information

Article

Combating Fake News with Transformers: A Comparative Analysis of Stance Detection and Subjectivity Analysis

Panagiotis Kasnesis *, Lazaros Toumanidis and Charalampos Z. Patrikakis

Department of Electrical and Electronic Engineering, University of West Attica, 12244 Athens, Greece; laztoum@uniwa.gr (L.T.); bpatr@uniwa.gr (C.Z.P.)
* Correspondence: pkasnesis@uniwa.gr; Tel.: +30-210-5381549

Abstract: The widespread use of social networks has brought to the foreground a very important issue, the veracity of the information circulating within them. Many natural language processing methods have been proposed in the past to assess a post's content with respect to its reliability; however, end-to-end approaches are not comparable in ability to human beings. To overcome this, in this paper, we propose the use of a more modular approach that produces indicators about a post's subjectivity and the stance provided by the replies it has received to date, letting the user decide whether (s)he trusts or does not trust the provided information. To this end, we fine-tuned state-of-the-art transformer-based language models and compared their performance with previous related work on stance detection and subjectivity analysis. Finally, we discuss the obtained results.

Keywords: deep learning; stance detection; subjectivity analysis; transformers; natural language processing, social media; misinformation

Citation: Kasnesis, P.; Toumanidis, L.; Patrikakis, C.Z. Combating Fake News with Transformers: A Comparative Analysis of Stance Detection and Subjectivity Analysis. *Information* **2021**, *12*, 409. https://doi.org/10.3390/info12100409

Academic Editors: Vincenza Carchiolo and Kostas Vergidis

Received: 18 August 2021
Accepted: 1 October 2021
Published: 3 October 2021

1. Introduction

Assessing a social media post's veracity is a quite impossible task to handle directly through the use of natural language processing [1], since the provided estimations cannot be considered reliable. However, there are many ways in which machine-generated outputs could provide useful insights to human beings and assist them in making decisions of their own about a post's veracity. For example, users could be provided with information about social media content and context in an intermediary-free approach and in a way that assists users in deriving their own conclusions regarding a social media post's trustworthiness [2]. This information could act as an indicator of trustworthiness, derived by machine learning algorithms trained on tasks such as subjectivity analysis and stance detection.

In particular, the spread of online misinformation has been linked to the presence of subjective knowledge, especially when it comes to scientific topics, due to the fact that it captures a person's perceived own ability to understand research [3]. Moreover, recent research work on the analysis of the subjectivity level in fake news fragments reinforces the concept that misinformation is correlated with the use of subjective language [4,5]. In general, subjectivity analysis is a classification task, which aims at categorising posts as factual or opinionated and can be used as an indicator, providing social media users with intuition about the trustworthiness of a selected post.

Another well-investigated indicator of a post's trustworthiness/veracity is that of stance detection, a multi-class classification task that captures the users' reactions in social media posts [1]. Given a source post (rumourous post) that includes a statement a and a set of reply posts B (it can be only one reply), stance detection aims at classifying the stance of the user who wrote post b towards post a. In particular, the stance detection task categorises the replies to that using the following four categories (Figure 1):

- *Support*: the author of the response supports the veracity of the rumour to which they are responding (e.g., "I've heard that also").

- *Deny*: the author of the response denies the veracity of the rumour to which they are responding (e.g., "That's a lie").
- *Query*: the author of the response asks for additional evidence in relation to the veracity of the rumour to which they are responding (e.g., "Really?").
- *Comment*: the author of the response makes their own comment without a clear contribution to assessing the veracity of the rumour to which they are responding (e.g., "True tragedy").

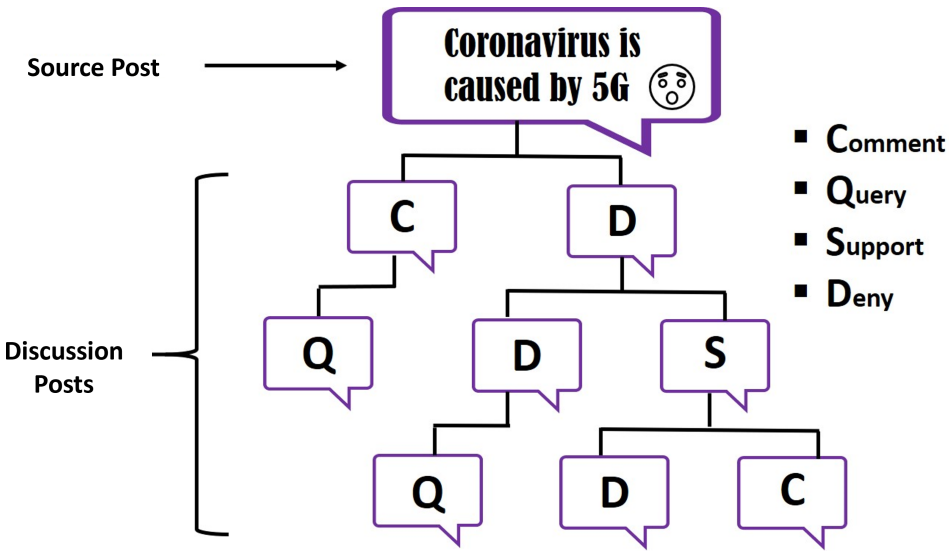

Figure 1. Example of tree-based structure used for SDQC stance detection.

