Unveiling the Truth: A Deep Dive into Claim Identification Methods

Shankha Shubhra Das[⋆], **Pritam Pal**[†] and **Dipankar Das**[⋆]
[⋆]Jadavpur University, Kolkata, India
[†]RCC Institute of Information Technology, Kolkata, India
{shankhasdas07, pritampal522, dipankar.dipnil2005}@gmail.com

Abstract

Claim identification, an important task in the field of natural language processing (NLP) is the stepping stone for more critical NLP tasks such as fact-checking, fake news and misinformation detection from social media and other real-world data. By leveraging advanced deep learning and recent transformer-based models, we investigate two claim identification methods in this article: one is a multilingual claim span detection from social media posts for English, Hindi, Bengali and CodeMixed texts and another is a fusion-based novel multi-task learning (MTL) framework for claim classification along with sentiment and language identification. Our best-performing claim span detection framework achieved an accuracy of around 80% and the best-performing MTL framework provides an F1 score of 0.74 for claim classification.

1 Introduction

The number of social media users has rapidly increased in the past few years. As per data provided by Kemp (2024), India had social media users of around 462 million in January 2024 whereas in 2019 there were around 310 million active social media users (Kemp, 2019). This social media enables different levels of people to express their feelings and opinions independently on any topic or event. However, in this large content of social media posts, it is sometimes difficult to find factual posts that contain some meaningful claims.

With the advancement of Natural Language Processing (NLP) and Artificial intelligence (AI), researchers have done state-of-the-art works on opinion mining or sentiment analysis, emotion analysis etc. in social media content and many other real-world textual data. In contrast, there is limited research was performed on detecting a specific phrase in a text that contains claims (claim span identification). Also, how the claim detection works in a multi-task learning environment is not well explored where we combine different tasks in a single neural network so that learning from one task helps each other in a shared environment.

In this paper, we focused on identifying the specific phrases in a social media post or other real-world text that contain some factual information or claim. Along with that, we proposed a multi-task learning (MTL) model to classify a text that contains a claim or not with additional tasks of sentiment analysis and language identification to specifically check how the claim classification works in a multi-task learning environment.

Our research is motivated by identifying factual information from social media and other real work texts which will be further useful for the verifiability of claims, detecting fake news, misinformation etc. The main contributions in this paper can be summarized as follows:

- We have proposed a multilingual claim span identification framework for Bengali, Hindi, English and CodeMixed texts.
- Followed by this a fusion-based novel multi-task learning framework is proposed for relatively dissimilar genres of tasks: claim, sentiment and language classification.

2 Related Work

Recent advancements of deep learning in the field of NLP have witnessed significant progress in claim span identification, claim classification and MTL. Starting from statistical analysis to machine learning to state-of-the-art transformer-based models such as BERT researchers proposed different methods in the field of claim-related works.

Claim Detection: Pavllo et al. (2018) and Smeros et al. (2019) develop rule-based heuristics for extracting quotes from general and scientific news articles using weakly supervised models. Levy et al. (2014) and Stab et al. (2018) pro-

pose ML models for claim detection and argument mining, providing publicly available datasets for training extraction models. Hassan et al. (2017) and Popat et al. (2017) employ claim classification models with fact-checking portals for verifying political claims.

Zlatkova et al. (2019) focus on claim extraction for images, while Karagiannis et al. (2020) present a framework for statistical claims verification. This approach (Smeros et al., 2021), unlike others, is specifically tailored for claims, utilizing advanced language models with and without contextualized embeddings fine-tuned with domain-specific knowledge and capable of processing various input sources like social media postings, blog posts, or news articles.

Multi-Task Learning: The concept of Multi-task Learning (MTL) was first proposed by Caruana (1997). Ruder (2017) discussed different schemes of MTLs in their paper such as hard parameter sharing, soft parameter sharing etc.

Numerous researchers proposed different MTL frameworks in the field of NLP. Specifically, in claim-related studies, Tzu-Ying Chen (2022) proposed a multi-task learning framework for claim detection and numerical category classification utilizing the transformer-based BERT (Devlin et al., 2019) model.

Besides, Liu et al. (2016), Liu et al. (2017) proposed MTL for text classification utilizing LSTMs and BiLSTMs. An MTL framework for sentiment and sarcasm classification was proposed by Majumder et al. (2019), Savini and Caragea (2020), El Mahdaouy et al. (2021) and Tan et al. (2023).

