future internet

MDPI

Article

Polarity and Subjectivity Detection with Multitask Learning and BERT Embedding

Ranjan Satapathy [1], Shweta Rajesh Pardeshi [2] and Erik Cambria [3,*]

[1] Graphene AI, 28 Genting Ln, Singapore 349585, Singapore; ranjan@graphenesvc.com
[2] Granular AI, Mumbai 410206, India; shweta@granular.ai
[3] School of Computer Science and Engineering, Nanyang Technological University, Singapore 639798, Singapore
* Correspondence: cambria@ntu.edu.sg

Abstract: In recent years, deep learning-based sentiment analysis has received attention mainly because of the rise of social media and e-commerce. In this paper, we showcase the fact that the polarity detection and subjectivity detection subtasks of sentiment analysis are inter-related. To this end, we propose a knowledge-sharing-based multitask learning framework. To ensure high-quality knowledge sharing between the tasks, we use the Neural Tensor Network, which consists of a bilinear tensor layer that links the two entity vectors. We show that BERT-based embedding with our MTL framework outperforms the baselines and achieves a new state-of-the-art status in multitask learning. Our framework shows that the information across datasets for related tasks can be helpful for understanding task-specific features.

Keywords: multitask learning; polarity detection; subjectivity detection; deep learning; market intelligence

Citation: Satapathy, R.; Pardeshi, S.R.; Cambria, E. Polarity and Subjectivity Detection with Multitask Learning and BERT Embedding. *Future Internet* **2022**, *14*, 191. https://doi.org/10.3390/fi14070191

Academic Editor: Young Im Cho

Received: 6 May 2022
Accepted: 16 June 2022
Published: 22 June 2022

1. Introduction

In order to accurately extract and manipulate text meaning, a Natural language processing (NLP) system must have access to a notable amount of knowledge about the world and the domain of discourse. With the data, the NLP system is reliant on the extraction of meaning from text that has resulted in an exponential interest in NLP tasks such as sentiment analysis [1], microtext normalization [2], and others. The rise of e-commerce has given rise to reliance on sentiment analysis for market research. The tasks in NLP are interrelated and could benefit from sharing each other's learning. Sentiment analysis is a subtask under NLP that predicts different affect states. Subjectivity detection and polarity detection are subtasks under sentiment analysis. Subjectivity detection aims to remove 'factual' or 'neutral' content, i.e., objective text that does not contain any opinion. Polarity detection aims to differentiate the opinion into 'positive' and 'negative'.

Multitask learning (MTL) [3] has displayed remarkable success in the field of image recognition. This success can be primarily attributed to learning-shared representations from multiple supervisory tasks. MTL acts as a regularizer by introducing an inductive bias and it reduces the risk of overfitting of the model. Extending MTL's success to NLP, we propose an MTL model to extract both sentiment (i.e., positive or negative) and subjectivity (i.e., subjective or objective) of a sentence. In multitask framework, we aim to leverage the inter-dependence of these two tasks to increase the confidence of individual tasks in prediction; e.g., information about sentiment can help in the prediction of subjectivity and vice-versa. For sentence-level classification, the neutral class cannot be ignored because an opinion document can contain many sentences that express no opinion or sentiment. A sentence is opinionated if it expresses or implies a positive or negative sentiment. A sentence is non-opinionated if it expresses or implies a neutral sentiment.

Recently, Bidirectional Encoder Representations from Transformers (BERT) [4] has caused a stir in the NLP community by presenting state-of-the-art results in a wide variety

of NLP tasks. BERT's critical technical innovation is applying the bidirectional training of Transformer, a popular attention model, to language modeling. In our proposed framework, we use BERT as an embedding for our input sentences. The results show that our MTL framework surpasses baselines in both tasks. We evaluate our proposed approach on subjective and polarity datasets [1,5,6]. Our famework can be used to extract insights from text or review to provide insights for market research.

