NEURAL KNOWLEDGE TRANSFER FOR SENTIMENT ANALYSIS IN TEXTS WITH FIGURATIVE LANGUAGE

Dionysios Karamouzas *Ioannis Mademlis* *Ioannis Pitas*

Aristotle University of Thessaloniki
Thessaloniki, Greece
Email: {imademlis,pitas}@csd.auth.gr

ABSTRACT

Sentiment analysis in texts, also known as opinion mining, is a significant Natural Language Processing (NLP) task, with many applications in automated social media monitoring, customer feedback processing, e-mail scanning, etc. Despite recent progress due to advances in Deep Neural Networks (DNNs), texts containing figurative language (e.g., sarcasm, irony, metaphors) still pose a challenge to existing methods due to the semantic ambiguities they entail. In this paper, a novel setup of neural knowledge transfer is proposed for DNN-based sentiment analysis of figurative texts. It is employed for distilling knowledge from a pretrained binary recognizer of figurative language into a multiclass sentiment classifier, while the latter is being trained under a multitask setting. Thus, hints about figurativeness implicitly help resolve semantic ambiguities. Evaluation on a relevant public dataset indicates that the proposed method leads to state-of-the-art accuracy.

***Index Terms*—** knowledge distillation, sentiment analysis, opinion mining, figurative language, natural language processing

1. INTRODUCTION

Sentiment analysis, also known as *opinion mining*, is one of the most important tasks in the field of Natural Language Processing (NLP), due to its many different uses. In the simplest case, it is the problem of assigning a class label to a corpus of written text, that may range from a single phrase up to a complete essay, where each class expresses a possible sentiment of the author concerning the content of the text [1]. Thus, in typical scenarios, the class labels are “positive”, “neutral”, “negative”, etc. Opinion mining can be employed for tasks such as automated social media monitoring, customer feedback processing or e-mail scanning. The huge rise in the volume of subjectively written texts that are publicly available on-line during the past decade, such as user reviews or social media posts, has further increased the practical significance of accurate sentiment analysis methods.

Recent progress due to advances in Deep Neural Networks (DNNs) has augmented tremendously the performance of state-of-the-art sentiment analysis methods. Current approaches rely on DNN architectures, such as Convolutional Neural Networks (CNNs) and/or Long Short-Term Memory (LSTM) variants, that are trained for classification or regression under a supervised learning setting. Under this framework, handcrafted input textual features have largely been replaced by DNN-derived learnt word embeddings [2], which manage to capture significant semantic properties.

However, texts containing figurative language (e.g., involving sarcasm, irony, metaphors, etc.) still pose a challenge to existing methods [3]. This is due to difficulties in identifying whether a figurative phrase actually implies negative or positive opinion, given its inherent semantic ambiguity. The issue is especially pronounced because of the ubiquitous nature of figurative language (FL) in human communication. For example, sarcasm and irony are known to attract higher attention in on-line communities [4], with many people being adept at their usage and exploiting them to increase the impact of their social media posts (e.g., on Twitter). *This situation motivates research specifically focused on sentiment analysis for FL-heavy texts.*

In this paper, a novel setup of knowledge distillation [5] is proposed for DNN-based sentiment analysis of figurative texts [3]. It is employed for transferring knowledge from a binary pretrained FL recognizer (teacher) into a multiclass sentiment classifier (student), while the latter is being trained under a multitask setting. Up to now, *knowledge distillation has mainly been applied in softmax-based classification settings, where the teacher and the student solve identical tasks.* This paper, instead, proposes *distillation from a binary classification teacher with sigmoidal output, in order to aid a multiclass classification student on a completely different task.* Although neural distillation in general has been repeatedly applied for NLP in the past, the specific algorithmic setup proposed in this paper is novel.

The research leading to these results has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 951911 (AI4Media).

Therefore, the main contributions of this paper are:

- An innovative way to apply neural distillation in NLP.
- Exploiting neural distillation for increasing sentiment analysis accuracy in texts with FL.

To the best of our knowledge this has not been done before, since all relevant previous papers simply applied generic sentiment recognizers on texts with FL, without tackling the peculiarities of FL. Evaluation on a relevant public dataset containing FL-heavy texts indicates that the proposed method leads to state-of-the-art performance: it surpasses all competing approaches for sentiment analysis.

