a significant amount of labelled data for each class to train networks in a fully-supervised manner. However, regarding applications such as topic extraction [1] and event detection [2], some classes are less common in the real world, making it difficult to collect sufficient samples for model training. Besides, with the widespread popularity of social media, a large number of emerging topics or events are constantly being introduced. Conventional classification methods are not capable of handling these problems.

Recent years have witnessed the development of zero-shot learning (ZSL), referring to the ability to recognise new classes that do not appear in the training set. This can be achieved by effective knowledge transfer from seen to unseen labels. Instead of denoting each class label as an index in the pre-defined output space, ZSL requires label representations to be equipped with additional semantic information about classes, e.g. class attributes [3] and textual description [4]. Moreover, relationships between labels, e.g. cosine similarity in a specific semantic space [5] and hierarchical structure from external knowledge resources [6], can be leveraged to perform better predictions for unseen labels. Whilst ZSL has been notably successful in computer vision, there has been relatively little progress in natural language processing (NLP) and understanding. We believe that the potential of ZSL in NLP has not been fully explored.

Newly-developed contextualised NLP models such as BERT [7], with bidirectional attention-based mechanism, i.e. transformers [8], can extract essential features from textual sequences and learn high quality contextualised representations. Pre-trained BERT can be effectively employed for knowledge transfer and has produced impressive results in a variety of downstream tasks such as open-domain question answering [9] and aspect-based sentiment analysis [10]. Natural language inference (NLI) [11], a fundamental NLP task, requires a model to take two inputs as the premise and hypothesis, respectively, and determines whether the hypothesis can be entailed by the premise. This is analogical to human logical thinking while one is performing the text classification without learning from any labelled examples [12].

We study the generalised zero-shot text classification problem with an entailment-based approach by utilising the Siamese network architecture with BERT (Sentence BERT) that has been pre-trained over NLI tasks [13]. Sentence BERT typically employs the concatenation of two pooled embeddings from BERT along with their element-wise difference to fit a softmax classifier, which may not be able to fully capture the relationship between two inputs. To this end, we propose a new cross attention module (CAM) for better performance in the deep semantic space. We choose two commonly used textual

datasets for experiments. We fine-tune the model with 50% of labels for each dataset and test it on the label space containing all labels (including both seen and unseen labels) to evaluate its capacity for generalisation.

The rest of the paper is organised as follows: In Section II, we review the related work regarding zero-shot text classification. Then, we define the task, formulate the problem and present our proposed model in Section III. In Section IV, we describe the experiments on two commonly used multi-class classification datasets and evaluate the performance. Finally, we conclude the paper and define possible future research in Section V.

II. Related Work

Zero-shot text classification aims to determine the label for a textual document without any explicit supervision, which was first investigated by Chang *et al.* [14]. They expanded the name of each class label to a text fragment containing several keywords and employed the Explicit Semantic Analysis (ESA) algorithm [15] to represent each label and text in a unified semantic space. Their work highlighted the importance of meaningful semantic interpretation in ZSL. Inspired by this, later, unsupervised methods such as Label Similarity [16] were proposed with the study of word embedding in NLP.

Deep learning techniques such as Convolution Neural Network (CNN) [17] and Long Short-Term Memory (LSTM) [18] have been widely adopted in the field of NLP. Pushp and Srivastava [19] reformulated the original problem into text-tag relatedness prediction and proposed an LSTM-based method to learn the relationship between a sentence and tag. Zhang *et al.* [20] employed CNN with data and feature augmentation to integrate semantic knowledge such as class description and class hierarchy. To better incorporate knowledge from graph-structured data, i.e. knowledge graph, Nayak and Bach [21] employed Graph Convolutional Network (GCN) [22] to generate label representations for zero-shot text classification.