Singh et al. (2022) combined sentiment, emotion and emoji classification tasks in an MTL framework utilizing transformer based 'XLM-RoBERTa' (Liu et al., 2019) model whereas Del Arco et al. (2021) combined sentiment, emotion, hate speech, offensive language and target (targeting a specific community such as women, black people, LGBT etc.) in a single MTL framework utilizing BERT.

This present article focuses on two claim-related tasks: a multilingual claim span identification framework in real-world social media content and a multi-task learning framework incorporating three relatively dissimilar tasks: claim, sentiment and language identification.

3 Dataset

3.1 Claim Span Identification

To accomplish the claim span identification task, we utilized the JUCSI (Jadavpur University Claim Span Identification) dataset that was specifically provided for our research. This dataset comprises approximately 750 training samples across multiple languages, including English, Hindi, Bengali, and CodeMix. The data predominantly focuses on topics related to COVID-19 vaccines and social distancing measures. Each entry in the dataset includes the original text, an indication of the language used, and the specific span within the text where the claim(s) can be found. This multilingual and topical diversity offers a rich resource for analyzing how different linguistic and cultural contexts handle information related to the pandemic. Figure 1 shows the language-wise data distribution.

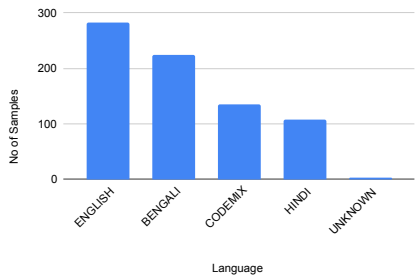

Figure 1: Distribution of Claim Span data

3.2 Claim Classification

The datasets from Rosenthal and McKeown (2012) paper were mainly used for claim classification tasks. This dataset consists of sentences from the LiveJournal blogs and Wikipedia talk pages that have been annotated for opinionated claims. Specifically, there are 2,190 entries from the LiveJournal and 2,197 from the Wikipedia. Each entry is labelled to indicate whether it contains a claim (Yes or No) and includes sentiment annotations for all the texts. Figure 2 provides the distribution of claim data.

3.3 Multi Task Learning

In the MTL framework, we tried to incorporate three tasks (claim classification, sentiment analysis and language identification) in a single neural network. For the claim detection task, the previously mentioned claim detection dataset (Rosenthal and McKeown, 2012) was used. We next calculate the

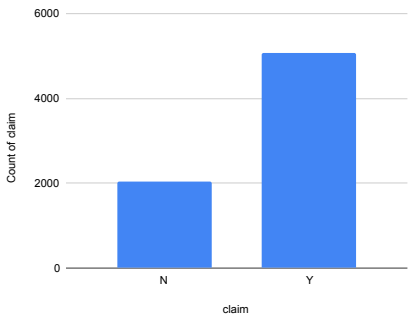

Figure 2: Distribution of Claim Data (N represents there is no claim in the sentence and Y represents there are claims in that sentence.)

sentiment labels for each sample in this dataset using a publicly available distilBERT-based sentiment classification[1] model.

For language identification, a different dataset was collected which is a preprocessed version of WiLI-2018[2], the Wikipedia language identification benchmark dataset. This version includes 22 specific languages: English, Arabic, French, Hindi, Urdu, Portuguese, Persian, Pushto, Spanish, Korean, Tamil, Turkish, Estonian, Russian, Romanian, Chinese, Swedish, Latin, Indonesian, Dutch, Japanese, and Thai. The distribution of sentiment data and language data is presented in Figure 3 and 4 respectively.

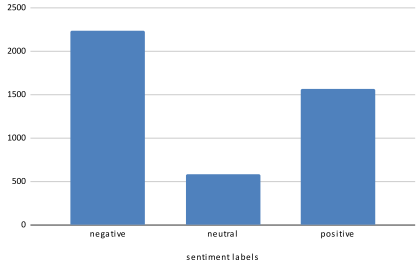

Figure 3: Distribution of sentiment labels

4 Methodologies

This section describes the proposed methodologies of our claim span identification, claim classification and multi-task learning works.