The rest of the paper is structured as follows: Section 2 introduces related works about sentiment analysis and multitask learning; Section 3 presents the proposed model, and Section 4 conducts experiments on multitask learning; Section 5 concludes the paper.

2. Related Work

This section introduces related work in multitask learning, sentiment analysis, and subjectivity detection.

Sentiment analysis, in particular, has seen growth in multitasking usage [7,8]. Ref. [9] uses MTL for adversarial text classification, mitigating the shared and private latent feature spaces from interfering with each other. Ref. [10] proposes an MTL approach that allows joint training of the primary and auxiliary tasks, improving the performance of rumour verification.

Subjectivity detection is an NLP task that consists of differentiating objective data ("factual" or "neutral" content) from subjective data (opinions). Mishra et al. (2018) [11] propose a multitask deep neural framework for document-level sentiment analysis that learns to predict the overall sentiment expressed in the given input document as the primary task and simultaneously learning to predict human gaze behavior and auxiliary linguistic tasks such as part-of-speech and syntactic properties of words in the document as the secondary task. Extracting subjective text segments poses a tremendous challenge that only a few works have attempted. Ref. [6] applied a graph-min-cut based technique to separate the subjective portion of the text from the irrelevant objective portions. Ref. [12] introduced a novel architecture for filtering out neutral content in a time- and resource-effective manner.

Glove [13] is a word vector technique that leverages both global and local statistics of a corpus in order to come up with a principled loss function that uses both these. We finetuned Glove with the dataset and created an embedding of **L X 300** dimension where L is the maximum length of the input text.

The authors in [12] proposed a Bayesian network-based extreme learning machine (BNELM) model. BNELM augments the standard recurrent neural network structure to generate a predictor that can take advantage of the beneficial properties of extreme learning machines and Bayesian networks. Emotion recognition is a task very close to sentiment classification. Rashkin et al. (2019) [14] used BERT to detect emotions and applied it to a dialogue system.

The authors in [15] propose an evaluation of MTL on semantic sequence prediction on data-dependent conditions. They derive attribute of datasets that make them favorable for MTL by comparing performance with information-theoretical metrics of the label frequency distribution. Generic MTL [3,16] has a rich history in machine learning. It has widespread applications in other fields, such as genomics [17], NLP [18–21], and computer vision [22–24]. The MTL paradigm provides an effective platform for achieving generalization. The inspiration is that if the tasks are related, the model can learn jointly, taking into account the shared information, which is expected to increase its generalization ability. Various tasks can exploit the inter-relatedness to improve individual performance through a shared representation. Overall, it provides three principal benefits over the single-task learning paradigm:

1. It helps in achieving generalization for multiple tasks;
2. Each task improves its performance in association with the other participating tasks;
3. Offers reduced complexity because a single system can handle multiple problems or tasks simultaneously.

The performance improvement from MTL [3] is due to the extra information in the training signals of related tasks:

- **Implicit data augmentation:** Learning only one task carries the risk of overfitting that task while learning jointly enables the model to obtain a better representation by averaging noise patterns. MTL effectively increases the sample size we are using to train our model by sharing the learnt features.
- **Attention focusing:** If the data are insufficient and high-dimensional, it can be challenging for a model to distinguish between relevant and irrelevant features.
- **Eavesdropping:** We can allow the model to eavesdrop through MTL; i.e., tasks challenging to learn for one model are learnt by the other model.
- **Representation bias:** MTL biases the model to prefer representations that other tasks also prefer, which helps the model to generalize new tasks in the future.

Subjectivity classification models classifies sentences into two classes: subjective and objective [25]. An objective sentence states some factual information, whereas a subjective sentence expresses personal feelings, views, judgments, or beliefs. We explore this relation in our MTL-based framework. Polarity classification models classifies sentences into two classes: positive and negative. If a sentence is classified as subjective or opinionated, we determine whether it expresses a positive or negative opinion, making sentiments and subjectivity closely related. We solve two tasks with a single network. Given a sentence sentence s_i , we assign it both a sentiment tag (positive/negative) and a subjective tag (subjective/objective), see Table 1.