Recently, the combination of attention mechanism, bidirectional scheme and improved word embedding techniques has resulted in remarkable success. BERT [7], a powerful pre-trained NLP model, has achieved many state-of-the-art performances on text classification tasks. So far, there has been some research that utilised the BERT model for zero-shot text classification. Chen *et al.* [23] projected both label name and text from the BERT embedding space to the knowledge graph space and computed their semantic similarity to determine whether a text can be associated with a label. Yin *et al.* [12] reduced the text classification problem to entailment vs. non-entailment prediction. With the inference ability of the pre-trained BERT model, it could apply the knowledge gained from entailment datasets to solve the problem effectively.

III. Methodology

In this section, we first define the zero-shot text classification task and then introduce the entailment-based classification process. The overall architecture of the proposed S-BERT-CAM is shown in Figure 1. The S-BERT and CAM components will be discussed in III-C and III-D, respectively.

A. Task Definition

Let $\mathcal{Y}^s$ and $\mathcal{Y}^u$ be sets of seen and unseen labels, respectively, such that they are disjoint, i.e. $\mathcal{Y}^s \cap \mathcal{Y}^u = \phi$. Given the set of labelled training instances $\mathcal{D}^{tr} = \{(x_i^{tr}, y_i^{tr})\}_{i=1}^{N_{tr}}$, where $y_i^{tr} \in \mathcal{Y}^s$, our aim is to train a model that can generalise well on the test set $\mathcal{D}^{ts} = \{(x_i^{ts}, y_i^{ts})\}_{i=1}^{N_{ts}}$, where $y_i^{ts} \in \mathcal{Y}^s \cup \mathcal{Y}^u$.

B. Entailment-based Classification

Since we employ an entailment-based approach for text classification, for each label $y \in \mathcal{Y}^s \cup \mathcal{Y}^u$, we represent it using the description of the corresponding entity from a knowledge base, DBpedia [1] and covert it to the hypothesis. We list 2 examples of label-hypothesis conversion for entailment-based classification in Table I, where the underlined text is adapted from the entity description in DBpedia. The text that is to be classified is the premise.

TABLE I
Example hypothesis for entailment-based classification

Class Label	Reference Entity	Example Hypothesis
Film	http://dbpedia.org/page/Film	This text is about a work of visual art through the use of moving images.
Village	http://dbpedia.org/page/Village	This text is about a clustered human settlement or community often located in rural areas.

To prepare the training data for the entailment vs. non-entailment prediction task, for each instance $(x^{tr}, y^{tr}) \in \mathcal{D}^{tr}$, we randomly generate a negative sample $(x^{tr}, y^{tr\prime})$, where $y^{tr\prime} \neq y^{tr}$ and $y^{tr\prime} \in \mathcal{Y}^s$. In the training stage, the model is required to learn if the hypothesis converted from a label can be entailed by the premise. As a result, we will learn an entailment model $f(x, y; \theta)$, where θ is the model parameter.

With respect to the testing, for each text x^{ts} in $\mathcal{D}^{ts}$, the model performs the entailment with premise x^{ts} and hypothesis converted from each label, y, $y \in \mathcal{Y}^s \cup \mathcal{Y}^u$, one by one. Normally, the label whose hypothesis yields the highest entailment score will be selected as the predicted one. However, in the generalised ZSL, the model tends to "overfit" to seen labels since there is no unseen label for training. Hence, we employ calibrated stacking [24] to reduce the bias in our work, i.e.

$$\hat{y} = \underset{y \in \mathcal{Y}^s \cup \mathcal{Y}^u}{\operatorname{argmax}} \; f(x, y; \theta) - \gamma \mathbb{I}[y \in \mathcal{Y}^s] \tag{1}$$

where $\mathbb{I}[\cdot]$ equals to 1 if $y \in \mathcal{Y}^s$, otherwise 0; γ is the calibration factor, a hyper-parameter will be determined through the validation.