In Figure 1, an example thread of posts is presented for the case of the SDQC (Support–Deny–Query–Comment) stance detection task. It should be noted that an alternative stance detection task is presented in the SemEval 2016 task 6 [6], which is another classification task that labels tweets into the classes (a) favour, (b) against, and (c) neither. The main reason for selecting the SDQC stance detection task is that it provides the classes Query and Comment, which provide an indication of the conversational context surrounding rumours. This may lead to the discovery of certain patterns of comments and questions that can be indicative of false rumours and others indicative of rumours that turn out to be true [1].

The main objective of this paper is to take advantage of the recent advances in deep-learning-based natural language processing originated by the transformer architecture [7] and implement a comparative analysis of their efficacy for subjectivity analysis and stance detection. In the rest of the paper, we provide an overview of the following:

- Related work on subjectivity analysis and stance detection;
- An overview of the deep-learning models' architectures we fine-tuned and evaluated;
- A description of the selected datasets that were used for training and evaluation;
- A presentation and discussion of the obtained results.

2. Related Work

One of the first studies on subjectivity analysis and text classification in general worth mentioning is that of Wang and Manning [8]. They examined the benefit of dropout training [9] without actually randomly sampling, thereby using all the data efficiently. Their approach uses a Gaussian approximation that is justified by the central limit theorem and empirical evidence, while it reduces the training time, also providing more stability. They applied their approach to the subjectivity dataset v1.0 (SUBJ) [10] using logistic regression for simplicity, achieving 93.60% accuracy.

A self-adaptive hierarchical sentence model, called AdaSent, is presented in [11]. AdaSent exploits recursive convolutional neural networks (gr-Conv) [12] while forming

a multi-scale hierarchical representation instead of a fixed-length continuous vectorial representation. The gated nature of AdaSent allows the information flow to vary with each classification task, so there is no need for a predefined parse tree. Moreover, it uses the intermediate representations at each level of the pyramid (i.e., not only the top one) to form a multiscale summarization. This is achieved by applying a convex combination to each level representation and adaptively giving more weight to some levels depending on the sentence and the task. AdaSent achieved 95.50% accuracy on the SUBJ dataset.

Amplayo et al. [13] propose the use of a neural attention-based multiple context fixing attachment (MCFA) that is used mainly for neural machine translation but is applicable to classification tasks as well. MCFA's main objective is to mitigate the possible problems when using translated sentences as context; it uses all the sentence vectors of a translated sentence a as context to fix a translated sentence vector b, and vice versa, which is accomplished by selectively moving the vectors to a location in the same vector space. Using a convolutional neural network (CNN) attached to MCFA significantly improved the classification performance of the CNN, achieving 94.80% accuracy on the SUBJ dataset.

Byte-level recurrent language models are explored in Radford et al. [14], which have the advantage of learning representations in an unsupervised manner, including disentangled features corresponding to high-level concepts when they are given huge amounts of training data. The learned representations are able to achieve state-of-the-art results given only a handful of labelled examples. The authors used a single-layer multiplicative LSTM [15] model with 4096 units, instead of a regular one, because it converges faster, and achieved 94.60% accuracy on the SUBJ dataset. Cer et al. [16] proposed the Universal Sentence Encoder (USE) mechanism for encoding sentences into embedding vectors and specifically targeted transfer learning for other NLP tasks, such as subjectivity analysis. It is a transformer-based network [7] combined with a deep averaging network [17], where input embeddings for words and bi-grams are averaged together and, afterwards, fed to a feedforward deep neural network to produce sentence embeddings. The USE augments unsupervised learning with training on the Stanford Natural Language Inference (SNLI) dataset [18], which improves the transfer learning performance. Unsupervised training data for the sentence encoding models are drawn from a variety of web sources (e.g., Wikipedia, web news, etc.). The performance of the USE on the SUBJ datasets is equal to 93.90%.

When it comes to stance detection, the model named EventAI [19] used an ensemble approach for stance detection, which includes neural network models combined with the traditional classification algorithms. This work combined message embeddings, which were produced by word2vec embeddings [20] passed through an attentional LSTM, and human-crafted features. The latter contain features such as whether there is a link in the post or not, the type of the link (e.g., video), whether the current post is a reply to the source message, the similarity with the source message, the content length, etc. The message embeddings and other features were concatenated together and fed into the neural network with two fully connected layers and a softmax layer for the final label output. Moreover, they used a rule-based model to handle some special cases, such as if the source message had a question mark and then should be a query. The EventAI team came third in the SemEval 2019 Subtask 7A [21], with a macro F1-score equal to 0.5776.

The authors in [22] proposed an LSTM-based sequential model (called BranchLSTM) that models the whole conversational structure of tweets. The input at each time step i of the LSTM layer is the vector representation of the tweet. They recorded the output of each time step so as to attach a label to each tweet in a branch. As features, they used word2vec embeddings, a lexicon (i.e., counting negation and swear words), the content length, the presence of punctuation, the presence of attachments (e.g., URLs), and the sentence similarity with previous posts. This model was the winner in the SemEval 2017 Subtask 8A [1] and was considered as the baseline in that of 2019, achieving a macro F1-score equal to 0.4929, coming fourth.

A fine-tuned BERT model was used in [23], without using any human-extracted features. Similarly to our approach (see Section 3), the authors formulated the problem as a stance classification, determining the rumour stance of a post with respect to the previous thread post and the source thread post. They trained 100 models and constructed a BERT ensemble, called TOP-Ns, where several BERT models were fused in order to increase the F1 measure, and their pre-softmax scores were averaged to produce the output class probabilities. The fine-tuned BERT model achieved an F1-score of 0.6167 on the provided test data. Even though this approach is effective, it should be noted that using 100 BERT models significantly increases the inference time.