2. RELATED WORK

This Section presents the existing state-of-the-art concerning: a) sentiment analysis in texts with figurative language (FL), and b) knowledge distillation.

2.1. Sentiment Analysis on Figurative Language

Sentiment analysis on FL was originally proposed at the Semantic Evaluation Workshop 2015 - Task 11 [3]. Given a set of tweets that are rich in metaphor, sarcasm and irony, the goal was to determine whether a user has expressed a positive, negative or neutral sentiment in each one of them. The participants were provided with both integer and real-valued labels, thus allowing the use of either classification or regression approaches. All published studies related to sentiment analysis on FL are evaluated on the S15-T11 tweet collection, due to the lack of other public relevant datasets.

Initial methods tackling the task relied on Support Vector Machines (SVMs), decision trees or regression learning models, operating on handcrafted features (e.g., computed on n-grams or tf–idf statistics) and lexicons (e.g., SentiWordnet, Depeche Mood, American National Corpus, etc.) [6] [7]. Later, [8] approached the task as a regression problem and exploiting a CNN architecture. Unigrams with more than two occurrences were employed as input textual features, while hashtags designating the FL category types were replaced by binary indicators. DESC [9] integrated a Bi-LSTM, an Attention LSTM and an MLP, with the first two being fed pretrained GloVe word embeddings [2] and the MLP being fed handcrafted tweet features, such as unigrams, bigrams and tf-idf statistics. DESC was extended in [10] by utilizing contextual input embeddings from a pretrained RoBERTa model [11], which are fed to a Recurrent Convolutional Neural Network (RCNN). The task was formulated as a classification problem.

2.2. Neural Knowledge Transfer

Knowledge distillation is the most prevalent form of neural knowledge transfer. It implies training a student DNN according to the response of a pretrained teacher DNN to input samples (so-called "transfer dataset"), possibly consisting of unknown/unlabeled data. Typically, the student is assumed to be initially untrained (encoding no prior knowledge), while the two networks may have different neural architectures [5]. The student DNN is trained with an objective function that penalizes the deviation of its output from the corresponding output of the fixed teacher DNN on the transfer set [5]. The goal is to transfer so-called *dark knowledge* from the teacher to the student.

In classification tasks, abnormally high softmax temperatures are typically employed for extracting both outputs while training the student, in order to obtain soft distributions over the class labels that implicitly encode similarities among data samples. After training, the student is deployed with a more conventional softmax temperature. If labels are available for supervising training on the transfer set, a regular classification term may also be added to the objective. In this case, the distillation term mainly has a regularizing effect.

Several variants of this base neural distillation scheme have surfaced over the years. For instance, in case the transfer set is known/labeled, the parameters/synaptic weights of the trained teacher may be employed for initializing the student smartly before proceeding with regular supervised training [12]. Hints from hidden layer activations in the teacher when processing the transfer set may also be employed for guiding student training, besides the final output [13]. Dimensionality reduction may be needed when the layer dimensions differ between student and teacher. To bypass this limitation, student training in [14] is guided by the similarities between transfer set data samples representations constructed by the teacher. Intermediate "teacher assistant" networks, in the context of a multi-step process, was also proposed in [15] for scenarios where the two models differ in complexity. Finally, distillation from a deep linear teacher for binary classification was presented in the context of a purely theoretical analysis in [16], while a similar investigation was conducted in [17] for shallow feed-forward neural networks.

3. PROPOSED METHOD

The proposed method exploits knowledge distillation [5] for increasing DNN-based sentiment analysis accuracy on texts with FL, that employ sarcasm, irony and/or metaphor. Thus, during training, a teacher-student architecture is utilized to enrich the student model with the knowledge of a pretrained FL recognizer. The latter one is a binary classifier, while the former one is a multiclass classifier tasked with identifying sentiment in input texts. Thus, due to the nature of the teacher and the different task it solves compared to the student, an atypical kind of distillation is proposed that has not been previously employed in regular DNNs for real-world tasks.