[1] https://www.dbpedia.org/

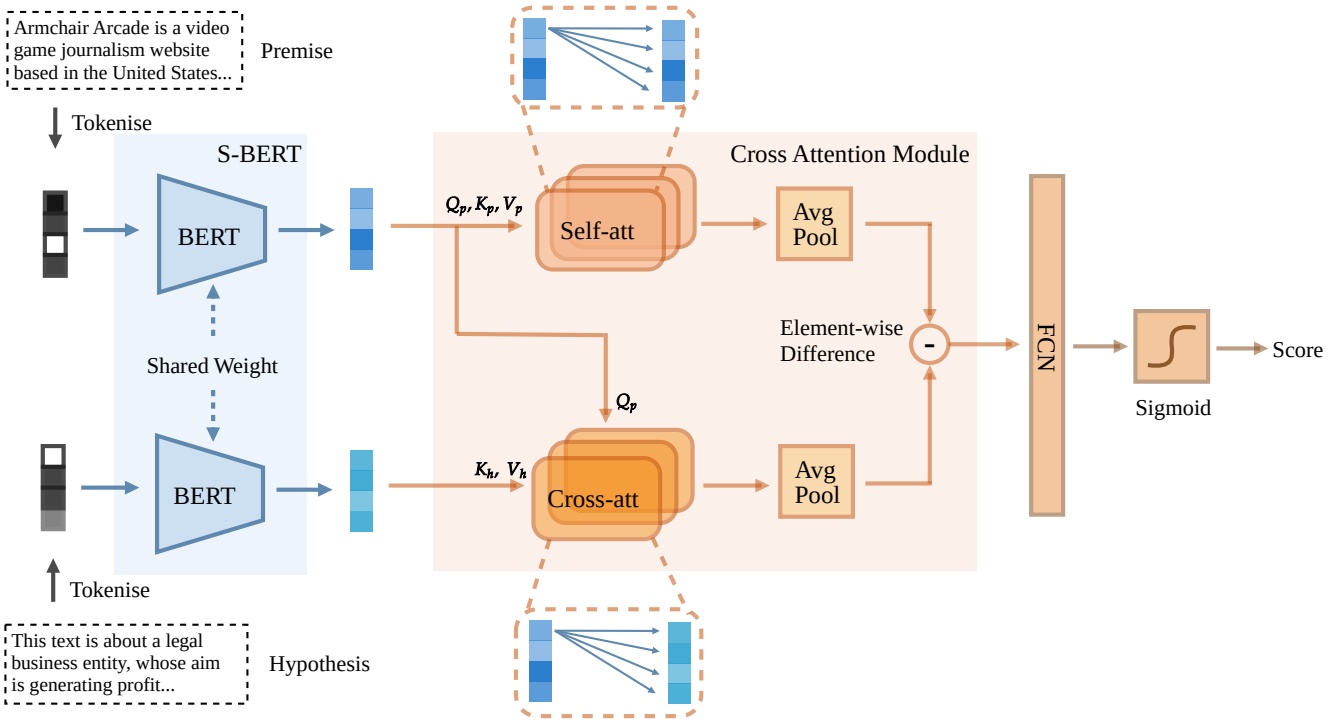


Fig. 1. Overall architecture of S-BERT-CAM

C. Sentence BERT

Over the years, the effectiveness of transfer learning has been widely confirmed: knowledge gained from pre-training tasks can be applied to solve other related problems. Pre-trained NLP models such as BERT are context-dependent and have the ability to capture the high-level concept of each sentence [7]. By fine-tuning the BERT model with only a limited amount of domain-specific data, one can obtain state-of-the-art results on multiple purpose-oriented NLP tasks. However, the BERT model, as a very deep cross-encoder, has a major limitation for entailment-based classification. As both premise and hypothesis will be passed to the model simultaneously, the time complexity would be extremely high, especially when there is a large number of documents and classes [13].

To mitigate the problem and generate compatible representations for both premise and hypothesis, we employ Sentence BERT (S-BERT) [13] as the base model, which has been pre-trained over NLI tasks on large general domain corpora. S-BERT consists of the Siamese network architecture to produce semantically meaningful representations while significantly reducing the cost of computation inference as each representation in the BERT embedding space can be pre-computed during testing.