Finally, Yang et al. [24] proposed an inference chain-based system, which fully utilizes conversation structure-based knowledge. The model they used for the stance detection task was a fine-tuned Generative Pretrained Transformer (GPT) [25]. Moreover, they expanded the training data in minority categories (i.e., support, deny and query) to alleviate class imbalance. For support and deny, they exploited fact verification datasets, such as Emergent [26]. For query, the authors used passages as the conversation context, and unanswerable questions as the target tweet in reading comprehension datasets, such as SQuAD 2.0 [27]. This approach achieved the highest performance in the 2019 contest, reaching a F1-score equal to 0.6187.

3. Methods

In this section, the architectures of the transformer-based architectures we evaluated are described. The reason for selecting transformer-based models is that they can be fine-tuned and obtain state-of-the-art results [28–30], by simply applying transfer learning (i.e., the model is already pretrained).

In general, transformers follow an encoder–decoder architecture. The encoder is composed of a stack of N = 6 identical layers, where each layer has two sublayers: (a) a multi-head self-attention mechanism, and (b) a position-wise fully connected feed-forward network. A residual connection is employed around each of the two sublayers, followed by a normalisation layer. On the other hand, the decoder is also composed of a stack of N = 6 identical layers. Similar to the encoder, residual connections are employed around each of the sublayers, followed by layer normalization. In addition to the two sublayers in each encoder layer, the decoder inserts a third sublayer, performing multi-head attention over the output of the encoder stack.

The attention mechanism that the transformer model employs is presented in the following equation; it is a function that maps a query and a set of key-value pairs to an output by computing their dot product (the query Q, keys K, values V and output are all vectors). The output is computed as a weighted sum of the values, where the weight assigned to each value is computed by a softmax function of the query with the corresponding key.

$$A(Q, K, V) = softmax\left(\frac{Q \cdot K^T}{\sqrt{d}}\right) \cdot V \quad (1)$$

where d is the dimensionality of the key vectors used as a scaling factor.

An effective alternative to the dot-product attention is multi-head attention that allows the model to attend to information from different representation subspaces at different positions.The following equations represent the multi-head attention mechanism having, as input, the text x.

$$Q_i = x \cdot W_i^Q, K_i = x \cdot W_i^K, V_i = x \cdot W_i^V \quad (2)$$

$$head_i = A_i(Q, K_i, V_i) \quad (3)$$

$$E(Q, K, V) = concat(head_1, ..., head_h)W^A \quad (4)$$

where $W_i^Q \in R^{d_{model} x d_q}$, $W_i^K \in R^{d_{model} x d_k}$, $WV_i \in R^{d_{model} x d_v}$, $W_i^A \in R^{d_{model} x d_a}$ and $d_q = d_k = d_v = d_a = d_{model}/h$. After computing A_i, we concatenate the h dot products and transform them into E using, again, a dense layer.

3.1. BERT

Bidirectional Encoder Representations from Transformers (BERT) is designed to pretrain deep bidirectional representations from unlabelled text by jointly conditioning on both the left and right context in all the layers [28].

BERT relies only on the encoder part of the Transformer and uses two self-supervised pretraining objectives for training, "masked language model" (MLM) and "next sentence prediction" (NSP). In MLM, BERT randomly masks some of the tokens from the input, and the objective is to predict the original vocabulary ID of the masked word (token) based only on its context. In the case of NSP, BERT jointly pretrains text-pair representations and classifies whether sentence a and sentence b are consecutive. For the pretraining corpus, the authors used BooksCorpus (800 M words) [31] and English Wikipedia (2500 M words). Fine-tuning BERT is quite simple since the self-attention mechanism in the Transformer allows BERT to model many downstream tasks, such as classification tasks, Natural Language Inference (NLI), Question Answering (QA) and Named Entity Recognition (NER).

BERT obtained state-of-the-art results in 2019 in eleven natural language processing tasks, including increasing the GLUE [32] score to 80.5%. Devlin et al. evaluated two versions of BERT, BERT_{BASE} and BERT_{LARGE}. The BERT_{BASE} model has 12 attentional blocks, 12 self-attentional heads, and its sublayers and the embedding layers; produces outputs of dimension equal to 768; and contains 110M parameters. On the other hand, BERT_{LARGE} has 24 attentional blocks, 16 self-attentional heads, and its sublayers and the embedding layers; produces outputs of dimension equal to 1024; and contains 340M parameters.

3.2. RoBERTa

Robustly optimised BERT pretraining Approach (RoBERTa) is a replication study of BERT pretraining that focuses on evaluating the effects of hyperparameter tuning and training set size [29]. In particular, the authors applied the following modifications to BERT:

1. The model was trained for longer, with bigger batches, over more data;
2. They removed the NSP objective;
3. It was trained on longer sequences;
4. The masking pattern applied to the training data was dynamically changed.

The main reason for not including the NSP objective is that it is designed to improve performance on downstream tasks, such as NLI, which require reasoning about the relationships between pairs of sentences, and does not generally benefit other NLP tasks. RoBERTa surpassed BERT's performance on almost eleven natural language processing tasks, including increasing the GLUE score to 88.5%. The authors also evaluated two versions of RoBERTa, RoBERTa_{BASE} and RoBERTa_{LARGE}, with both sharing the same parameter size with the corresponding BERT architectures. Thus, the RoBERTa_{BASE} model has 12 attentional blocks, 12 self-attentional heads, and its sublayers and the embedding layers; produces outputs of dimension equal to 768; and contains 110M parameters. On the other hand, RoBERTa_{LARGE} has 24 attentional blocks, 16 self-attentional heads, and its sublayers and the embedding layers; produces outputs of dimension equal to 1024; and contains 340M parameters.