4.1 Claim Span Identification

The main aim of the claim span identification task was to identify the specific phrase in a sentence or

[1]https://bit.ly/multilingual-cased-sentiments-student
[2]https://bit.ly/language-identification-datasst

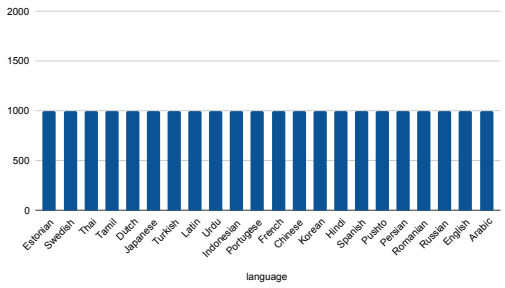

Figure 4: Distribution of language labels

text that contains some claim. In other words, we treat this task as a token classification task where the input would be $\{t_1, t_2, t_3, ..., t_n\}$ where t_i's are the tokens of text and the output would be $\{c_1, c_2, c_3, ..., c_n\}$ where each c_i's $\in \{0, 1, 2\}$ and 1, 2 and 0 represents beginning word of claim, intermediate phrase of claim and outside phrases of claim respectively.

Tokenization: Tokenization involves breaking down a text into smaller units known as tokens. To execute this work, we used different publicly available pre-trained models such as Multilingual-BERT (mBERT) (Devlin et al., 2019), XLM-RoBERTa (Liu et al., 2019), and MuRIL (Khanuja et al., 2021). So, these models' corresponding tokenizers were used to tokenize the input sentence.

B-I-O Tagging: After tokenization, each token must be assigned a B-I-O tag, where 'B' stands for the Beginning of a Claim, 'I' indicates the Inside of a Claim, and 'O' signifies the Outside of a Claim. The use of `return_offsets_mapping=True` in the tokenizer configuration allows us to retrieve the start index and the end index (plus one) for each token within the original text.

Additionally, the start and end indices of each claim span within the original text are calculated and recorded. This enables us to determine which tokens correspond to which parts of the claim. When the offset mapping of a token falls within the range of the start and end indices of a claim span, the appropriate B-I-O tagging is applied to that token. This process ensures that each token is accurately labelled according to its position within or outside the claim spans.

Model Selection: As previously mentioned, to accomplish this work, we used publicly available pre-trained models mBERT, XLM-RoBERTa and MuRIL. The mBERT and XLM-RoBERTa were trained on around 104 and 100 languages respectively including Hindi and Bengali. In contrast,

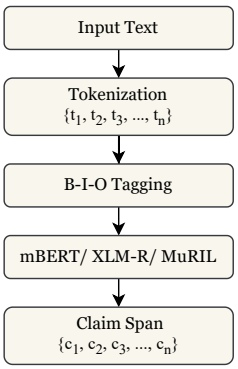

Figure 5: Flow diagram for claim span identification during training.

the MuRIL model was trained explicitly on 14 Indian languages, so this model can learn the Indian contexts in a better way.

Initially, we performed tokenization on the sentences. After this step, each token was annotated with B-I-O tags. The annotated tokens were subsequently input into the model, which generated the predicted claim span(s). Figure 5 shows an abstract overview of our model.

4.2 Multi-Task Learning

Whereas in the claim span identification framework we aim to identify specific phrases that contain certain claims, in MTL our main objective was to classify a text as containing or not containing certain claims with two additional tasks sentiment analysis and language detection.

Given a tokenized sequence S and S is associated with three labels: claim (yes/no), sentiment (positive/negative/neutral) and language (One out of 22 languages as given in Figure 4).

Text Preprocessing: Before diving into the classification, a few basic preprocessing steps were performed in such as i) removal of HTML tags, ii) lowercase conversion, iii) username standardization (convert any Twitter username to '@user'), iv) URL standardization (convert URLs to 'http') and v) conversion of emoji to their corresponding text.

Tokenization: After preprocessing, input text S was tokenized into a sequence of tokens $k_1, k_2, k_3, ..., k_n$. Since sentence lengths vary, we standardize them by padding with zeros to achieve a fixed-size sequence. Consequently, every sentence S transforms into a token sequence $\{k_1, k_2, k_3, ..., k_L\}$ where $L = 300$.

4.2.1 LSTM-based MTL framework

Figure 6(a) and 6(b) represent two MTL frameworks, one is simple MTL with task-specific heads and another is MTL with fusion (MTL$_{fusion}$). Both frameworks utilized bidirectional LSTM (BiLSTM) architecture and pre-trained GloVe (Pennington et al., 2014) embedding with dimension 300.

MTL with Task-Specific Long Heads: As per Figure 6(a) the output of the "GlobalMaxPooling1D" layer is fed into three separate task-specific dense layers of 300 neurons for some task-specific learnings.

$$D_* = ReLU(Z_{GlobalMaxPooling1D})$$
$$D_{dropout*} = Dropout(D_*)$$

where D_* and $D_{dropout*}$ represent the task-specific dense layers and dropout layers respectively.