Table 1. Statistics of the datasets used in this paper after preprocessing and output of keras tokenizer.

Dataset	Train	Dev	Test	Max Length	Avg. Length	Vocabulary
POL	7.2K	800	2K	40	15	16.5k
SUBJ	7.2K	800	2K	85	17	18.5k

3. Proposed Multitask Learning (MTL) Based Framework

MTL is an approach to inductive transfer that improves generalization by using the domain information contained in the training signals of related tasks as an inductive bias. We use MTL, where a single framework performs two classification tasks simultaneously, i.e., subjectivity detection and polarity classification simultaneously. We assign a sentiment tag (pos or neg) and a subjectivity tag (yes or no) to each text, see Algorithm 1.

The methods of MTL have often been subdivided into two groups with a familiar dichotomy: hard parameter sharing vs. soft parameter sharing [26]. In hard parameter sharing, the model weights are shared between multiple tasks whereas in soft parameter sharing, all tasks have task specific models with separate weights. We have used hard parameter sharing, which is the most commonly used approach to MTL. It is generally applied by sharing the hidden layers between all tasks while keeping several task-specific output layers. In our experiments, we have used two different datasets and shared their information via Neural Tensor Network (NTN) [27]. NTN infuses knowledge sharing capabilities between individual tasks. Our model (Figure 1) shows that the information across datasets for related tasks can be helpful for understanding task-specific features.

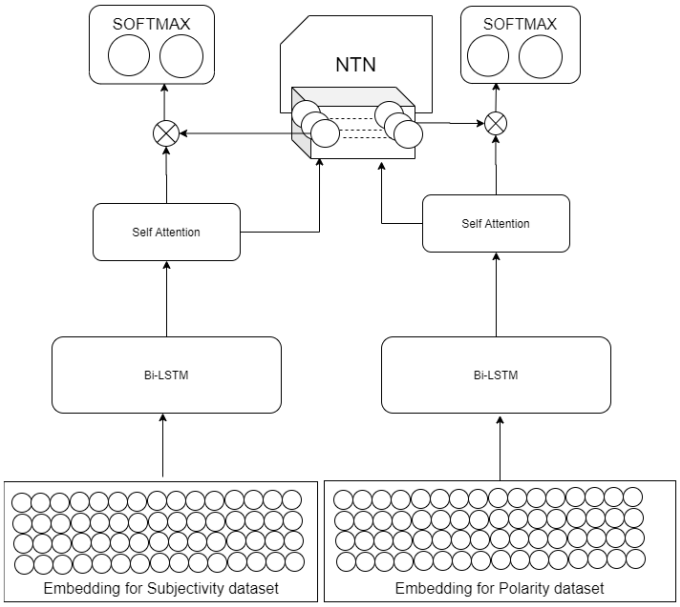


Figure 1. Framework of our proposed model.

Algorithm 1: Multitask BERT based Sentiment and Subjectivity

Result: Class

1. $E_{BERT_{T_i}}$ = BERT(S);
2. L_{T_i} = BILSTM($E_{BERT_{T_i}}$);
3. F_{T_i} = TDFC(L_{T_i});
4. F_{T_i} = Drop(F_{T_i});
5. SA_{T_i} = Attention(F_{T_i});
6. D_{T_i} = FC(SA_{T_i});
7. D_{T_i} = Drop(D_{T_i});
8. Fn_{T_i} = Flatten(D_{T_i});
9. X_{T_i} = FC(Fn_{T_i});
10. N = NTN($[Fn_{T_1} \oplus Fn_{T_2}]$);
11. $C_{T_i} = X_{T_i} \oplus N$;
12. Y_{T_i} = FC(C_{T_i});

Result: BERT Embedding initialization;

1. Token = BERTTokenizer(S);
2. id = Map(Token, ID)
3. S-new = Pad(S, maxlen)
4. embedding = transformer(S-new)

3.1. Embedding

We used two different feature extraction for our experimental setup, namely, BERT- and Global Vectors (Glove)-based embeddings. Glove embedding acts as a baseline for our proposed MTL setting.