Below, all neural models are assumed to be trained with error back-propagation and a variant of stochastic gradient

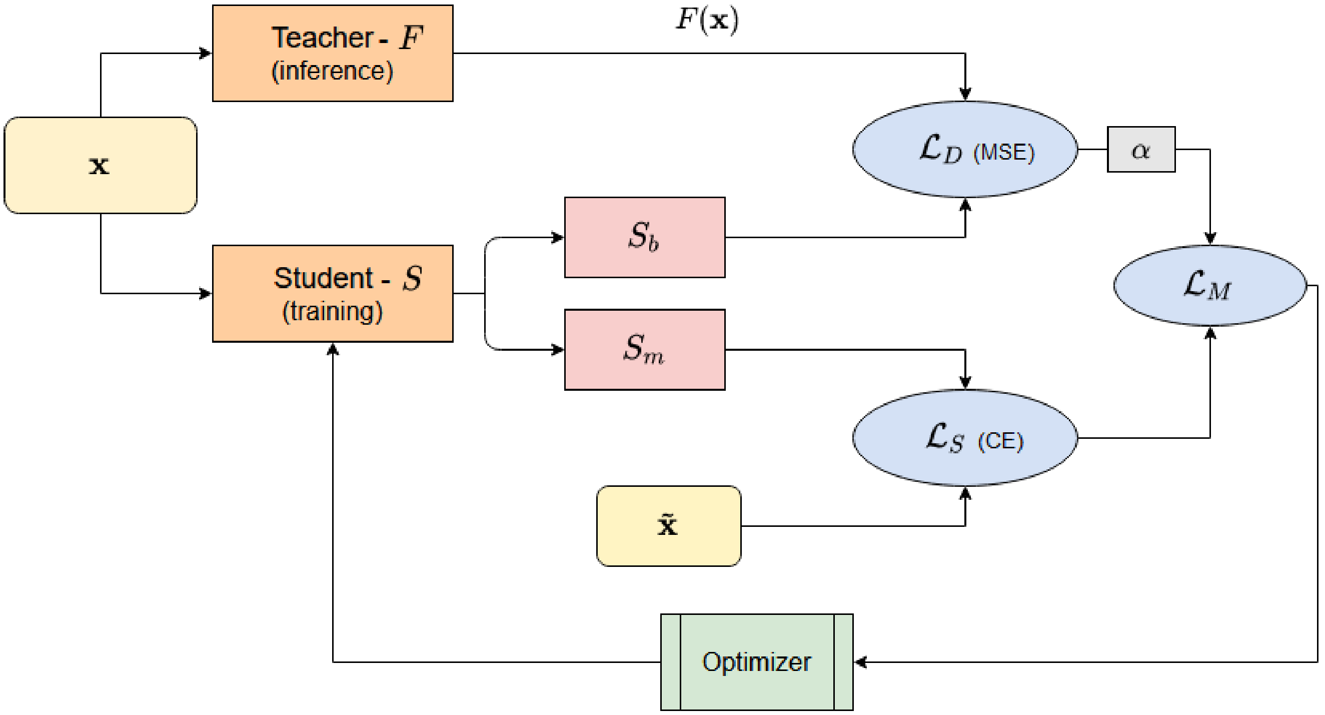

Fig. 1: The proposed teacher-student training architecture.

descent. Let us also assume that a DNN-based binary text classifier F has been pretrained under a regular supervised setting on a database containing two classes: "figurative", "literal/non-figurative". Since it is common for binary neural classifiers to end with a single sigmoidal neuron, we assume this is the case for F. Thus, a real-valued scalar output of 0/1 corresponds to figurative/literal class prediction, respectively, while a typical output $F(\mathbf{x}) \in \mathbb{R}$ for a respective input data point $\mathbf{x}$ would actually lie in the interval $[0, 1]$.

The student S is the neural model we actually want to optimize; on a different, sentiment-annotated dataset. Without loss of generality, we assume that it is being trained under a supervised multiclass text classification setting. Typically, $N \geq 3$ classes are employed for the sentiment analysis/opinion mining task ("positive", "neutral", "negative", etc.) and a final softmax activation layer used for deriving the class prediction. S is trained by a regular, suitable loss function $\mathcal{L}_S$, such as Cross-Entropy (CE).