One possible reason behind using long task-specific heads is the simple fact that the dissimilar tasks have very few things in common among them, and each task needs extra standalone attention. For this reason, we have used more layers in the individual task-specific layers.

MTL$_{fusion}$: Figure 6(b) represents the MTL with fusion technique where the outputs of the task-specific dense layers were passed to dropout layers, and then merge the outputs from the previous layers and feed them into the final task-specific dense layers as follows:

$$Merge_1 = Dropout(D_{claim}) \otimes Dropout(D_{sen})$$
$$Dense_{sen} = ReLU(Merge_1)$$

and,

$$Merge_2 = D_{claim} \otimes D_{sen} \otimes Dropout(D_{lang})$$
$$Dense_2 = ReLU(Merge_2)$$

where $\otimes$ represents the concatenation of the outputs of the dense or dropout layers.

4.2.2 BERT-based MTL framework

Figure 6(c) represents the MTL framework utilizing the pre-trained 'multilingual BERT base uncased' model where the tokenized sequences (input_ids) along with the attention masks which were generated by the 'BertTokenizer' were passed as an input to the BERT model. Next, the 'PoolerOutput' of the BERT model was passed to a dropout layer of 0.1. Then the output of the dropout layer

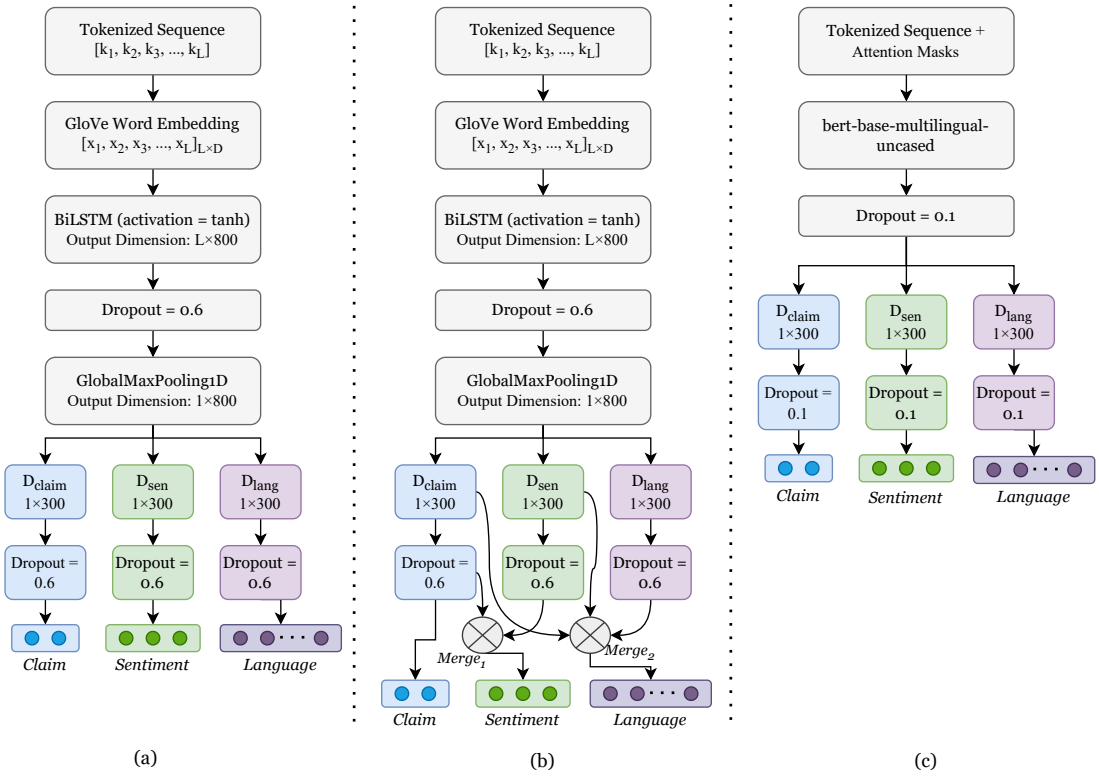

Figure 6: Proposed MTL frameworks. (a) LSTM-based, (b) LSTM with fusion, (c) BERT-based

was passed to three separate dense layers with 300 neurons followed by a dropout of 0.1.

$$D_* = ReLU(Dropout(BERT_{pooler\_output}))$$
$$D_{dropout*} = Dropout(D_*)$$

4.2.3 Classification:

We used separate dense layers for classification in all MTL frameworks. For MTL with long task-specific heads and the BERT-based MTL, the outputs of the individual dropout layers were fed into task-specific dense layers which use softmax as their activation function.

$$P_* = softmax(D_{dropout*})$$

Here P_* represents the probability values for each task-specific output layer.