BERT Embedding

We use pre-trained BERT [28] and computed sentence-level BERT embeddings. We used padding to normalize the variable-length input sentences to a fixed-length. The dimension of each embedding is **L x 768**, where L is the maximum length of the input text. In our experiments, we use the BERT base model with 12 encoder layers (i.e., transformer blocks) and feed-forward networks with 768 hidden units and 12 attention heads to capture

transformer-based contextual word representations from both directions. The "Attention Mask" in BERT is simply an array of 1s and 0s indicating the presence or absence of padding tokens. This mask tells the "Self-Attention" mechanism in BERT not to include these PAD tokens in its interpretation of the sentence. Our proposed model uses the output of the final layer of BERT.

3.2. Bidirectional LSTM Layer

To obtain the input's context-rich representation, we fed the embeddings to the Bidirectional Long Short-Term Memory (biLSTM) Layer of size 128. The output of the LSTM layer is used for both subjectivity and sentiment analyses.

In the next layer, we fed the LSTM output to obtain two matrices using two fully connected layers, which are input into two different tasks.

$$i_t = \sigma(W_{x^i} x_t) + W_{hi} h_{t-1} + W_{ci} c_{t-1} + B_i$$
$$f_t = \sigma(W_{xf} x_t + W_{hf} h_{t-1} + W_{cf} c_{t-1} + b_f)$$
$$c_t = f_t c_{t-1} + i_t tanh(W_{xc} x_t + W_{hc} h_{t-1} + b_c)$$
$$o_t = \sigma(W_{xo} x_t + W_{ho} h_{t-1} + W_{co} c_t + b_o)$$
$$h_t = o_t tanh(c_t)$$

3.3. Self Attention Network

Self-attention is an attention mechanism [29] associating the various positions of a single sequence to compute its representation. We use the self-attention mechanism in the next layer as it prioritizes the words necessary for classification:

$$P = \tanh(H_* W^{ATT}), \quad (1)$$
$$\alpha = softmax(P^T W^{\alpha}), \quad (2)$$
$$s_* = \alpha H_*^T, \quad (3)$$

where $W^{ATT} \in \mathbb{R}^{D_t \times 1}$, $W^{\alpha} \in \mathbb{R}^{L \times L}$, $P \in \mathbb{R}^{L \times 1}$, and $s_* \in \mathbb{R}^{D_t}$. In Equation (2), $\alpha \in [0,1]^L$ provides the relevance of words for the task multiplied by Equation (3) by the context-aware word representations in H_*.

The output of the attention layer is flattened before feeding to the next layer, which is the Neural Tensor Network (NTN).

3.4. Neural Tensor Network (NTN)

We use the Neural Tensor Network [27] model to combine both tasks. NTN consists of a bilinear tensor layer that links the two entity vectors. In standard neural networks, the entity vectors are simply concatenated, whereas NTN can relate these vectors multiplicatively:

$$s_{NTN} = \tanh(s_{subj} T^{[1:D_{ntn}]} s_{pol}^T + (s_{subj} \oplus s_{pol}) W + b),$$

where $T \in \mathbb{R}^{D_{ntn} \times D_t \times D_t}$, $W \in \mathbb{R}^{2D_t \times D_{ntn}}$, $b, s_+ \in \mathbb{R}^{D_{ntn}}$, and $\oplus$ stands for concatenation. The vector s_+ contains shared information of both sentiment and subjectivity.

3.5. Classification

For the two tasks, we use two different softmax layers (task specific) for classifications detailed below.