Figure 2 demonstrates the architecture of a single BERT model. For each input (premise or hypothesis), we first add two special tokens (<CLS> and <SEP>) at the beginning and end, respectively, and tokenise the input text by splitting it into different tokens. After the token-level representations $\boldsymbol{H}^0 = \{\boldsymbol{h}_1^0, \boldsymbol{h}_2^0, ..., \boldsymbol{h}_n^0\}$ are generated with the summation of position embedding, token embedding and segment embedding, the contextualised representation will be calculated recursively with fully-connected bidirectional transformers [8], i.e.

$$\boldsymbol{H}^l = \text{TransformerBlock}_l(\boldsymbol{H}^{l-1}) \tag{2}$$

where $\boldsymbol{H}^l = \{\boldsymbol{h}_1^l, \boldsymbol{h}_2^l, ..., \boldsymbol{h}_n^l\}$, standing for the contextualised representation in the l-th layer. The output of the last hidden layer is used as the final representation of the premise/hypothesis in the BERT embedding space. We denote them as $\boldsymbol{H}_p$ and $\boldsymbol{H}_h$, respectively.

D. Cross Attention Module (CAM)

While performing the classification task, S-BERT usually employs the concatenation of two pooled embeddings from BERT along with their element-wise difference to fit a softmax classifier [13]. We observe that although this kind of concatenation is useful to some extent, it is not sufficient to model the relationship between the promise and hypothesis. More discriminative features should be derived for better performance.

Attention mechanism [8] aims to locate and highlight the relevant part of the input, e.g. areas in an image or words

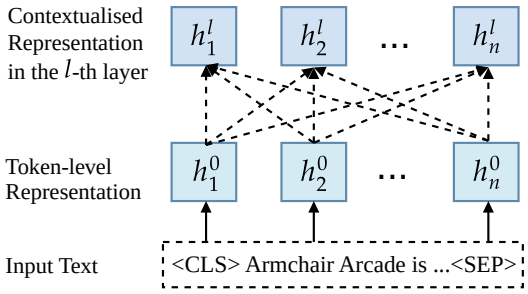


Fig. 2. Architecture of a single BERT

in a text, which has been widely used in the multi-modality matching [25]. In our work, although all inputs are in the same modality, premise and hypothesis are, in fact, from different resources. To this end, we propose a new cross attention module (CAM) to address the semantic discrepancy problem and guarantee the discriminability of the output feature. CAM takes two inputs: $\boldsymbol{H}_p$ and $\boldsymbol{H}_h$ from the S-BERT model. Each input has its corresponding query, key and value that can be obtained by multiplying trainable weights $\boldsymbol{W}^Q$, $\boldsymbol{W}^K$ and $\boldsymbol{W}^V$. Therefore, the query, key and value of the premise and hypothesis can be formulated in the following manner:

$$\begin{pmatrix} \boldsymbol{Q}_p, \boldsymbol{K}_p, \boldsymbol{V}_p \\ \boldsymbol{Q}_h, \boldsymbol{K}_h, \boldsymbol{V}_h \end{pmatrix} = \begin{pmatrix} \boldsymbol{H}_p\boldsymbol{W}^Q, \boldsymbol{H}_p\boldsymbol{W}^K, \boldsymbol{H}_p\boldsymbol{W}^V \\ \boldsymbol{H}_h\boldsymbol{W}^Q, \boldsymbol{H}_h\boldsymbol{W}^K, \boldsymbol{H}_h\boldsymbol{W}^V \end{pmatrix} \tag{3}$$

CAM consists of a self-attention and cross-attention mechanism. As for the self-attention mechanism, every element in the input is required to interact with every other element in the same input (including itself) in order to identify which part of the input to focus on and which element is essential. This intra-information is important for the document that we would like to classify. Hence, we calculate the self-attention matrix of the premise as follows:

$$\text{Self-att}(\boldsymbol{Q}_p, \boldsymbol{K}_p, \boldsymbol{V}_p) = \text{Softmax}(\frac{\boldsymbol{Q}_p\boldsymbol{K}_p^\top}{\sqrt{\boldsymbol{d}_k}})\boldsymbol{V}_p \tag{4}$$

where $\boldsymbol{d}_k$ is the scaling factor.