For MTL with task-specific dense layers and fusion, the output of $Dropout_{claim}$ was fed as an input to the claim detection layer, fed the output of $Dense_1$ as an input to the sentiment classification layer and fed the output of $Dense_2$ as an input to language identification task layer.

$$P_{claim} = softmax(Dropout_{claim})$$
$$P_{sen} = softmax(Dense_1)$$
$$P_{lang} = softmax(Dense_2)$$

Where P_{claim}, P_{sen} and P_{lang} represent the probability value for each class of claim, sentiment and language classification.

4.3 Training

To accomplish the training process, both the JUCSI dataset and the MTL dataset were split into a 7:2:1 ratio where 70% of the data was used for training, 20% of data was taken as validation split and 10% of data was chosen for testing.

The AdamW (Loshchilov and Hutter, 2019) optimizer was chosen to train the claim span identification framework with a learning rate of 2e-5 and trained the models up to 4 epochs.

For the Multi-task loss function, we used the '`SparseCategoricalCrossEntropy`' loss function with Adam (Kingma and Ba, 2014) optimizer and learning rate of 5e-4 and 3e-5 for BiLSTM and BERT models respectively and monitored the loss for validation split for the dataset.

$$L_{total} = \sum_{i=1}^{K} L_i$$

Where L_i is the loss for different tasks and K is the number of tasks. To train the proposed MTL models we had initially taken 50 epochs but used the

'EarlyStopping' method provided by TensrFlow to prevent overfitting during the training process.

5 Experiment and Result

5.1 Experimental Setup

To accomplish the claim span identification task, all the previously mentioned pre-trained models and their corresponding tokenizers were imported from HuggingFace and the models were trained using the libraries of HuggingFace and PyTorch.

For multi-task learning, we used the libraries from TensorFlow and Keras to develop the proposed models and used the Collaboratory environment to train the proposed frameworks.

5.2 Result

5.2.1 Claim Span Identification

Among the previously mentioned three pre-trained models (mBERT, XLM-RoBERTa and MuRIL) the XLM-RoBERTa model identified claim spans more precisely than mBERT and MuRIL models with an accuracy of 0.807 and F1-score of 0.541. This performance shows an improvement of 7.6% and 0.5% in accuracy and 7.2% and 0.9% in the F1-score compared to the MuRIL and mBERT models respectively. The overall results for three models are provided in Table 1

Model	Accuracy	F1
mBERT	0.803	0.536
XLM-RoBERTa	**0.807**	**0.541**
MuRIL	0.746	0.502

Table 1: Results for different models in claim span identification.

5.2.2 Multi-Task Learning

Here we compare and contrast the performance of claim classification in different MTL frameworks with the single-task learning (STL) framework. Along with the claim + sentiment + language combination of MTL, we developed all the other combinations of MTLs such as claim + sentiment and claim + language and reported the results in Table 2 for both BERT and BiLSTM models.

Furthermore, for additional sentiment and language tasks, we also developed all the combinations of MTLs along with the STL frameworks and reported the results in Tables 3 and 4 for sentiment and language classification tasks respectively.

Model	Task	Precision	Recall	F1
BiLSTM	STL	0.711	0.711	0.711
	claim + sen	0.709	0.709	0.709
	claim + lang	0.588	0.567	0.542
	MTL	0.589	0.564	0.534
	MTL$_{fusion}$	0.610	0.599	0.590
BERT	STL	0.744	0.732	0.730
	claim + sen	**0.755**	**0.742**	**0.740**
	claim + lang	0.753	**0.742**	**0.740**
	MTL	0.738	0.716	0.711

Table 2: Result of claim classification of STL and MTL framework

Model	Task	Precision	Recall	F1
BiLSTM	STL	0.660	0.669	0.664
	claim + **sen**	0.709	0.642	0.664
	sen + lang	0.571	0.589	0.576
	MTL	0.597	0.528	0.550
	MTL$_{fusion}$	0.630	0.566	0.586
BERT	STL	**0.796**	0.748	**0.762**
	claim + **sen**	0.756	0.750	0.750
	sen + lang	0.730	**0.755**	0.740
	MTL	0.740	0.702	0.717

Table 3: Result of sentiment classification of STL and MTL framework

It is noticeable from Tables 2, 3 and 4 that, in all the claim, sentiment and language identification tasks, the BERT-based frameworks provide superior performance compared to the BiLSTM-based frameworks in both MTLs and STL.