3.3. ELECTRA

ELECTRA (Efficiently Learning an Encoder that Classifies Token Replacements Accurately) proposes the use of a pretraining task called replaced token detection (RTD) that is more sample-efficient than MLM [30]. MLM pretraining methods produce models that generalize well in several NLP tasks; however, they generally require large amounts of computation to be effective. RTD corrupts the input sentence by replacing some tokens with plausible alternatives sampled from a small generator network. Thus, they train a discriminative model that predicts whether each token in the corrupted input was replaced by a generator sample or not.

A key advantage of this discriminative task is that the model learns from all the input tokens instead of just the small masked-out subset, making it more computationally efficient. Moreover, although the proposed approach is similar to training the discriminator of a generative adversarial network (GAN), it is not adversarial, since the generator producing corrupted tokens is trained with maximum likelihood due to the difficulty of applying GANs to text [33]. It should be noted that, in the ELECTRA pretraining architecture, the authors used a relatively small Transformer for the generator and a large one for the discriminator, after experimentation. ELECTRA surpassed both BERT's and RoBERTa's performance in many tasks and achieved a GLUE score equal to 89.4%. The authors also evaluated two versions of ELECTRA, ELECTRA$_{BASE}$ and ELECTRA$_{LARGE}$, with both sharing the same parameter size with the corresponding BERT and RoBERTa architectures.

4. Results

4.1. Experimental Set-Up

The experiments were executed on a computer workstation equipped with an NVIDIA GTX 1080 Ti GPU featuring 11 gigabytes of RAM, 3584 CUDA cores and a bandwidth of 484 GB/s. We used Python as the programming language and, specifically, the Numpy (https://numpy.org/, accessed on 2 August 2021) library for matrix multiplication, Re (https://docs.python.org/3/library/re.html#module-re, accessed on 2 August 2021) library for text preprocessing (i.e., regular expression operations) and PyTorch (https://pytorch.org/, accessed on 2 August 2021), transformers (https://github.com/huggingface/transformers, accessed on 2 August 2021) and simple transformers (https://github.com/ThilinaRajapakse/simpletransformers, accessed on 2 August 2021) libraries for retraining and evaluating the deep-learning models (BERT, RoBERTa and ELECTRA). Moreover, the wandb (https://wandb.ai/home, accessed on 11 August 2021) platform was used for visualizing the results. In order to accelerate the tensor multiplications, we used the CUDA Toolkit with support from the cuDNN (https://developer.nvidia.com/cudnn, accessed on 15 August 2021), which is the NVIDIA GPU-accelerated library for deep neural networks. The workstation has the Ubuntu 16.04 Linux operating system.

4.2. Datasets

4.2.1. SUBJ

We used the Cornell movie review dataset [10], which is a publicly available dataset, to retrain and evaluate the performance of the selected networks; it consists of 5000 subjective (class label equal to 1) and 5000 objective (class label equal to 2) processed sentences. The starting points for data acquisition were snippets of movie reviews from Rotten Tomatoes (http://www.rottentomatoes.com/, accessed on 15 August 2021) and plot summaries for movies from the Internet Movie Database (http://www.imdb.com, accessed on 15 August 2021). Since the class distribution in this dataset is balanced, we only used accuracy as an evaluation metric. We split the dataset randomly into 8000 samples for training and 2000 for evaluation. An example of the text samples included in the SUBJ dataset is depicted in Table 1.

Table 1. An example of the text samples included in the SUBJ dataset.

Text	Label
Celebrities are talking about him on MTV and girls are fighting over him on Jerry springer.	Objective
Funny in a sick , twisted sort of way.	Subjective
If Oscar had a category called best bad film you thought was going to be really awful but wasn't, guys would probably be duking it out with the queen of the damned for the honor.	Subjective
Colt seeks the repair of a femininity damaged by an earlier incest.	Objective

4.2.2. SemEval 2019 Subtask 7A

We used the public available dataset SemEval 2019 Subtask 7A. This dataset contains Twitter and Reddit threads, and each one is part of a tree-structured thread, which is categorised into one of the aforementioned four categories: Support, Deny, Query, and Comment. Table 2 displays the label distribution of the used stance detection dataset. It is observable that the dataset is too imbalanced since around 70% of the training labels are equal to the Comment class, and the same goes for the test set labels.

Table 2. Stance detection dataset label distribution.

Set	Support	Deny	Query	Comment	Total
Twitter Train	1004	415	464	3685	5568
Reddit Train	23	45	51	1015	1134
Total Train	1027	460	515	4700	6702
Twitter Test	141	92	62	771	1066
Reddit Test	16	54	31	705	806
Total Test	157	146	93	1476	1872

An example of the tree-based nature of the threads included in the dataset is depicted in Figure 2.

user1: We understand that there are two gunmen and up to a dozen hostages inside the cafe under siege at Sydney.. ISIS flags remain on display #7News **[support]**
- ***user2:*** @user1 not ISIS flags **[deny]**
- ***user3:*** @user1 sorry - how do you know its an ISIS flag? Can you actually confirm that? **[query]**
  - **user4:** @user3 no she cant cos its actually not **[deny]**
- ***user5:*** @user1 More on situation at Martin Place in Sydney, AU LINK **[comment]**
- ***user6:*** @user1 Have you actually confirmed its an ISIS flag or are you talking shit **[query]**

Figure 2. Post examples included in the SemEval 2019 Subtask 7A dataset.

