SLPL-Sentiment at SemEval-2022 Task 10: Making Use of Pre-Trained Model's Attention Values in Structured Sentiment Analysis

Sadrodin Barikbin
Speech and Language Processing Lab
Sharif University of Technology

barikbin@ce.sharif.edu

Abstract

Sentiment analysis is a useful problem which could serve a variety of fields from business intelligence to social studies and even health studies. Using SemEval 2022 Task 10 formulation of this problem and taking sequence labelling as our approach, we propose a model which learns the task by finetuning a pretrained transformer, introducing as few parameters (150k) as possible and making use of precomputed attention values in the transformer. Our model improves shared task baselines on all task datasets.

1 Introduction

Sentiment analysis has many applications in different fields. From social and political studies (Rodríguez-Ibáñez et al., 2021) to business intelligence and even health studies (Alamoodi et al., 2021). SemEval 2022 Task 10 (Barnes et al., 2022) formulates this problem as to extract a graph of sentiment-related entities and names that, **Structured Sentiment Analysis**. An example of such graph is depicted in figure2 Specifically given an input sentence, one should extract a list of quadruples. Each quadruple consists of a sentiment expression, a target expression, a holder expression and the polarity. SemEval 2022 Task 10 has two subtasks, monolingual and cross-lingual. We participated in the first track. Shared task data is consisted of seven datasets in English, Spanish, Norwegian, Catalan and Basque. We trained and evaluated our model on each of these datasets.[1]

2 Related Work

Structured sentiment analysis aims to extract sentiment, target and holder expressions along with their relations and also sentiment polarities. (Barnes et al., 2021) solves this problem by taking dependency graph parsing approach. It uses BiLSTM and multilingual BERT (Devlin et al., 2019) to encode input sentence and a neural dependency parser to jointly predict expressions and relations. Every sentiment, target or holder expression in the sentence would become a star-shaped subtree in the output graph, with last token of expression as its root and its edges labelled with type of entity. Relation between a sentiment expression and a target or holder expression is also represented with an edge from the former's root to that of the latter. This work evaluates its model on five datasets in four languages namely, NoReC$_{\text{Fine}}$ (Øvrelid et al., 2020), Multibooked$_{\text{EU}}$, Multibooked$_{\text{CA}}$ (Barnes et al., 2018), MPQA (Wiebe et al., 2005) and Darmstadt$_{\text{Unis}}$ (Toprak et al., 2010). Its main evaluation metrics are Targeted F1 and Sentiment Graph F1. First one measures exact prediction of target expression and polarity, and the second one measures exact match at graph level, weighting overlap between gold and predicted entities, averaged across three entity types.

A less comprehensive formulation of sentiment analysis is **End2End** sentiment analysis which aims to predict target and sentiment expression along with polarity but does not consider relation between expressions. (He et al., 2019) uses this formulation. Taking sequence labelling as its approach, it predicts target expression together with sentiment expression by one module and polarity by another one. Both modules along with a feature extraction module make use of a CNN to make their specific and shared latent vectors respectively. Furthermore, polarity prediction module has a self-attention layer which gets predicted probability of tokens' being in sentiment expression from the other module and incorporates it in computing attention values so as to tokens with higher proba-

[1] Our code is available at https://github.com/sadra-barikbin/novel-solutions-for-sentiment-analysis

Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022), pages 1382 - 1388
July 14-15, 2022 ©2022 Association for Computational Linguistics

bility get more attention. Beside its main task, this work employs two auxiliary document-level tasks to enjoy the benefits of multi-task learning. This work also uses a mechanism called message passing in which, predicted probabilities of all modules is iteratively fed back into model to make the model like a RNN. For data, they use review datasets from (Pontiki et al., 2014) and (Pontiki et al., 2015) which are annotated with sentiment expressions by (Wang et al., 2016, 2017). Among their evaluation metrics, F1-I is equivalent with Targeted-F1 of (Barnes et al., 2021).

(Chen and Qian, 2020), similarly attempts to solve end2end sentiment analysis and takes sequence labelling approach. It introduces a layer called RACL which consists of three separate but interconnected modules for sentiment and target extraction and polarity prediction. Each module makes a module-specific representation of input using a CNN and information is exchanged among them through attention mechanism. The work uses a stack of RACL layers as its model in which features in a layer is fed to upper layer and finally prediction is done by average pooling over prediction of all layers. This work has used same data and evaluation metrics as (He et al., 2019).