3.5.1. Sentiment Classification

We use the output of NTN and concatenate it with the output of self-attention to determine the sentiment of the input text using the softmax layer.

$$\mathcal{P}_{pol} = \text{softmax}(s_{pol}\, W_{pol}^{softmax} + b_{pol}^{softmax}),$$
$$\hat{y}_{pol} = \underset{j}{\text{argmax}}(\mathcal{P}_{pol}[j]),$$

where $W_{pol}^{softmax} \in \mathbb{R}^{D_t \times C}$, $b_{pol}^{softmax} \in \mathbb{R}^{C}$, $\mathcal{P}_{pol} \in \mathbb{R}^{C}$, j is the class value (0 for negative and 1 for positive), and $\hat{y}_{sen}$ is the estimated class value.

3.5.2. Subjectivity Classification

We use only the output of the attention layer as sentence representation for subjectivity classification since subjectivity detection is a subtask of sentiment analysis. We use the softmax layer to obtain the final output:

$$\mathcal{P}_{subj} = \text{softmax}((s_{subj} \oplus s_{+})\, W_{subj}^{softmax} + b_{subj}^{softmax}),$$
$$\hat{y}_{subj} = \underset{j}{\text{argmax}}(\mathcal{P}_{subj}[j]),$$

where $W_{subj}^{softmax} \in \mathbb{R}^{(D_t + D_{ntn}) \times C}$, $b_{subj}^{softmax} \in \mathbb{R}^{C}$, $\mathcal{P}_{subj} \in \mathbb{R}^{C}$, j is the class value (0 for objective and 1 for subjective), and $\hat{y}_{subj}$ is the estimated class value.

4. Experiments

This section introduces the experiments performed and compares the experiments with the baselines on both single task and multitask settings available for polarity and subjective detection tasks.

4.1. Dataset

We have used the same number of sentences for both the models. Both the datasets [1,5,6] are balanced with an equal number of classes as well:

1. POL : The dataset contains 5331 positive and 5331 negative processed sentences. We selected 5000 sentences from each class randomly, i.e., 5000 positive and 5000 negative sentences.
2. SUBJ : The dataset contains 5000 subjectively and 5000 objectively processed sentences.

Both the datasets can be downloaded from here (https://www.cs.cornell.edu/people/pabo/movie-review-data/ accessed on 5 May 2022).

4.2. Baselines and Model Variants

We implemented three different models and compare our results with six state-of-the art models, as described in Table 2.

Table 2. Accuracy comparision of different models.

	Framework	Subjective	Polarity
Baselines	SenticNet 6 [30]	-	92.8%
	Subjectivity detector [6]	92%	-
	AdaSent [31]	**95.5%**	83.1%
	CNN+MCFA [32]	**95.2%**	83.2%
	Multitask uniform layer [33]	93.4%	87.1%
	Multitask shared-layer [33]	94.1%	87.9%
	$MTL_{sharedNTNGlove}$	**92.3%**	**92.1%**
BERT Embedding	$BILSTM_{pol}$	-	77.5%
	$BILSTM_{subj}$	93.5%	-
	$MTL_{sharedNTN}$	**95.1%**	**94.6%**

4.3. Hyperparameters and Training

1. Trainable parameters for the MTL model: 14,942,052.
2. Trainable parameters for the individual models: 1,923,746.

ADAM [34] is an optimization algorithm that can be applied instead of the classical stochastic gradient descent algorithm to update network weights based on training data iteratively. The code is run for 20 epochs with 64 GB RAM and 32 GB Nvidia v100 Tesla.

$$\theta = \{U^{[z,r,h]}, W^{[z,r,h]}, W_*, b_*, W^{ATT}, W^{\alpha}, T, W, b, W_*^{softmax}, b_*^{softmax}\}.$$

We use categorical crossentropy (J_*; * is $subj$ or pol) as the loss function:

$$J_* = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=0}^{C-1} y_{ij}^{*}\ \log \mathcal{P}_{*i}[j],$$

where N is the number of samples which is 10K, i is the index of a sample, j is the class value, and the following is the case.