4.3. Text Preprocessing

Since the posts contained many links to websites and media, we used the token "<url>" to represent all of them, while we used the "@user" token to denote all the user names. Moreover, emojis were converted into text using the emoji Python library, e.g., 🙂 was translated into ":slightly_smiling_face:". Finally, we lowercased all the capital characters (e.g., "New York" to "new york").

4.4. Hyperparameter Tuning

For the case of subjectivity analysis, we used almost 1000 text samples of the training set as a validation set, while for that for stance detection, we used almost one fourth (around 1400 post pairs) of the training set as a validation set, which were already marked as *dev* in the dataset.

These splits were performed for:

- Freezing the parameters of the model that achieved the highest F1-score in the validation set;
- Hyperparameter optimization (selecting the best hyperparameters).

Table 3 demonstrates the selected hyperparameters for the stance detection task. Adam [34] was selected as a model optimizer. Moreover, it is worth mentioning that the batch size was constrained by the GPU RAM memory.

Table 3. Selected hyperparameters for subjectivity analysis and the stance detection tasks.

Hyperparameter	Subjectivity Analysis	Stance Detection
Learning rate α	1×10^{-5}	2×10^{-5} (1×10^{-5} for ELECTRA)
Adam ϵ	1×10^{-8}	1×10^{-8}
Adam $\beta 1$	0.9	0.9
Adam $\beta 2$	0.999	0.999
Dropout	0.1	0.1
Batch size	16	8
Max. sequence length	128	128

4.5. Performance

4.5.1. Subjectivity Analysis

Table 4 presents the results obtained using the selected and fine-tuned transformer-based models compared to other state-of-the-art subjectivity analysis models. As expected, the performance of the transformer-based models exceeded the accuracy of previous work, just as they did in the other tasks included in the GLUE benchmark. Moreover, unsurprisingly, the large versions of these models achieved better results than the base ones. ELECTRA$_{LARGE}$ achieved the highest accuracy score, which was equal to 98.30%.

Table 4. Comparison of fine-tuned transformer-based models with other state-of-the-art subjectivity analysis models.

Model	Accuracy, %
AdaSent [11]	95.50
CNN+MCFA [13]	94.80
Byte mLSTM [14]	94.60
USE [16]	93.90
Fast Dropout [8]	93.60
BERT$_{BASE}$	96.40
BERT$_{LARGE}$	97.20
RoBERTa$_{BASE}$	97.10
RoBERTa$_{LARGE}$	97.75
ELECTRA$_{BASE}$	97.05
ELECTRA$_{LARGE}$	98.30

These results led to an almost perfect confusion matrix, displayed in Figure 3, where it is worth noting that the $ELECRA_{LARGE}$ model tends to misclassify more objective pieces of text as subjective (24) than the other way around (10). The variation of true positives (subjective) and true negatives (objective) during training is displayed in Figure 4. The best equilibrium was accomplished around the 80th step (learning rate update), meaning around the eighth training epoch.

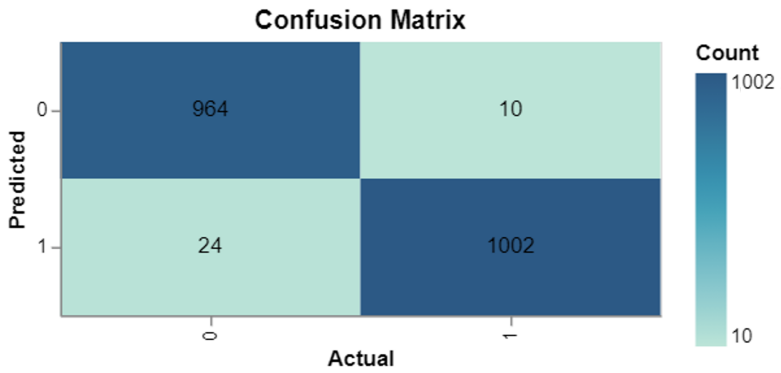

Figure 3. Confusion matrix of the ELECTRA$_{LARGE}$ model on the test SUBJ dataset.

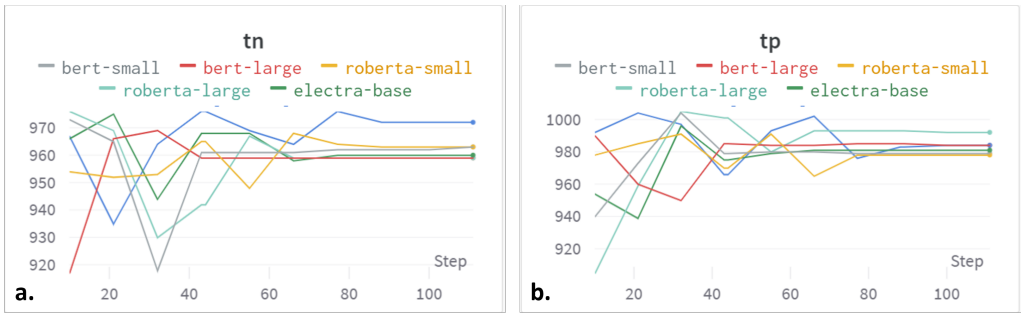

Figure 4. Variation of the results retrieved during training steps: (**a**) true negatives (objective) and (**b**) true positives (subjective).