The claim classification task failed to achieve the best performance in both BiLSTM and BERT-based MTL frameworks. However, the claim + sentiment and claim + language combination of MTL achieved the best recall and F1-score of 0.742 and 0.740 respectively and the best precision with 0.755 was achieved by only the claim + sentiment combination of MTL. Additionally, the BERT-based best MTL framework provides an F1-score improve-

Model	Task	Precision	Recall	F1
BiLSTM	STL	0.788	0.738	0.723
	sen + **lang**	0.746	0.697	0.695
	claim + **lang**	0.687	0.692	0.681
	MTL	0.762	0.705	0.715
	MTL$_{fusion}$	0.722	0.706	0.707
BERT	STL	0.988	0.980	0.983
	sen + **lang**	**0.990**	**0.991**	**0.990**
	claim + **lang**	0.984	0.981	0.982
	MTL	**0.990**	0.983	0.986

Table 4: Result of language classification of STL and MTL framework

Original Claim	XLM-R	MuRIL	mBERT
'Under which provision you got re elected as RS member inspite of getting defeated in #BengalElection2021	'Under which provision you got re elected as RS member inspite of getting defeated in #BengalElection2021	'@ swapan55 Under which provision you got re elected as RS member inspite of getting defeated in	'@', 'Under which provision you got re elected as RS member inspite of getting defeated in # BengalElection2021'
jankibaat1 and Pardip would continue to bark for the next 6 months to make that a reality!	First BJPee would establish a fake theory! Then Low Level Dallals, @jankibaat1 and Pardip would continue to bark for the next 6 months to make that a reality	First BJPee would establish a fake theory! Then Low Level Dallals, @jankibaat1 and Pardip would continue to bark for the next 6 months to make that a reality	First BJPee would establish a fake theory! Then Low Level Dallals, @jankibaat1 and Pardip would continue to bark for the next 6 months to make that a reality!
they have spent huge amount in #BengalElection2021 for BJP	not expect that #JIO will solve your problems', 'they are making fool to the customers as they have spent huge amount in #BengalElection2021 for BJP	##jio Do not expect that # JIO will solve your problems..they are making fool to the customers as they have spent huge amount in	have spent huge amount in # BengalElection2021 for BJP

Table 5: Few examples of identified claim spans in different models.

ment of 3.92% compared to the best-performing framework in BiLSTM.

In the case of sentiment classification, the best Precision and F1 scores of 0.796 and 0.762 were achieved by the BERT-based STL framework and the best recall score was provided by the sentiment + language combination of the MTL framework.

The language identification task significantly improves performance in the BERT-based frameworks with an F1-score of 0.99 in sentiment + language combination of MTL whereas the BiLSTM-based best-performing framework (STL) achieved an F1-score of only 0.723.

6 Error Analysis

6.1 Claim Span Identification

Although the evaluation metrics indicate that XLM-RoBERTa performs the best overall, this is not always consistent for claim span identification. Table 5 presents three examples to highlight the strengths and weaknesses of our models. For the first sentence, XLM-RoBERTa achieved perfect results, whereas MuRIL and mBERT included a few extra words at the beginning. In the second sentence, all models performed poorly, capturing more words than the actual claim span. For the third sentence, the mBERT model performed the best, accurately identifying the claim span, while XLM-RoBERTa and MuRIL captured more than the necessary span.

6.2 Multi-Task Learning

Table 6 presents some examples of predicted labels from both the STL and MTL frameworks with their ground truth labels.

From Table 6, in example S_1, it is seen that although claim and language labels are correctly assigned, the MTL (BiLSTM) framework failed to predict the positive sentiment of the sentence.

In sentence S_2, for claim detection, STL (BiLSTM) and MTL$_{fusion}$ (BiLSTM) frameworks failed to predict the claim correctly, but the MTL (BiLSTM), STL (BERT) and MTL (BERT) frameworks did. For sentence S_3, only the STL (BiLSTM) model correctly predicted the claim but the MTL (BiLSTM) models couldn't. However, the BERT-based both STL and MTL frameworks correctly predict the proper claim labels. The MTL (BiLSTM) framework also incorrectly predicts it as a sentence in Dutch whereas it is an English sentence.

Despite the superior performance of the BERT-based frameworks, in some cases, the BiLSTM-based MTL framework correctly detects its labels where BERT cannot. For example, in S_3 and S_4, the MTL (BiLSTM) framework correctly predicts its actual label whereas the other frameworks failed to predict the correct label.

7 Conclusion

In this article, we studied two schemes of claim identification strategy, first a claim span identification framework utilizing transformer-based pre-trained models followed by an MTL framework for claim, sentiment and language classification.