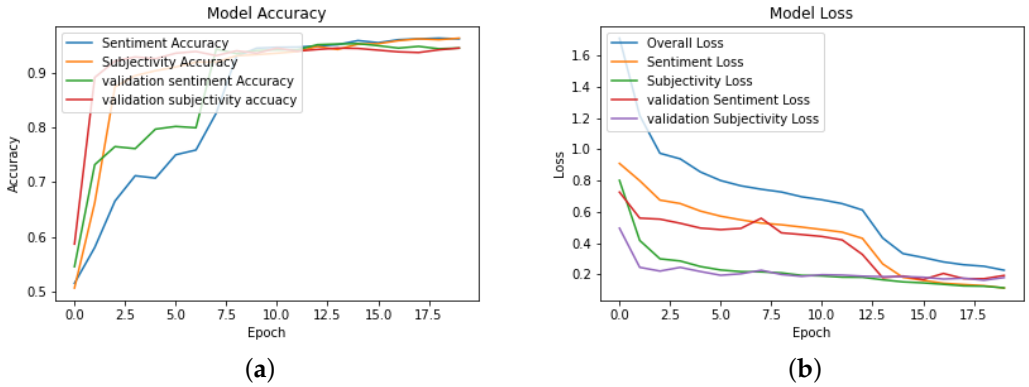
$$y_{ij}^{*} = \begin{cases} 1, & \text{if expected class value of sample } i \text{ is } j, \\ 0, & \text{otherwise.} \end{cases}$$

4.4. Results and Discussions

We divided the dataset into train, validation, and test set for our experiments. The dataset was divided into train and test as 80:20 with random shuffling. The training dataset was further divided into train and validation as 90:10. We used the ADAM algorithm as an optimizer with categorical cross-entropy to calculate the loss.

We implemented single task frameworks and multitask frameworks to compare the performance. The models are trained on two different datasets, so the output shows if the tasks are related; they can share essential knowledge, which can benefit both task-specific outputs. We observed that, with the addition of NTN layer, which fuses the representations of polarity and subjectivity, our MTL model performed better. In the polarity detection case, NTN improved the accuracy by 15% with a single task framework and 10% with the MTL framework, whereas in the subjectivity detection case, NTN improved accuracy by 2–3% with a single task framework and no improvements were found for MTL frameworks. The comparisions with baseline also suggests an improvement in performance across both tasks. Table 2 depicts the comparision of our results to baselines, a single task and multitask frameworks. The loss and accuracy graphs of our proposed BERT embedding based MTL

network is shown in Figure 2. We have also released our code and datasets on Github (https://github.com/shwetapardeshi1/Polarity-and-sentiment-detection accessed on 5 May 2022).



(**a**) (**b**)

Figure 2. MTL based on BERT embedding. (**a**) Accuracy graph. (**b**) Loss graph.

5. Conclusions

Multitask learning often aids in improving the performance of similar tasks. Related tasks often have interdependence on each other and function better when solved in a joint framework. In this paper, we present a BERT-based MTL framework that combines sentiment and subjective detection. In the current paper, we proposed a multilayer multitask LSTM for the main task of polarity detection and subjectivity detection. We used a Neural Tensor Network to combine the task to improve the functionality of individual tasks. The polarity task in MTL improved by at least 15% when compared to single-task performance. In comparison, the subjectivity detection task has improved only by 2–4% in a similar setting. As shown in the experiments, BERT embedding for individual tasks are not as good as the MTL setting. As the tasks are related, so are the features that the model learnt to classify. However, the key takeaway from our experiment is that linguistic tasks such as polarity and subjectivity detection are related tasks and, when trained on different datasets, they still surpass the baselines.