Finally (Li et al., 2019) solves **targeted sentiment analysis** which aims to extract only target expression and polarity. To this end, it has used BERT (Devlin et al., 2019) along with a task specific layer. Here BERT makes a contextual and rich representation of input words and the task-specific layer performs sequence labelling. For this layer, different types such as fully connected, CRF (Lafferty et al., 2001), GRU and self-attention layer are examined that the last two outperformed the others.

3 Datasets and Evaluation

Shared task is on seven datasets in five languages. We trained and evaluated our model on each one separately. General and detailed information of datasets is shown in table1 and table2 respectively. **NoReC**$_{\text{Fine}}$ is norwegian professional reviews in multiple domains. **MultiB**$_{\text{CA}}$, **MultiB**$_{\text{EU}}$, **Opener**$_{\text{EN}}$ and **Opener**$_{\text{ES}}$ (Agerri et al., 2013) are hotel reviews in Catalan, Basque, English and Spanish respectively. **Darmstadt**$_{\text{Unis}}$ consists of English online university reviews and finally **MPQA** contains annotated news articles in English. Evaluation metric of the task is ***Sentiment Graph F1***. Prediction and gold answer of problem are a list of quadroples $q = (q_s, q_t, q_h, q_{pol})$ in which the first three entities are sets of tokens for sentiment, target, and holder expressions respectively. q_pol is polarity with value among $Negative, Neutral, Positive$. Match score of two given source and target quadroples namely, $score(src, tgt)$ is as follows.

$$\frac{\sum_{e\in\{s,t,h\}} \frac{|src_e \cap tgt_e|}{|src_e|}}{3} * 1\{src_{pol} = tgt_{pol}\} \quad (1)$$

As it can be seen, this is a weighted match over amount of overlap between entities, averaged across three entity types. Polarities should be strictly equal. Denominator $|src_e|$ is for score to be comparable with another one having different source quadrople. As target and holder expressions could be empty, $|src_e|$ is replaced with 1 in the case src_e is empty. $|src_e \cap tgt_e|$ is also set to 1 when $|src_e|$ and $|tgt_e|$ are empty. Given N input sentences, $pred_n$ as the list of predicted quadroples for n-th sentence and $gold_n$ as its gold counterpart, ***Precision*** is computed as follows.

$$\frac{\sum_{n=1}^{N} \sum_{p\in pred_n} \max_{q\in gold_n} score(p, q)}{\sum_{n=1}^{N} |pred_n|} \quad (2)$$

max is to select score of best matching gold quadrople for a given predicted one. Similarly, ***Recall*** is computed as follows.

$$\frac{\sum_{n=1}^{N} \sum_{q\in gold_n} \max_{p\in pred_n} score(q, p)}{\sum_{n=1}^{N} |gold_n|} \quad (3)$$

Finally, ***Sentiment Graph F1*** is measured as follows.

$$GraphF1 = \frac{2 * Precision * Recall}{Precision + Recall} \quad (4)$$

4 System Description

Our model is comprised of a base pre-trained model that is supposed to extract rich contextualized features and two separate modules for extracting expressions and predicting edges. For english datasets (Opener$_{\text{EN}}$, Darmstadt$_{\text{Unis}}$ and MPQA) we used RoBERTa$_{\text{base}}$ (Liu et al., 2019) model and for non-english ones (Opener$_{\text{ES}}$, Norec$_{\text{Fine}}$,

		#	avg.	max
NoReC$_{\text{Fine}}$	train	8634	21.3	125
	dev	1531	21.4	86
MultiB$_{\text{CA}}$	train	1174	18.7	238
	dev	167	16.3	129
MultiB$_{\text{EU}}$	train	1063	14.5	105
	dev	152	14.6	60
MPQA	train	5873	25.6	134
	dev	2063	25.5	228
Darmstadt$_{\text{Unis}}$	train	2253	21.6	226
	dev	232	19.5	80
Opener$_{\text{EN}}$	train	1744	15.6	131
	dev	249	15.2	129
Opener$_{\text{ES}}$	train	1438	19.3	175
	dev	206	18.9	108

Table 1: General information of shared task datasets. First five datasets are used in (Barnes et al., 2021). Max and average are measures of tokens' counts.