Moreover, to explore the relationship between premise and hypothesis, we employ a cross-attention mechanism. In contrast to the self-attention mechanism, it compares every element in the first input to every element in the second input. This way measures how important each element in the first input is with respect to all the elements in the second input. The cross-attention matrix of premise and hypothesis can be calculated as follows:

$$\text{Cross-att}(\boldsymbol{Q}_p, \boldsymbol{K}_h, \boldsymbol{V}_h) = \text{Softmax}(\frac{\boldsymbol{Q}_p\boldsymbol{K}_h^\top}{\sqrt{\boldsymbol{d}_k}})\boldsymbol{V}_h \tag{5}$$

We choose the average pooling to calculate both the self-attention and cross-attention vector for overall information preservation and dimension reduction. If the given hypothesis can be entailed by the premise, the self-attention and cross-attention vector will tend to be similar since an important word in the premise should also be relevant to the hypothesis or part of it. Based on this, we compute the element-wise difference of the pooled self-attention and cross-attention vector as the output features of the CAM, i.e.

$$\text{CAM}(\boldsymbol{H}_p, \boldsymbol{H}_h) = \Big\{ \text{Avg-pool}\big(\text{Self-att}(\boldsymbol{Q}_p, \boldsymbol{K}_p, \boldsymbol{V}_p)\big) - \text{Avg-pool}\big(\text{Cross-att}(\boldsymbol{Q}_p, \boldsymbol{K}_h, \boldsymbol{V}_h)\big) \Big\}^{|\cdot|} \tag{6}$$

Finally, the output features will be passed to a fully-connected network (FCN) followed by the sigmoid activation function to acquire the entailment score (0-1).

IV. Experiments

A. Datasets

For our experiments, we utilised two widely used labelled textual datasets: 1) DBpedia ontology dataset [26], which is made up of 14 non-overlapping Wikipedia topics. Each class contains 20,000 training and 5,000 testing samples; 2) 20newsgroup dataset[2] was created by selecting textual data from social media, including 20 news topics. Each class contains around 700 training and 300 testing samples.

B. Baseline Methods

- **Label Similarity:** this method [16] computes cosine similarity between each label embedding and the sum of n-gram word embeddings in a document. For the multi-class classification task, the label with the highest similarity score will be selected as the predicted label.
- **LSTM:** it refers to the second architecture in the work of Pushp and Srivastava [19], where the final hidden state of an LSTM is concatenated with a label embedding to predict the relatedness of the given text and label.
- **CNN-AUG:** it refers to the work of Zhang *et al.* [20], which is a CNN-based model with data and feature augmentation to integrate semantic knowledge.
- **InferSent-GloVe:** it is a Siamese network with Bi-LSTM pre-trained over NLI tasks to produce semantic representations [27]. It employs GloVe [28] to initialise the embedding of input. For the classification task, it concatenates two pooled embeddings from Bi-LSTM along with their element-wise difference and multiplication to fit the final classifier.
- **S-BERT:** we re-implemented S-BERT [13], a Siamese network with BERT pre-trained over NLI tasks. For the classification task, it concatenates two pooled embeddings from BERT along with their element-wise difference to fit the final classifier.

[2]http://qwone.com/∼jason/20Newsgroups/

C. Setup

We downloaded the pre-trained base model from huggingface[3] and employed the Adam optimiser with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. We mainly followed recommendations from Devlin *et al.* [7] to set the learning rate ($3e-5$) and batch size (32). We set the maximum length of each input sequence to 128.

The unseen rate was set to 50%, i.e. 50% labels would not be observed during training. To determine the value of γ in Equation (1) and tune other hyper-parameters, we performed a simple validation based on the work of Chao *et al.* [24]: further splitting the training labels into pseudo-seen and pseudo-unseen labels following the unseen rate (No unseen labels were included). We simulated the entailment-based classification setting mentioned in Section III-B to derive a validation set and train the model with pseudo-seen labels only and validate on all training labels. We selected hyper-parameters that can maximise the overall performance during the validation and trained the model with all training labels.