Methodology: The proposed method consists in training S with the following multitask loss function:

$$\mathcal{L}_M = \mathcal{L}_S + \alpha \mathcal{L}_D, \tag{1}$$

where $\mathcal{L}_D$ is being computed at each iteration by exploiting the pretrained F. α is a scalar hyperparameter balancing the two individual training objectives $\mathcal{L}_S$ and $\mathcal{L}_D$. $\mathcal{L}_D$ essentially distills $F(\mathbf{x})$: the output of F for input $\mathbf{x}$, where $\mathbf{x}$ is the current training data point. As noted in [16] for the deep linear scenario, sigmoidal output activation for the binary classification case is equivalent to *soft labels* typically employed for softmax-based multiclass distillation [5]. To compute this loss term, a parallel output layer s_b serving as an auxiliary binary classification head is architecturally plugged onto the penultimate layer of S, while $F(\mathbf{x})$ serves as real-valued/continuous substitute "ground-truth" for $\mathcal{L}_D$. To avoid confusion, the normal softmax-based multiclass classification head of S is denoted below by $\mathbf{s}_m$. Thus, assuming N sentiment classes, $\mathbf{s}_m/s_b$ is a final neural layer consisting of $N/1$ neuron(s), respectively. Then, $\mathbf{s}_m(\mathbf{x}) \in \mathbb{R}^N$ and $s_b(\mathbf{x}) \in \mathbb{R}$ are the outputs of $\mathbf{s}_m$ and s_b, respectively, when S is given $\mathbf{x}$ as input.

By employing the Mean Squared Error cost (MSE) for $\mathcal{L}_D$, Eq. (1) is concretized into:

$$\mathcal{L}_M = \mathcal{L}_S\left(\mathbf{s}_m(\mathbf{x}), \tilde{\mathbf{x}}\right) + \alpha\left(s_b(\mathbf{x}) - F(\mathbf{x})\right)^2, \tag{2}$$

where $\tilde{\mathbf{x}}$ is the actual, *sentiment* ground-truth class label corresponding to $\mathbf{x}$, in the context of multiclass classification. As previously mentioned, $\mathcal{L}_S$ can be any supervised loss function for classification, such as Cross-Entropy (CE). Notably, no *figurativeness* ground-truth label is exploited or required to exist for $\mathbf{x}$.

An overview of the proposed method is depicted in Figure 1. Importantly, no actual/real ground-truth annotation concerning the presence or type of FL is required or exploited while training S for sentiment analysis. *Of course, after S* *has been fully trained, both the entire F model and the auxiliary output layer/binary classification head s_b can be safely discarded.*

The underlying intuition behind the proposed multitask loss function is the conjecture that dark knowledge concerning the degree of figurativeness of an input text should aid a sentiment classifier in resolving ambiguities about the expressed sentiment, that arise due to sarcastic, metaphorical or ironical language. The proposed distillation loss term should have the effect of tuning the multiclass sentiment classifier towards identifying and overcoming such ambiguities. The FL recognizer F was selected to be a binary classifier in order to maximize its inference-stage success rate in this auxiliary task, by making the classification problem as easy as possible. Notably, knowledge distillation from binary classifiers with sigmoidal output, in order to aid a multiclass classifier on a different task, has not been previously presented for regular DNNs.

4. QUANTITATIVE EVALUATION

4.1. Implementation Details

The neural architecture ROB-RCNN from [10] was recreated in PyTorch and adopted for the base sentiment analysis student model S. This neural architecture utilizes a pretrained RoBERTa language model [11], combined with an RCNN [18], in order to efficiently capture contextual text information when representing each word. The final prediction is the output of a softmax layer. The reason behind choosing ROB-RCNN as a baseline was solely practical; *in principle, the proposed method can be used to augment any other sentiment classifier, as well.*

RoBERTa is followed by a Bi-LSTM layer, that attempts to model dependencies within the derived embeddings. The RoBERTa and the Bi-LSTM layer outputs are concatenated and jointly fed to a fully connected layer that simulates 1D convolution with a large kernel, in order to capture spatiotemporal dependencies in the projected latent space. After max pooling, a final softmax output layer leads to the final prediction.

The CNN/Bi-LSTM neural architecture OSLCfit [26] was pretrained for FL recognition, following the training process prescribed in [26], and then adopted as the binary classification teacher model F. The CNN applies convolution of sizes 3, 4 and 5, thereby learning fixed length features of 3-grams, 4-grams, and 5-grams, respectively. The convolution is followed by a ReLU activation function. These convolution features are then downsampled by using a 1D max pooling function. The CNN outputs are concatenated and together with the BiLSTM output are jointly fed into a fully connected sigmoidal output layer to produce the final sentiment score. Its input text representations are derived by using 200-dimensional embeddings from a pretrained GloVe model [2].