Multibooked$_{\text{CA}}$ and Multibooked$_{\text{EU}}$), we used LaBSE (Feng et al., 2020) as model's base.

We treat quadruples in a sentence as a graph which its nodes are expressions and its edges connect expressions within a quadruple. We name the module for extracting expression, **Node Extractor** and the module for determining edges, **Edge Predictor**. System structure is depicted in figure 1. Firstly, input sentence is fed into model base. Then node extractor gets the encoded input and for each one of three entity types, predicts expressions in the sentence. Polarities are also predicted by this module. Edge predictor, then, gets the expressions and for every pair of sentiment-target and sentiment-holder expressions, predicts if there is an edge between them. Finally a sentiment expression plus its polarity along with all of target and holder expressions connected thereto would become one of predicted quadroples.

4.1 Node Extractor

This module consists of three feed forward neural networks for predicting label of each token in BIO scheme, each network for an entity type. Using these three networks, sequence tokens are labeled for sentiment, target and holder expressions independently. In this networks, ReLU is used as activation and each network has less than 50k parameters. Loss function of the module is cross entropy. We integrate predicting polarity of sentiment expressions in sequence labelling task by replicating BIO labels for negative, neutral and positive polarities.

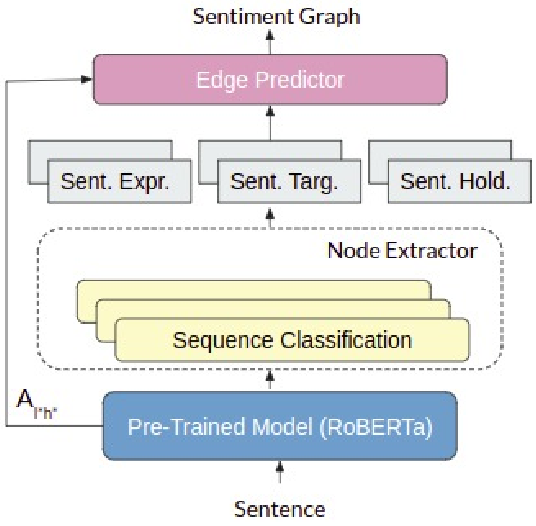

Figure 1: System architecture

Note that node extractor and jointly predicting sentiment expression and polarity is inspired of what is being used in one of our baselines, the one using sequence labelling approach.

4.2 Edge Predictor

Aimed for leveraging pretrained model's knowledge as much as possible, we simply used its computed attentions to predict edges. To this end, we examined different settings that are as follows:

Base: This is the base setting of edge predictor. We use attention values of a predetermined head in a specific layer in this way that we compute sum of attentions of two given expressions to each other and apply sigmoid to predict being an edge between them. More specifically, we choose head 7 in layer 8 of RoBERTa$_{\text{base}}$ for english datasets and head 9 layer 11 of LaBSE for others. This choice was based on the observation that those heads were more semantic-aware in the sense that sentiment expression tokens attend more to target expression tokens. Those heads are also less position dependant, meaning that a token's attention distribution does not lean toward a specific position in the sequence.

Consider two nodes a and b which span intervals (a_{begin}, a_{end}) and (b_{begin}, b_{end}) in the sentence respectively. Probability of being an edge being between a and b, $P_{e_{ab}}$ in the is computed as follows:

$$P_{e_{ab}} = \sigma(\sum_{i=a_{begin}}^{a_{end}} \sum_{j=b_{begin}}^{b_{end}} A_{l^*h^*ij} + A_{l^*h^*ji}) \quad (5)$$