In future, we'll extend the existing claim span and MTL dataset to validate the robustness of the proposed frameworks. Additionally, we are planning to incorporate the claim span identification task in the MTL framework.

			Claim		Sentiment		Language	
Id	**Text**	**Task**	**True**	**Pred**	**True**	**Pred**	**True**	**Pred**
S_1	I will admit it has less than the Sabbath albums before it, but it still very much holds onto the blues	STL(BiLSTM)	yes	yes	pos	pos	eng	eng
		MTL(BiLSTM)	yes	yes	pos	neg	eng	eng
		MTL_F(BiLSTM)	yes	yes	pos	pos	eng	eng
		STL(BERT)	yes	yes	pos	pos	eng	eng
		MTL(BERT)	yes	yes	pos	pos	eng	eng
S_2	Maybe I could do my own statistics.	STL(BiLSTM)	yes	no	neu	neu	eng	eng
		MTL(BiLSTM)	yes	yes	neu	neu	eng	indo
		MTL_F(BiLSTM)	yes	no	neu	neg	eng	eng
		STL(BERT)	yes	yes	neu	pos	eng	eng
		MTL(BERT)	yes	yes	neu	pos	eng	eng
S_3	Have not got around to sorting out the history yet.	STL(BiLSTM)	no	no	neg	neg	eng	eng
		MTL(BiLSTM)	no	yes	neg	neg	eng	dut
		MTL_F(BiLSTM)	no	yes	neg	pos	eng	eng
		STL(BERT)	no	no	neg	neu	eng	eng
		MTL(BERT)	no	no	neg	neu	eng	dut
S_4	müller mox figura centralis circulorum doctorum vindobonesium fiebat quibus intererant petrus	STL(BiLSTM)	no	no	neu	neg	lat	lat
		MTL(BiLSTM)	no	yes	neu	neu	lat	por
		MTL_F(BiLSTM)	no	no	neu	neg	lat	spa
		STL(BERT)	no	yes	neu	neg	lat	lat
		MTL(BERT)	no	no	neu	pos	lat	lat

Table 6: Examples of predictions in STL and MTL frameworks. (red coloured texts define wrong predictions)

8 Limitations

Upon performing all experiments and analysing the results, we delve into a few noteworthy issues for claim span identification and MTL framework.

8.1 Claim Span Identification

Although we developed the claim span identification task for English, Hindi, Bangla and CodeMixed text, our dataset was relatively small (around 750 samples). To thoroughly validate the overall performance of our models, a larger dataset is necessary. Additionally, we have not explored other potentially effective models such as GPT or BERT-large. Further, we need to perform more hyperparameter tuning to enhance the models' performance. Our current training is based solely on social media data; in the future, we plan to extend our training to other types of texts, such as news articles and online blogs, to evaluate and improve the models' versatility and robustness across various domains.

8.2 Multi-Task Learning

Firstly, it is observed that the performances of dissimilar tasks are not that good in our MTL setting. This is because learning from one task cannot help other tasks properly in dissimilar tasks, and we see a performance drop in the MTL frameworks.

Secondly, the MTL framework is only limited to two models BiLSTM and BERT. Also, we haven't developed any fusion-based MTL framework using the BERT model In future, we'll try to develop a fusion-based MTL framework by exploring other state-of-the-art transformer-based models.

And lastly, an imbalance of claim and sentiment data in the final dataset may include performance bias in their corresponding tasks.

References

Rich Caruana. 1997. Multitask learning. *Machine Learning*, 28:41–75.

Flor Miriam Plaza Del Arco, Sercan Halat, Sebastian Padó, and Roman Klinger. 2021. Multi-Task Learning with Sentiment, Emotion, and Target Detection to Recognize Hate Speech and Offensive Language. *arXiv (Cornell University)*.

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.

Abdelkader El Mahdaouy, Abdellah El Mekki, Kabil Essefar, Nabil El Mamoun, Ismail Berrada, and Ahmed Khoumsi. 2021. Deep multi-task model for sarcasm detection and sentiment analysis in Arabic language. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 334–339, Kyiv, Ukraine (Virtual). Association for Computational Linguistics.

Naeemul Hassan, Fatma Arslan, Chengkai Li, and Mark Tremayne. 2017. Toward automated fact-checking: Detecting check-worthy factual claims by claimbuster. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '17, page 1803–1812, New York, NY, USA. Association for Computing Machinery.