Table 1: Evaluation results on the S15-T11 dataset. Higher/lower is better for the COS/MSE metric, respectively. Best results are in bold.

Method	***COS***	***MSE***
ELMo [19]	0.71	3.61
USE [20]	0.71	3.17
NBSVM [21]	0.69	3.23
FastText [22]	0.72	2.99
XLnet [23]	0.76	1.84
BERT-Cased [24]	0.72	1.97
BERT-Uncased [24]	0.79	1.54
RoBERTa [11]	0.78	1.55
UPF [6]	0.71	2.46
ClaC [25]	0.76	2.12
DESC [9]	0.82	2.48
ROB-RCNN [10]	0.82	1.92
ROB-RCNN + Proposed ($\mathcal{L}_{\mathcal{D}}$)	**0.85**	**1.50**

This teacher model was pretrained on the annotated dataset from [27], which contains 81,4K tweets grouped under 4 different class labels (“sarcasm”, “irony”, “figurative” and “regular”). The first three classes were combined in a general “figurative” class, in order to train F as a binary figurative text classifier.

The student S was trained using Adam optimization and Cross Entropy (CE) as the main multiclass classification student loss function $\mathcal{L}_S$.

4.2. Evaluation Setup

The S15-T11 dataset [3] was used for evaluating the proposed method and comparing it against competing approaches. It contains 8000/4000 tweets for training/test, respectively, including tweets with ironic, sarcastic and metaphorical language. The 12000 data points are grouped under 11 classes annotated with integers in an 11-point scale, ranging from -5 to +5, that denote the polarity of each tweet, from “very negative” to “very positive”. Since it is a sentiment analysis dataset, it *does not* contain ground-truth annotations/labels concerning the presence or type of FL.

Two evaluation metrics were employed: cosine similarity (COS, higher is better) and Mean Squared Error (MSE, lower is better). Assuming a test set of T data points, both are computed by comparing two T-dimensional integer vectors, respectively containing the predicted and the ground-truth class labels. The proposed method implementation is in fact ROB-RCNN [10] augmented with $\mathcal{L}_{\mathcal{D}}$ during training, while the baseline that we improve upon is ROB-RCNN trained with simple $\mathcal{L}_S$, instead of the proposed Eq. (2). Optimal hyperparameters were adopted from [10], while 5-fold cross-validation resulted in best $\alpha = 0.5$. Test-phase evaluation results are presented in Table 1, including the accuracy achieved by several competing methods. All reported figures are lifted from [10], except the ones for ROB-RCNN. The latter method was recreated, trained and evaluated ab initio by us, following strictly all implementation minutiae and hyperparameter values detailed in [10].

Overall, the proposed method implementation ROB-RCNN + $\mathcal{L}_{\mathcal{D}}$ achieves state-of-the-art performance in both metrics, thus confirming the validity of our underlying intuition. Remarkably, compared to DESC, it manages to decrease MSE from 2.48 to 1.50, while simultaneously increasing COS from 0.82 to 0.85. In contrast, baseline ROB-RCNN achieves MSE improvements over DESC, without gains in COS performance.

5. CONCLUSIONS

Natural Language Processing tasks, such as sentiment analysis in texts, have significantly progressed thanks to advances in Deep Neural Networks. However, any presence of figurative language (sarcasm, metaphor, irony) significantly increases the difficulty of the sentiment analysis task. This paper exploits the intuition that estimations about the existence of figurative language in an input text can boost the accuracy of a sentiment classifier, by helping it to internally resolve semantic ambiguities. Thus, the proposed method consists in a novel setup of knowledge distillation from a pretrained binary recognizer of figurative language, employed as an auxiliary task while training a multiclass sentiment analysis neural model under a multitask setting. Notably, no ground-truth annotation about figurativeness is required to exist while training for the sentiment analysis task. Evaluation on a relevant public dataset indicates that the proposed method leads to state-of-the-art performance, surpassing all competing approaches.

6. REFERENCES

[1] K. Ravi and V. Ravi, "A survey on opinion mining and sentiment analysis: tasks, approaches and applications," *Knowledge-based systems*, vol. 89, pp. 14–46, 2015.

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

[3] A. Ghosh, G. Li, T. Veale, P. Rosso, E. Shutova, J. Barnden, and A. Reyes, "Semeval-2015 Task 11: Sentiment analysis of figurative language in Twitter," in *Proceedings of the International Workshop on Semantic Evaluation (SemEval)*, 2015.