		holders			targets			sent. expr.			polarity		
		#	avg.	max	#	avg.	max	#	avg.	max	+	neu	-
NoReC$_{\text{Fine}}$	train	898	1.3	18	6778	3.0	44	8448	6.8	51	5695	0	2753
	dev	120	1.1	4	1152	3.0	22	1432	7	38	988	0	444
MultiB$_{\text{CA}}$	train	169	1.3	4	1705	3.0	20	1989	3.2	21	1273	0	716
	dev	15	1.8	8	211	2.9	11	258	3.2	12	151	0	107
MultiB$_{\text{EU}}$	train	204	1.4	8	1277	2.3	14	1679	3.2	14	1401	0	278
	dev	33	1.5	5	152	2.3	9	203	3.6	13	167	0	36
MPQA	train	1425	3.2	43	1479	7.0	52	1706	2.2	16	671	337	698
	dev	405	3.3	28	494	6.1	44	570	2.2	9	231	124	215
Darmstadt$_{\text{Unis}}$	train	63	1.2	4	806	1.5	6	806	2.2	25	340	102	364
	dev	9	1.3	2	98	1.7	4	98	2.4	13	29	15	54
Opener$_{\text{EN}}$	train	266	1.0	3	2679	1.9	11	2884	2.6	18	2101	0	783
	dev	49	1.0	2	371	2.0	10	400	2.6	14	284	0	116
Opener$_{\text{ES}}$	train	176	1.0	3	2748	2.4	13	3042	2.6	16	2472	0	570
	dev	23	1.0	2	363	2.6	11	387	2.6	17	317	0	70

Table 2: Details of shared task datasets. First five datasets are used in (Barnes et al., 2021). Max and average are measures of tokens' counts in expressions.

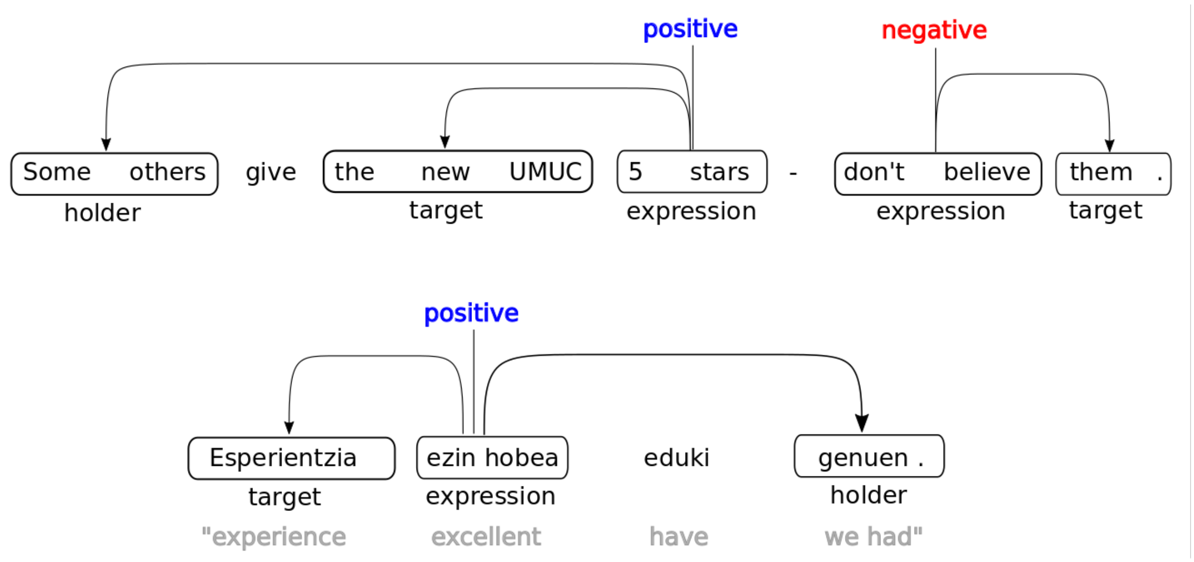

Figure 2: A sentiment graph example. Entities of a quadrople are connected with edges and sentiment expressions are labeled with polarity. Image is from (Barnes et al., 2022)

A is the pretrained model's computed attention. l^* and h^* determine prespecified head and layer respectively. σ is the sigmoid function. This equation simply means attention of two nodes to each other determines probability of being an edge between them.

+AvgH: In this setting we do not rely only on specific heads and use attention values of all heads in all layers. To this end we introduce new learnable weights, one for each head, and compute linear combination of attention values across all heads. Total number of new weights is number of layers times number of heads in each layer, which is 144 in both RoBERTa$_{\text{base}}$ and LaBSE. So in this setting, $P_{e_{ab}}$ becomes

$$\sigma(\sum_{l,h=1}^{n_l,h_l} \sum_{i=a_{begin}}^{a_{end}} \sum_{j=b_{begin}}^{b_{end}} w_{lh} * (A_{lhij} + A_{lhji})) \quad (6)$$

in which n_l and n_h are number of layers and heads in the pretrained Model respectively. w is the learned weight assigned to each head. At training start w is set to 1 in setting one head and 0 in other heads.