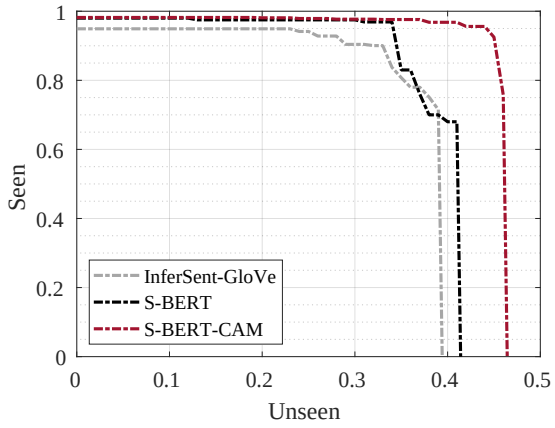
We implemented our model using PyTorch 1.10.2 and Python 3.6, and used a GeForce RTX 3060 Ti GPU as hardware acceleration.

D. Evaluations

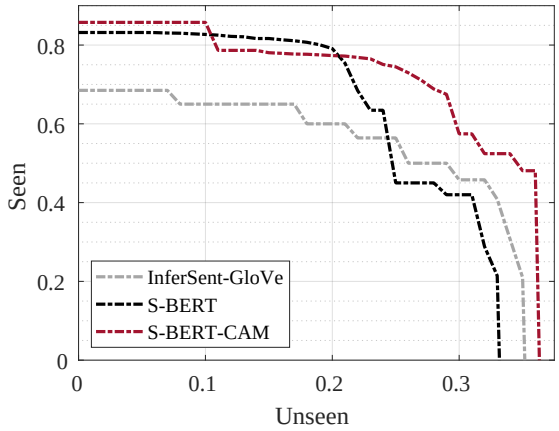
1) Accuracy on Testing Data: We chose accuracy as the main metric to evaluate the performance of models since the testing data is relatively balanced on two datasets. The performance of different models on testing data is reported in Table II. Label Similarity [16] gave the highest accuracy on unseen labels among all methods. However, it could not deliver satisfactory results on seen labels compared to all other supervised methods. LSTM [19], on the contrary, produced good predictions on seen labels while performing poorly on unseen labels, indicating that it has no ability to apply the knowledge gained from seen labels to tackle the zero-shot classification on unseen labels. These two methods failed to generalise well across all labels. It is worth noting that the three entailment based models, InferSent-GloVe [27], S-BERT [13] and S-BERT-CAM, significantly improved the overall performance on both datasets, which demonstrated the effectiveness of entailment-based classification. Moreover, the proposed S-BERT-CAM model outperformed the baseline S-BERT by 2.1% and 1.9% on DBpedia and 20News, respectively, which proved its capability of generalisation.

2) Robustness to Generalised ZSL: We plotted Area Under Seen-Unseen Accuracy Curve (AUSUC) [24] for further comparison. AUSUC is a metric proposed by Chao *et al.* [24], which is obtained by varying the calibration factor γ during the validation. The AUSUC directly indicates how robust a model is to the generalised ZSL. In Figure 3, it is observed that our model generated the highest AUSUC and outperformed the baselines on both datasets.

[3]https://huggingface.co/transformers/



(a) DBpedia



(b) 20News

Fig. 3. Area Under Seen-Unseen Accuracy Curve (AUSUC) during validation on two datasets