4.5.2. Stance Detection

The results we obtained for stance detection are included in Table 5. Apart from selected fine-tuned transformer-based models, we include the most efficient deep-learning models that have been submitted for the SemEval 2019 Subtask 7A challenge. In this task, the best results (in terms of the Macro-averaged F1-score) were achieved by the RoBERTa$_{LARGE}$ model. This is probably due to the fact the dataset has a lot of slang words (e.g., LOL) and phrases, and RoBERTa contains a larger corpus (around 50,000 tokens) and has been pretrained on Twitter/Reddit posts [29]. RoBERTa$_{LARGE}$ achieved a 0.6301 F1-score, which is state of the art but is not that high. On the other hand, upon having a look at the confusion matrix (Figure 5), it is observable that the model misclassified only two Support comments as Deny and none vice versa; thus, there were not many "extreme" misclassifications; most of the misclassifications were due to the fact that the dataset is highly imbalanced, and a lot of the posts are classified as Comment.

Table 5. Comparison of fine-tuned transformer-based models with other state-of-the-art stance detection models.

Model	Macro F1-Score %
EventAI [19]	0.5776
BranchLSTM [22]	0.4929
BUT-FIT [23]	0.6167
BLCU_NLP [24]	0.6187
BERT$_{BASE}$	0.5324
BERT$_{LARGE}$	0.5635
RoBERTa$_{BASE}$	0.5715
RoBERTa$_{LARGE}$	0.6301
ELECTRA$_{BASE}$	0.5424
ELECTRA$_{LARGE}$	0.5841

The variation of the experimental results of the six selected transformer-based models with respect to the training loss, evaluation loss and macro-averaged F1-score are shown in Figure 6. It is worth mentioning that the models were overfitted (i.e., the evaluation loss increased while the training kept decreasing) after the third epoch to the training data, and that resulted in acquiring the best F1-scores around the third epoch (step range: 25–45).

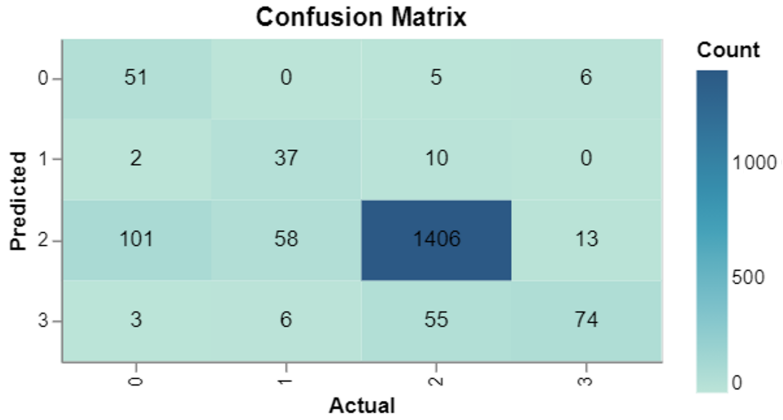

Figure 5. Confusion matrix of the RoBERTa$_{LARGE}$ model on the test SemEval dataset. Label 1 stands for Support; 2, for Deny; 3, for Comment; and 4, for Query.

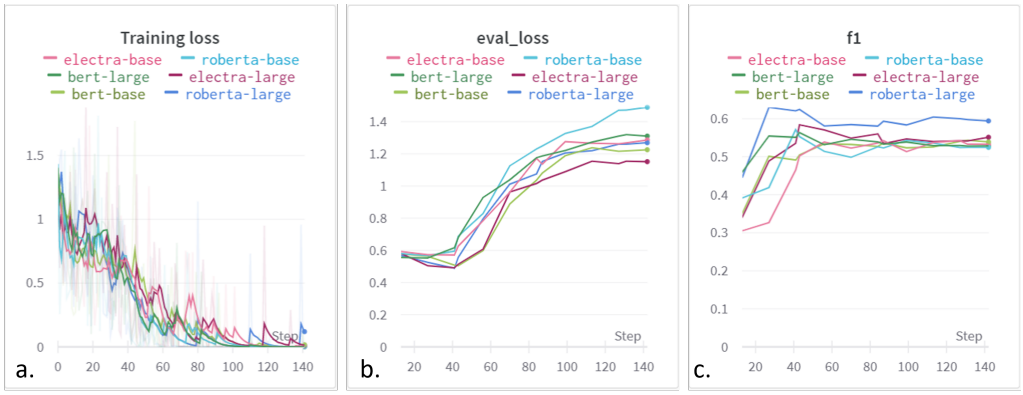

Figure 6. Experimental results for the 6 selected transformer-based models for the stance detection task: (**a**) training loss, (**b**) evaluation loss, and (**c**) macro-averaged F1-score.

4.6. Discussion

A common pattern noticed in all of the different models is that larger model versions performed better in both of the examined tasks. That was somehow expected since larger transformer-based architectures achieve higher performance in several NLP tasks [28–30], and the overall scale of a language model is considered to be far more important than finding its precise architectural hyperparameters [35]. Furthermore, previous studies have shown that even the larger and longest-trained models appear to underfit the training data and would benefit from additional training or from more parameters [29,36], while fine-tuning the architectural hyperparameters is considered to be unimportant compared to the overall scale of the language model.

Another observation is that BERT performs poorly when compared to RoBERTa and ELECTRA; these two models were introduced after BERT and have obtained better results in several NLP tasks [29,30], and especially, RoBERTa is an optimized version of BERT; thus, achieving better results is expectable.