+SepH: We use separate heads for predicting edges for sentiment-target expression pair and sentiment-holder expression pair. Specifically, for predicting and edge between a sentiment and a target expression, we use head 7 in layer 8 for english datasets and head 9 in layer 11 for non-english

	$\text{Opener}_{\text{EN}}$	$\text{Opener}_{\text{ES}}$	$\text{Norec}_{\text{Fine}}$	$\text{Darmstadt}_{\text{Unis}}$	$\text{MultiB}_{\text{EU}}$	$\text{MultiB}_{\text{CA}}$	MPQA	Average
base	0.64	0.65	0.44	0.45	0.67	0.64	0.38	0.55
+AvgH	-0.03	-0.02	-0.05	-0.14	0.02	0.01	-0.31	-0.07
+SepH	0.03	0.02	-0.05	-0.03	-0.00	-0.02	-0.08	-0.02
+SepH+AvgH	0.01	-0.00	-0.03	-0.07	-0.02	-0.00	-0.24	-0.05

Table 3: Performance of different variants of our model. First row is the default variant and numbers in other rows show increase or decrease in comparison with the default variant. Results are based on evaluation on development data.

ones. For predicting an edge between a sentiment expression and a holder expression, we use head 10 in layer 11 for english datasets and head 4 in layer 11 for non-english ones.

+SepH+AvgH: This is the mixture of last two settings. We use all heads in all layers but we consider separate linear combinations of them for sentiment-target and sentiment-holder expression pairs. So the number of introduced weights is doubled to 288 in this setting.

5 Experiments

5.1 Baselines

We compare our model with two baselines proposed in the shared task. First one is basically (Barnes et al., 2021) which is described in section 2. The second one takes sequence labelling approach. It sets two modules for predicting expressions and relations respectively. Expression prediction module consists of three BiLSTMs which predict sentiment, target and holder expressions respectively. This module is fed by an embedding layer which is initialized with pretrained word embeddings. Relation prediction module, given two extracted expressions and the input sentence, uses separate BiLSTMs and max pooling to make contextualized representations from them. Then it feeds concatenation of three max pooling layers to a linear layer and sigmoid to predict if there is relation between two expressions or not. In the comparisons we call the baselines, **GP** (for graph parsing) and **SL-BiLSTM** (for sequence labelling) respectively.

5.2 Settings

We trained each variant of our model 5 times, each with different random seed on every dataset for at most 20 epochs. learning rate was 1e-4 and 1e-3 for pretrained model and newly introduced weights respectively. We did warm-up in epoch 1 and applied step LR scheduling with gamma as .1 and step size as 9. Baselines were also trained using script given by shared task organizers. Evaluation measure being used was Sentiment Graph F1. Evaluation was done using the script given by shared task organizers. We used PyTorch (Paszke et al., 2019) plus Pytorch-Ignite (Fomin et al., 2020) as training framework and WandB (Biewald, 2020) as Experiment Tracking tool.

Dataset	Attentionist	GP	SL-BiLSTM
$\text{NoReC}_{\text{Fine}}$	**0.32**	0.28	0.19
$\text{MultiB}_{\text{CA}}$	**0.61**	0.54	0.33
$\text{MultiB}_{\text{EU}}$	**0.59**	0.57	0.34
MPQA	**0.26**	0.15	0.02
$\text{Darmstadt}_{\text{Unis}}$	**0.25**	0.21	0.13
$\text{Opener}_{\text{EN}}$	**0.58**	0.52	0.31
$\text{Opener}_{\text{ES}}$	**0.55**	0.50	0.26

Table 4: Test results of our model and baselines. Evaluation metric is Sentiment Graph F1

6 Results

Results are shown in table4. **Attentionist** is our proposed model. For our model, only result of the best variant is shown. As it can be seen, our model outperforms the baseline in all datasets.

7 Ablation Study

In order to assess and compare performance of different variants of our model, we trained each variant four times using different random seeds on each dataset and compared them on development data. Results are depicted in table3.