[4] S. Muresan, R. Gonzalez-Ibanez, D. Ghosh, and N. Wacholder, "Identification of nonliteral language in social media: A case study on sarcasm," *Journal of the Association for Information Science and Technology*, vol. 67, no. 11, pp. 2725–2737, 2016.

[5] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.

[6] F. Barbieri, F. Ronzano, and H. Saggion, "UPF-taln: SemEval 2015 Tasks 10 and 11. Sentiment analysis of literal and figurative language in Twitter," in *Proceedings of the International Workshop on Semantic Evaluation (SemEval 2015)*, 2015.

[7] C. Van Hee, E Lefever, and V. Hoste, "LT3: sentiment analysis of figurative tweets: piece of cake# NotReally," in *Proceedings of the International Workshop on Semantic Evaluations (SemEval 2015)*, 2015.

[8] T. Hercig and L. Lenc, "The impact of figurative language on sentiment analysis," in *Proceedings of the Recent Advances in Natural Language Processing (RANLP)*, 2017.

[9] R.-A. Potamias, G. Siolas, and A. Stafylopatis, "A robust deep ensemble classifier for figurative language detection," in *Proceedings of the International Conference on Engineering Applications of Neural Networks (EANN)*. Springer, 2019.

[10] R. A. Potamias, G. Siolas, and A.-G. Stafylopatis, "A Transformer-based approach to irony and sarcasm detection," *Neural Computing and Applications*, vol. 32, no. 23, pp. 17309–17320, 2020.

[11] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized BERT pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

[12] T. Chen, I. Goodfellow, and J. Shlens, "Net2Net: Accelerating learning via knowledge transfer," *arXiv preprint arXiv:1511.05641*, 2015.

[13] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio, "Fitnets: Hints for thin deep nets," *arXiv preprint arXiv:1412.6550*, 2014.

[14] N. Passalis and A. Tefas, "Unsupervised knowledge transfer using similarity embeddings," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 3, pp. 946–950, 2018.

[15] S. I. Mirzadeh, M. Farajtabar, A. Li, N. Levine, A. Matsukawa, and H. Ghasemzadeh, "Improved knowledge distillation via teacher assistant," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.

[16] M. Phuong and C. Lampert, "Towards understanding knowledge distillation," in *Proceedings of the International Conference on Machine Learning*, 2019.

[17] L. Saglietti and L. Zdeborová, "Solvable model for inheriting the regularization through knowledge distillation," *arXiv preprint arXiv:2012.00194*, 2020.

[18] S. Lai, L. Xu, K. Liu, and J. Zhao, "Recurrent Convolutional Neural Networks for text classification," in *Proceedings of AAAI conference on artificial intelligence*, 2015.

[19] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," *arXiv preprint arXiv:1802.05365*, 2018.

[20] D. Cer, Y. Yang, S. Kong, N. Hua, N. Limtiaco, R. John, N. Constant, M. Guajardo-Céspedes, S. Yuan, and C. Tar, "Universal sentence encoder," *arXiv preprint arXiv:1803.11175*, 2018.

[21] S. I. Wang and C. D. Manning, "Baselines and bigrams: Simple, good sentiment and topic classification," in *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2012.

[22] A. Joulin, E. Grave, P. Bojanowski, M. Douze, H. Jégou, and T. Mikolov, "FastText.zip: Compressing text classification models," *arXiv preprint arXiv:1612.03651*, 2016.

[23] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le, "Xlnet: Generalized autoregressive pretraining for language understanding," *Proceedings of the Advances in Neural Information Processing Systems (NIPS)*, 2019.

[24] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pretraining of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[25] C. Ozdemir and S. Bergler, "CLaC-SentiPipe: Semeval2015 Subtasks 10 b, e, and Task 11," in *Proceedings of the International Workshop on Semantic Evaluation (SemEval 2015)*, 2015.

[26] R. Kiran, P. Kumar, and B. Bhasker, "OSLCFit (Organic Simultaneous LSTM and CNN Fit): a novel deep learning-based solution for sentiment polarity classification of reviews," *Expert Systems with Applications*, vol. 157, pp. 113488, 2020.

[27] J. Ling and R. Klinger, "An empirical, quantitative analysis of the differences between sarcasm and irony," in *Proceedings of the European Semantic Web Conference*. Springer, 2016.