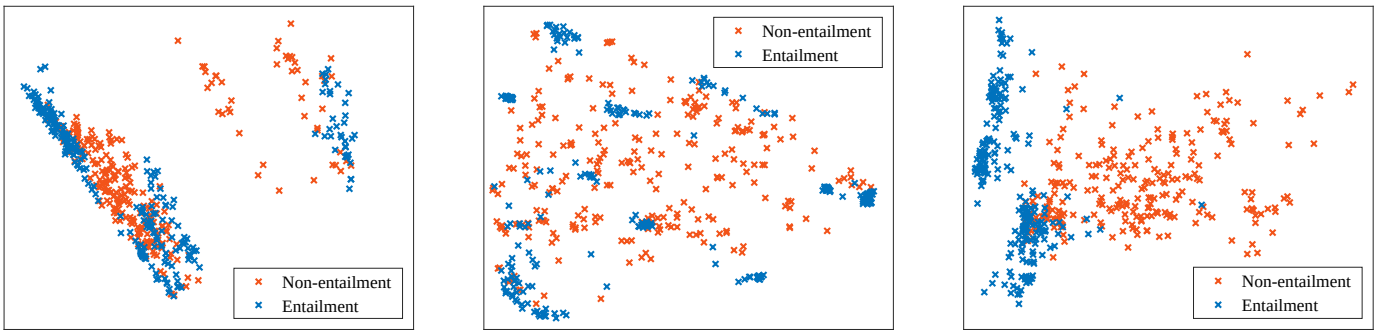
3) Visualisation: To visualise the output features before the final classification in different methods, we also provided PCA visualisation in Figure 4. Compared with S-BERT, whose entailment features were scattered around, S-BERT-CAM generated better clusters so as to confirm the validity of CAM. As for the InferSent-GloVe, the main clusters of entailment and non-entailment features were highly overlapped, and there were some outliers. On the contrary, these two types of entailment features from S-BERT-CAM presented a clear separation, allowing the model to easily determine whether a hypothesis could be entailed by the premises.

V. Conclusion and Future Work

Based on the entailment-based models, we designed S-BERT-CAM to better capture the relationship between premise and hypothesis. It is pre-trained over NLI tasks, fine-tuned with 50% of the labels, and evaluated with all labels from two widely used textual datasets. The experimental results show that the proposed model can significantly improve the overall performance, which confirms its generalisation ability and robustness to the generalised ZSL. This also shows that the model can generate more discriminative features and alleviate the semantic discrepancy problem.

TABLE II
COMPARISON OF PERFORMANCE ON TESTING DATA

Models	DBpedia			20News		
	Unseen	**Seen**	**Overall**	**Unseen**	**Seen**	**Overall**
Label Similarity [16]	**36.9**	40.1	38.6	**26.6**	29.3	28.0
LSTM [19]	4.4	96.0	50.2	5.2	70.9	38.1
CNN-AUG [20]	19.7	**98.2**	59.0	16.8	**76.7**	46.9
InferSent-GloVe [27]	30.6	90.4	60.5	23.9	56.4	40.2
S-BERT [13]	33.5	89.8	61.7	21.0	75.4	48.2
S-BERT-CAM (Ours)	36.1	91.5	**63.8**	25.6	74.5	**50.1**



(a) InferSent-GloVe (b) S-BERT (c) S-BERT-CAM

Fig. 4. PCA visualisation of output features from different entailment models

In future, we will consider the more challenging, label fully-unseen problem for ZSL and design new models for this based on our current work. The work presented in this paper only focuses on multi-class text classification, and we employ some textual descriptions from external knowledge bases. In many real-world scenarios, e.g. social media data, each message can be associated with more than one label. For future work, we will work on multi-label classification by label-entity alignment and utilise relationships between labels, e.g. similarity and subsumption [29], from external knowledge bases to further improve the usability and performance.

REFERENCES

[1] A. Rekik and S. Jamoussi, "Deep learning for hot topic extraction from social streams," in *International Conference on Hybrid Intelligent Systems*. Springer, 2016, pp. 186–197.

[2] S. Dabiri and K. Heaslip, "Developing a twitter-based traffic event detection model using deep learning architectures," *Expert Systems with Applications*, vol. 118, pp. 425–439, 2019.

[3] C. H. Lampert, H. Nickisch, and S. Harmeling, "Learning to detect unseen object classes by between-class attribute transfer," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 951–958.

[4] L. J. Ba, K. Swersky, S. Fidler, and R. Salakhutdinov, "Predicting deep zero-shot convolutional neural networks using textual descriptions," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 4247–4255.