References

Rodrigo Agerri, Montse Cuadros, Sean Gaines, and German Rigau. 2013. OpeNER: Open polarity enhanced named entity recognition. In *Sociedad Española para el Procesamiento del Lenguaje Natural*, volume 51, pages 215–218.

A.H. Alamoodi, B.B. Zaidan, A.A. Zaidan, O.S. Albahri, K.I. Mohammed, R.Q. Malik, E.M. Almahdi, M.A. Chyad, Z. Tareq, A.S. Albahri, Hamsa Hameed, and Musaab Alaa. 2021. Sentiment analysis and its applications in fighting covid-19 and infectious diseases: A systematic review. *Expert Systems with Applications*, 167:114155.

Jeremy Barnes, Toni Badia, and Patrik Lambert. 2018. MultiBooked: A corpus of Basque and Catalan hotel reviews annotated for aspect-level sentiment classification. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Jeremy Barnes, Robin Kurtz, Stephan Oepen, Lilja Øvrelid, and Erik Velldal. 2021. Structured sentiment analysis as dependency graph parsing. *CoRR*, abs/2105.14504.

Jeremy Barnes, Oberländer Laura Ana Maria Kutuzov, Andrey and, Enrica Troiano, Jan Buchmann, Rodrigo Agerri, Lilja Øvrelid, Erik Velldal, and Stephan Oepen. 2022. SemEval-2022 task 10: Structured sentiment analysis. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, Seattle. Association for Computational Linguistics.

Lukas Biewald. 2020. Experiment tracking with weights and biases. Software available from wandb.com.

Zhuang Chen and Tieyun Qian. 2020. Relation-aware collaborative learning for unified aspect-based sentiment analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3685–3694, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2020. Language-agnostic BERT sentence embedding. *CoRR*, abs/2007.01852.

V. Fomin, J. Anmol, S. Desroziers, J. Kriss, and A. Tejani. 2020. High-level library to help with training neural networks in pytorch. `https://github.com/pytorch/ignite`.

Ruidan He, Wee Sun Lee, Hwee Tou Ng, and Daniel Dahlmeier. 2019. An interactive multi-task learning network for end-to-end aspect-based sentiment analysis. *CoRR*, abs/1906.06906.

John Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data.

Xin Li, Lidong Bing, Wenxuan Zhang, and Wai Lam. 2019. Exploiting BERT for end-to-end aspect-based sentiment analysis. *CoRR*, abs/1910.00883.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Lilja Øvrelid, Petter Mæhlum, Jeremy Barnes, and Erik Velldal. 2020. A fine-grained sentiment dataset for Norwegian. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 5025–5033, Marseille, France. European Language Resources Association.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. dAlché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. SemEval-2015 task 12: Aspect based sentiment analysis. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 486–495, Denver, Colorado. Association for Computational Linguistics.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. SemEval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 27–35, Dublin, Ireland. Association for Computational Linguistics.

Margarita Rodríguez-Ibáñez, Francisco-Javier Gimeno-Blanes, Pedro Manuel Cuenca-Jiménez, Cristina Soguero-Ruiz, and José Luis Rojo-Álvarez. 2021. Sentiment analysis of political tweets from the 2019 spanish elections. *IEEE Access*, 9:101847–101862.

Cigdem Toprak, Niklas Jakob, and Iryna Gurevych. 2010. Sentence and expression level annotation of opinions in user-generated discourse. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 575–584, Uppsala, Sweden. Association for Computational Linguistics.

Wenya Wang, Sinno Jialin Pan, Daniel Dahlmeier, and Xiaokui Xiao. 2016. Recursive neural conditional random fields for aspect-based sentiment analysis. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages

616–626, Austin, Texas. Association for Computational Linguistics.

Wenya Wang, Sinno Jialin Pan, Daniel Dahlmeier, and Xiaokui Xiao. 2017. Coupled multi-layer attentions for co-extraction of aspect and opinion terms. *Proceedings of the AAAI Conference on Artificial Intelligence*, 31(1).

Janyce Wiebe, Theresa Wilson, and Claire Cardie. 2005. Annotating expressions of opinions and emotions in language. *Language Resources and Evaluation*, 39(2-3):165–210.