Another remarkable observation is that RoBERTa surpassed, by around 5%, the F1-score achieved by ELECTRA in the SDQC stance detection task, which is extremely high compared to the accuracy obtained on the SUBJ dataset and other previously reported results [30]. This could be explained by the fact that RoBERTa is pretrained on many datasets including OpenWebText, a dataset that comprises web content extracted from URLs shared on Reddit. Posts on Reddit and Twitter contain arbitrary use of the English language, informal words and expressions, and a lot of abbreviations. Thus, the fact that RoBERTa has already been pretrained on such vocabulary is crucial. Finally, RoBERTa

is also enforced when it comes processing effectively rare words since it has a subword vocabulary of 50K units, while BERT's and ELECTRA's have 30K units.

5. Conclusions

In this paper, we performed a comparative study on exploiting different transformer-based language models for subjectivity analysis and stance detection, using two public available datasets. As expected, larger models led to greater performance, with RoBERTa$_{LARGE}$ and ELECTRA$_{LARGE}$ achieving the best F1-scores and accuracy for stance detection and subjectivity analysis, respectively.

Future steps include further experimentation for stance detection, exploiting a more tree-based structure of the replies included in a discussion thread such as the BranchLSTM approach [22], but using a RoBERTa$_{LARGE}$ model. Furthermore, graph neural networks could also be another option for finding relational dependencies between posts included in large discussion threads and updating their hidden representations through graph-based message passing [37]. Finally, we could examine the performance of modular approaches (i.e., a model having, as features, trustworthiness indicators) against end-to-end model architectures in veracity assessment.

Author Contributions: Conceptualization, P.K. and C.Z.P.; methodology, P.K.; software, P.K. and L.T.; validation, P.K. and L.T.; formal analysis, P.K.; investigation, P.K.; data curation, P.K.; writing—original draft preparation, P.K. and L.T.; writing—review and editing, P.K. and C.Z.P.; visualization, P.K.; supervision, C.Z.P.; project administration, C.Z.P.; funding acquisition, C.Z.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the European Commission's H2020 Innovation Action programme (under project EUNOMIA), grant number 825171.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data supporting the reported results can be found in https://www.cs.cornell.edu/people/pabo/movie-review-data/ (accessed on 25 June 2021) and in https://alt.qcri.org/semeval2019/index.php?id=tasks (accessed on 7 July 2021) for the tasks of subjectivity analysis and stance detection, respectively.

Acknowledgments: The work presented in this paper was supported through the European Commission's H2020 Innovation Action programme under project EUNOMIA (grant agreement no. 825171).

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

References

1. Derczynski, L.; Bontcheva, K.; Liakata, M.; Procter, R.; Wong Sak Hoi, G.; Zubiaga, A. SemEval-2017 Task 8: RumourEval: Determining rumour veracity and support for rumours. In Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), Vancouver, BC, Canada, 3–4 August 2017; pp. 69–76. [CrossRef]
2. Toumanidis, L.; Heartfield, R.; Kasnesis, P.; Loukas, G.; Patrikakis, C. A Prototype Framework for Assessing Information Provenance in Decentralised Social Media: The EUNOMIA Concept. In *E-Democracy—Safeguarding Democracy and Human Rights in the Digital Age*; Katsikas, S., Zorkadis, V., Eds.; Springer International Publishing: Cham, Switzerland, 2020; pp. 196–208. [CrossRef]
3. Williams Kirkpatrick, A. The spread of fake science: Lexical concreteness, proximity, misinformation sharing, and the moderating role of subjective knowledge. *Public Underst. Sci.* **2020**, *30*, 55–74. [CrossRef] [PubMed]
4. Jeronimo, C.L.M.; Marinho, L.B.; Campelo, C.E.C.; Veloso, A.; da Costa Melo, A.S. Fake News Classification Based on Subjective Language. In Proceedings of the 21st International Conference on Information Integration and Web-based Applications & Services, Munich, Germany, 2–4 December 2019; pp. 15–24. [CrossRef]
5. Vieira, L.L.; Jeronimo, C.L.M.; Campelo, C.E.C.; Marinho, L.B. Analysis of the Subjectivity Level in Fake News Fragments. In Proceedings of the Brazilian Symposium on Multimedia and the Web, São Luís, Brazil, 30 November–4 December 2020; pp. 233–240. [CrossRef]