Georgios Karagiannis, Mohammed Saeed, Paolo Papotti, and Immanuel Trummer. 2020. Scrutinizer: A mixed-initiative approach to large-scale, data-driven claim verification.

Simon Kemp. 2019. Digital 2019: India — DataReportal – Global Digital Insights.

Simon Kemp. 2024. Digital 2024: India — DataReportal – Global Digital Insights.

Simran Khanuja, Diksha Bansal, Sarvesh Mehtani, Savya Khosla, Atreyee Dey, Balaji Gopalan, Dilip Kumar Margam, Pooja Aggarwal, Rajiv Teja Nagipogu, Shachi Dave, Shruti Gupta, Subhash Chandra Bose Gali, Vish Subramanian, and Partha Talukdar. 2021. Muril: Multilingual representations for indian languages.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv (Cornell University)*.

Ran Levy, Yonatan Bilu, Daniel Hershcovich, Ehud Aharoni, and Noam Slonim. 2014. Context dependent claim detection. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1489–1500, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. 2016. Recurrent neural network for text classification with multi-task learning. *arXiv (Cornell University)*, pages 2873–2879.

Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. 2017. Adversarial multi-task learning for text classification. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1–10, Vancouver, Canada. Association for Computational Linguistics.

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.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization.

Navonil Majumder, Soujanya Poria, Haiyun Peng, Niyati Chhaya, Zhaoxia Wang, and Alexander Gelbukh. 2019. Sentiment and sarcasm classification with multitask learning. *IEEE Intelligent Systems*, 34(3):38–43.

Dario Pavllo, Tiziano Piccardi, and Robert West. 2018. Quootstrap: Scalable unsupervised extraction of quotation-speaker pairs from large news corpora via bootstrapping.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Kashyap Popat, Subhabrata Mukherjee, Jannik Strötgen, and Gerhard Weikum. 2017. Where the truth lies: Explaining the credibility of emerging claims on the web and social media. In *Proceedings of the 26th International Conference on World Wide Web Companion*, WWW '17 Companion, page 1003–1012, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

Sara Rosenthal and Kathleen McKeown. 2012. Detecting opinionated claims in online discussions. In *2012 IEEE Sixth International Conference on Semantic Computing*, pages 30–37.

Sebastian Ruder. 2017. An overview of multi-task learning in deep neural networks. *CoRR*, abs/1706.05098.

Edoardo Savini and Cornelia Caragea. 2020. A multitask learning approach to sarcasm detection (student abstract). *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(10):13907–13908.

Gopendra Vikram Singh, Dushyant Singh Chauhan, Mauajama Firdaus, Asif Ekbal, and Pushpak Bhattacharyya. 2022. Are emoji, sentiment, and emotion Friends? a multi-task learning for emoji, sentiment, and emotion analysis. In *Proceedings of the 36th Pacific Asia Conference on Language, Information and Computation*, pages 166–174, Manila, Philippines. Association for Computational Linguistics.

Panayiotis Smeros, Carlos Castillo, and Karl Aberer. 2019. Scilens: Evaluating the quality of scientific news articles using social media and scientific literature indicators. In *The World Wide Web Conference*, WWW '19, page 1747–1758, New York, NY, USA. Association for Computing Machinery.

Panayiotis Smeros, Carlos Castillo, and Karl Aberer. 2021. Sciclops: Detecting and contextualizing scientific claims for assisting manual fact-checking. In

Proceedings of the 30th ACM International Conference on Information & Knowledge Management, CIKM '21, page 1692–1702, New York, NY, USA. Association for Computing Machinery.

Christian Stab, Tristan Miller, Benjamin Schiller, Pranav Rai, and Iryna Gurevych. 2018. Cross-topic argument mining from heterogeneous sources. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3664–3674, Brussels, Belgium. Association for Computational Linguistics.

Yik Yang Tan, Chee-Onn Chow, Jeevan Kanesan, Joon Huang Chuah, and YongLiang Lim. 2023. Sentiment Analysis and Sarcasm Detection using Deep Multi-Task Learning. *Wireless Personal Communications*, 129(3):2213–2237.

Hui-Lun Lin Chia-Tzu Lin Yung-Chung Chang Chun-Wei Tung Tzu-Ying Chen, Yu-Wen Chiu. 2022. Tmunlp at the ntcir-16 finnum-3 task: Multi-task learning on bert for claim detection and numeral category classification.

Dimitrina Zlatkova, Preslav Nakov, and Ivan Koychev. 2019. Fact-checking meets fauxtography: Verifying claims about images. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2099–2108, Hong Kong, China. Association for Computational Linguistics.