[5] C. Gan, M. Lin, Y. Yang, Y. Zhuang, and A. G. Hauptmann, "Exploring semantic inter-class relationships (sir) for zero-shot action recognition," in *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, 2015, pp. 3769–3775.

[6] S. Kordumova, T. Mensink, and C. G. Snoek, "Pooling objects for recognizing scenes without examples," in *Proceedings of the 2016 ACM on International Conference on Multimedia Retrieval*, 2016, pp. 143–150.

[7] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "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, Volume 1 (Long and Short Papers)*, 2019, pp. 4171–4186.

[8] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017, pp. 6000–6010.

[9] W. Yang, Y. Xie, A. Lin, X. Li, L. Tan, K. Xiong, M. Li, and J. Lin, "End-to-end open-domain question answering with bertserini," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, 2019, pp. 72–77.

[10] Y. Wang, Q. Chen, and W. Wang, "Multi-task bert for aspect-based sentiment analysis," in *2021 IEEE International Conference on Smart Computing (SMARTCOMP)*. IEEE, 2021, pp. 383–385.

[11] S. Bowman, G. Angeli, C. Potts, and C. D. Manning, "A large annotated corpus for learning natural language inference," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 2015, pp. 632–642.

[12] W. Yin, J. Hay, and D. Roth, "Benchmarking zero-shot text classification: datasets, evaluation and entailment approach," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 3914–3923.

[13] N. Reimers and I. Gurevych, "Sentence-bert: sentence embeddings using siamese bert-networks," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019, pp. 3982–3992.

[14] M.-W. Chang, L. Ratinov, D. Roth, and V. Srikumar, "Importance of semantic representation: dataless classification," in *Proceedings of the 23rd National Conference on Artificial Intelligence*, 2008, pp. 830–835.

[15] E. Gabrilovich and S. Markovitch, "Computing semantic relatedness using wikipedia-based explicit semantic analysis," in *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, 2007, pp. 1606–1611.

[16] P. V. Sappadla, J. Nam, E. L. Mencía, and J. Fürnkranz, "Using semantic

similarity for multi-label zero-shot classification of text documents." in *European Symposium on Artificial Neural Networks*, 2016.

[17] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[18] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[19] P. K. Pushp and M. M. Srivastava, "Train once, test anywhere: Zero-shot learning for text classification," *arXiv preprint arXiv:1712.05972*, 2017.

[20] J. Zhang, P. Lertvittayakumjorn, and Y. Guo, "Integrating semantic knowledge to tackle zero-shot text classification," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019, pp. 1031–1040.

[21] N. V. Nayak and S. H. Bach, "Zero-shot learning with common sense knowledge graphs," *arXiv preprint arXiv:2006.10713*, 2020.

[22] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[23] Q. Chen, W. Wang, K. Huang, and F. Coenen, "Zero-shot text classification via knowledge graph embedding for social media data," *IEEE Internet of Things Journal*, 2021.

[24] W.-L. Chao, S. Changpinyo, B. Gong, and F. Sha, "An empirical study and analysis of generalized zero-shot learning for object recognition in the wild," in *European Conference on Computer Vision*. Springer, 2016, pp. 52–68.

[25] X. Wei, T. Zhang, Y. Li, Y. Zhang, and F. Wu, "Multi-modality cross attention network for image and sentence matching," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 10 941–10 950.

[26] X. Zhang, J. Zhao, and Y. LeCun, "Character-level convolutional networks for text classification," *Advances in Neural Information Processing Systems*, vol. 28, pp. 649–657, 2015.

[27] A. Conneau, D. Kiela, H. Schwenk, L. Barrault, and A. Bordes, "Supervised learning of universal sentence representations from natural language inference data," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 670–680.

[28] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1532–1543.

[29] H. Dong, W. Wang, K. Huang, and F. Coenen, "Automated social text annotation with joint multilabel attention networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 5, pp. 2224–2238, 2020.