6. Mohammad, S.; Kiritchenko, S.; Sobhani, P.; Zhu, X.; Cherry, C. SemEval-2016 Task 6: Detecting Stance in Tweets. In Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016), San Diego, CA, USA, 16–17 June 2016; pp. 31–41. [CrossRef]
7. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is All You Need. In Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS'17, Long Beach, CA, USA, 4–9 December 2017; pp. 6000–6010. [CrossRef]
8. Wang, S.; Manning, C. Fast dropout training. In Proceedings of the 30th International Conference on Machine Learning, Atlanta, GA, USA, 16–21 June 2013; pp. 118–126.
9. Hinton, G.E.; Srivastava, N.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv* **2012**, arXiv:1207.0580.
10. Pang, B.; Lee, L. A Sentimental Education: Sentiment Analysis Using Subjectivity Summarization Based on Minimum Cuts. In Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics, Barcelona, Spain, 21–26 July 2004. [CrossRef]
11. 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. [CrossRef]
12. Cho, K.; van Merriënboer, B.; Bahdanau, D.; Bengio, Y. On the Properties of Neural Machine Translation: Encoder–Decoder Approaches. In Proceedings of the SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation, Doha, Qatar, 25 October 2014; pp. 103–111. [CrossRef]
13. Amplayo, R.K.; Lee, K.; Yeo, J.; won Hwang, S. Translations as Additional Contexts for Sentence Classification. *arXiv* **2018**, arXiv:1806.05516.
14. Radford, A.; Józefowicz, R.; Sutskever, I. Learning to Generate Reviews and Discovering Sentiment. *arXiv* **2017**, arXiv:1704.01444.
15. Krause, B.; Lu, L.; Murray, I.; Renals, S. Multiplicative LSTM for sequence modelling. *arXiv* **2017**, arXiv:1609.07959.
16. Cer, D.; Yang, Y.; Kong, S.; Hua, N.; Limtiaco, N.; St. John, R.; Constant, N.; Guajardo-Cespedes, M.; Yuan, S.; Tar, C.; et al. Universal Sentence Encoder for English. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 31 October–4 November 2018; pp. 169–174. [CrossRef]
17. Iyyer, M.; Manjunatha, V.; Boyd-Graber, J.; Daumé, H., III. Deep unordered composition rivals syntactic methods for text classification. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, Beijing, China, 26–31 July 2015; pp. 1681–1691. [CrossRef]
18. Bowman, S.R.; Angeli, G.; Potts, C.; Manning, C.D. A large annotated corpus for learning natural language inference. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, 17–21 September 2015; pp. 632–642. [CrossRef]
19. Li, Q.; Zhang, Q.; Si, L. eventAI at SemEval-2019 Task 7: Rumor Detection on Social Media by Exploiting Content, User Credibility and Propagation Information. In Proceedings of the 13th International Workshop on Semantic Evaluation, Minneapolis, MN, USA, 6–7 June 2019; pp. 855–859. [CrossRef]
20. Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G.S.; Dean, J. Distributed representations of words and phrases and their compositionality. In Proceeding of the Conference on Neural Information Processing Systems, Lake Tahoe, NV, USA, 5–10 December 2013; pp. 3111–3119. [CrossRef]
21. Gorrell, G.; Kochkina, E.; Liakata, M.; Aker, A.; Zubiaga, A.; Bontcheva, K.; Derczynski, L. SemEval-2019 Task 7: RumourEval, Determining Rumour Veracity and Support for Rumours. In Proceedings of the 13th International Workshop on Semantic Evaluation, Minneapolis, MN, USA, 6–7 June 2019; pp. 845–854. [CrossRef]
22. Kochkina, E.; Liakata, M.; Augenstein, I. Turing at SemEval-2017 Task 8: Sequential Approach to Rumour Stance Classification with Branch-LSTM. In Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), Vancouver, BC, Canada, 3–4 August 2017; pp. 475–480. [CrossRef]
23. Fajcik, M.; Smrz, P.; Burget, L. BUT-FIT at SemEval-2019 Task 7: Determining the Rumour Stance with Pre-Trained Deep Bidirectional Transformers. In Proceedings of the 13th International Workshop on Semantic Evaluation, Minneapolis, MN, USA, 6–7 June 2019; pp. 1097–1104. [CrossRef]
24. Yang, R.; Xie, W.; Liu, C.; Yu, D. BLCU_NLP at SemEval-2019 Task 7: An Inference Chain-based GPT Model for Rumour Evaluation. In Proceedings of the 13th International Workshop on Semantic Evaluation, Minneapolis, MN, USA, 6–7 June 2019; pp. 1090–1096. [CrossRef]
25. Radford, A.; Narasimhan, K. Improving Language Understanding by Generative Pre-Training. 2018, in press.
26. Ferreira, W.; Vlachos, A. Emergent: A novel data-set for stance classification. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego, CA, USA, 12–17 June 2016; pp. 1163–1168. [CrossRef]
27. Rajpurkar, P.; Jia, R.; Liang, P. Know What You Don't Know: Unanswerable Questions for SQuAD. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, Melbourne, Australia, 15–20 July 2018; pp. 784–789. [CrossRef]
28. 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, Minneapolis, MN, USA, 2–7 June 2019; pp. 4171–4186. [CrossRef]

29. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *arXiv* **2019**, arXiv:1907.11692.
30. Clark, K.; Luong, M.T.; Le, Q.V.; Manning, C.D. ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators. *arXiv* **2020**, arXiv:2003.10555.
31. Zhu, Y.; Kiros, R.; Zemel, R.; Salakhutdinov, R.; Urtasun, R.; Torralba, A.; Fidler, S. Aligning Books and Movies: Towards Story-Like Visual Explanations by Watching Movies and Reading Books. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Los Alamitos, CA, USA, 7–13 December 2015; pp. 19–27. [CrossRef]
32. Wang, A.; Singh, A.; Michael, J.; Hill, F.; Levy, O.; Bowman, S. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, Brussels, Belgium, 1 November 2018; pp. 353–355. [CrossRef]
33. Caccia, M.; Caccia, L.; Fedus, W.; Larochelle, H.; Pineau, J.; Charlin, L. Language gans falling short. *arXiv* **2018**, arXiv:1811.02549.
34. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
35. Kaplan, J.; McCandlish, S.; Henighan, T.; Brown, T.B.; Chess, B.; Child, R.; Gray, S.; Radford, A.; Wu, J.; Amodei, D. Scaling Laws for Neural Language Models. *arXiv* **2020**, arXiv:2001.08361
36. Yang, Z.; Dai, Z.; Yang, Y.; Carbonell, J.; Salakhutdinov, R.R.; Le, Q.V. XLNet: Generalized Autoregressive Pretraining for Language Understanding. In *Advances in Neural Information Processing Systems*; Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2019; Volume 32.
37. Yasunaga, M.; Ren, H.; Bosselut, A.; Liang, P.; Leskovec, J. QA-GNN: Reasoning with Language Models and Knowledge Graphs for Question Answering. *arXiv* **2021**, arXiv:2104.06378.