Author Contributions: Conceptualization, R.S. and S.R.P.; Data curation, S.R.P.; Methodology, R.S. and S.R.P.; Software, S.R.P.; Supervision, R.S. and E.C.; Validation, S.R.P.; Writing—original draft, R.S.; Writing—review & editing, S.R.P. and E.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Not Applicable, the study does not report any data.

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

References

1. Pang, B.; Lee, L. Seeing Stars: Exploiting Class Relationships For Sentiment Categorization With Respect To Rating Scales. *arXiv* **2005**, arXiv:cs/0506075.
2. Satapathy, R.; Cambria, E.; Nanetti, A.; Hussain, A. A review of shorthand systems: From brachygraphy to microtext and beyond. *Cogn. Comput.* **2020**, *12*, 778–792. [CrossRef]
3. Caruana, R. Multitask learning. *Mach. Learn.* **1997**, *28*, 41–75. [CrossRef]
4. Devlin, J.; Chang, M.W.; 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, Volume 1 (Long and Short Papers), Minneapolis, MN, USA, 2–7 June 2019; pp. 4171–4186.
5. Pang, B.; Lee, L.; Vaithyanathan, S. Thumbs Up? Sentiment Classification Using Machine Learning Techniques. *arXiv* **2002**, arXiv:cs/0205070.
6. Pang, B.; Lee, L. A Sentimental Education: Sentiment Analysis Using Subjectivity. *arXiv* **2004**, arXiv:cs/0409058.

7. Balikas, G.; Moura, S.; Amini, M.R. Multitask learning for fine-grained twitter sentiment analysis. In Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, Shinjuku, Tokyo, Japan, 7–11 August 2017; pp. 1005–1008.
8. Majumder, N.; Poria, S.; Peng, H.; Chhaya, N.; Cambria, E.; Gelbukh, A. Sentiment and sarcasm classification with multitask learning. *IEEE Intell. Syst.* **2019**, *34*, 38–43. [CrossRef]
9. Liu, P.; Qiu, X.; Huang, X.J. Adversarial Multi-task Learning for Text Classification. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Vancouver, BA, Canada, 30 July–4 August 2017; pp. 1–10.
10. Kochkina, E.; Liakata, M.; Zubiaga, A. All-in-one: Multi-task Learning for Rumour Verification. In Proceedings of the 27th International Conference on Computational Linguistics, Santa Fe, NM, USA, 20–26 August 2018; pp. 3402–3413.
11. Mishra, A.; Tamilselvam, S.; Dasgupta, R.; Nagar, S.; Dey, K. Cognition-Cognizant Sentiment Analysis With Multitask Subjectivity Summarization Based on Annotators' Gaze Behavior. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; pp. 5884–5891.
12. Chaturvedi, I.; Ragusa, E.; Gastaldo, P.; Zunino, R.; Cambria, E. Bayesian network based extreme learning machine for subjectivity detection. *J. Frankl. Inst.* **2018**, *355*, 1780–1797. [CrossRef]
13. Pennington, J.; Socher, R.; Manning, C.D. GloVe: Global Vectors for Word Representation. In Proceedings of the Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–29 October 2014; pp. 1532–1543.
14. Rashkin, H.; Smith, E.M.; Li, M.; Boureau, Y.L. Towards Empathetic Open-domain Conversation Models: A New Benchmark and Dataset. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Florence, Italy, 28 July–2 August 2019; pp. 5370–5381.
15. Alonso, H.M.; Plank, B. When is multitask learning effective? Semantic sequence prediction under varying data conditions. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers, Valencia, Spain, 3 April 2017; pp. 44–53.
16. Stein, C. Inadmissibility of the Usual Estimator for the Mean of a Multivariate Normal Distribution. In Proceedings of the Third Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Contributions to the Theory of Statistics, Berkeley, CA, USA, 1 January 1956; pp. 197–206.
17. Obozinski, G.; Taskar, B.; Jordan, M.I. Joint covariate selection and joint subspace selection for multiple classification problems. *Stat. Comput.* **2010**, *20*, 234–252. [CrossRef]
18. Collobert, R.; Weston, J. A unified architecture for natural language processing: Deep neural networks with multitask learning. In Proceedings of the 25th international conference on Machine learning, Helsinki, Finland, 5–9 July 2008; pp. 160–167.
19. Collobert, R.; Weston, J.; Bottou, L.; Karlen, M.; Kavukcuoglu, K.; Kuksa, P. Natural language processing (almost) from scratch. *J. Mach. Learn. Res.* **2011**, *12*, 2493–2537.
20. Liu, X.; Gao, J.; He, X.; Deng, L.; Duh, K.; Wang, Y.Y. Representation Learning Using Multi-Task Deep Neural Networks for Semantic Classification and Information Retrieval. In Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, CO, USA, 4 June 2015; pp. 912–921.
21. Bansal, T.; Belanger, D.; McCallum, A. Ask the gru: Multi-task learning for deep text recommendations. In Proceedings of the 10th ACM Conference on Recommender Systems, Boston, MA, USA, 15 September 2016; pp. 107–114.
22. Yim, J.; Jung, H.; Yoo, B.; Choi, C.; Park, D.; Kim, J. Rotating your face using multi-task deep neural network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 676–684.
23. Torralba, A.; Murphy, K.P.; Freeman, W.T. Sharing visual features for multiclass and multiview object detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2007**, *29*, 854–869. [CrossRef] [PubMed]
24. Misra, I.; Shrivastava, A.; Gupta, A.; Hebert, M. Cross-stitch networks for multi-task learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NA, USA, 27–30 June 2016; pp. 3994–4003.
25. Wiebe, J.; Bruce, R.; O'Hara, T.P. Development and use of a gold-standard data set for subjectivity classifications. In Proceedings of the 37th annual meeting of the Association for Computational Linguistics, College Park, MD, USA, 20–26 June 1999; pp. 246–253.
26. Crawshaw, M. Multi-task learning with deep neural networks: A survey. *arXiv* **2020**, arXiv:2009.09796.
27. Socher, R.; Chen, D.; Manning, C.D.; Ng, A. Reasoning with neural tensor networks for knowledge base completion. *Adv. Neural Inf. Process. Syst.* **2013**, *26*, 926–934.
28. Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; Funtowicz, M.; et al. HuggingFace's Transformers: State-of-the-art Natural Language Processing. *arXiv* **2019**, arXiv:1910.03771.
29. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, 5998–6008. [CrossRef]
30. Cambria, E.; Li, Y.; Xing, F.; Poria, S.; Kwok, K. SenticNet 6: Ensemble Application of Symbolic and Subsymbolic AI for Sentiment Analysis. In Proceedings of the CIKM, Virtual Event, Ireland, 19–23 October 2020; pp. 105–114.
31. Zhao, H.; Lu, Z.; Poupart, P. Self-adaptive hierarchical sentence model. In Proceedings of the 24th International Conference on Artificial Intelligence, Buenos Aires, Argentina, 25–31 July 2015; pp. 4069–4076.
32. Amplayo, R.K.; Lee, K.; Yeo, J.; Hwang, S.W. Translations as additional contexts for sentence classification. In Proceedings of the 27th International Joint Conference on Artificial Intelligence, Stockholm, Sweden, 13–19 July 2018; pp. 3955–3961.

33. Liu, P.; Qiu, X.; Huang, X. Recurrent neural network for text classification with multi-task learning. In Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, New York, NY, USA, 9–15 July 2016; pp. 2873–2879.
34. Kingma, D.P.; Ba, J.L. Adam: A method for stochastic optimization. In Proceedings of the International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, 7–9 May 2015; pp. 2873–2879.