MuLX-QA: Classifying Multi-Labels and Extracting Rationale Spans in Social Media Posts

SOHAM PODDAR, Indian Institute of Technology, Kharagpur, India
RAJDEEP MUKHERJEE, Indian Institute of Technology, Kharagpur, India
AZLAAN SAMAD, Leibniz University, Hanover, Germany
NILOY GANGULY, Indian Institute of Technology, Kharagpur, India
SAPTARSHI GHOSH, Indian Institute of Technology, Kharagpur, India

While social media platforms play an important role in our daily lives in obtaining the latest news and trends from across the globe, they are known to be prone to widespread proliferation of harmful information in different forms leading to misconceptions among the masses. Accordingly, several prior works have attempted to tag social media posts with labels/classes reflecting their veracity, sentiments, hate content, and so on. However, in order to have a convincing impact, it is important to additionally extract the post snippets on which the labelling decision is based. We call such a post snippet the *rationale*. These rationales significantly improve human trust and debuggability of the predictions, especially when detecting misinformation or stigmas from social media posts. These rationale spans or snippets are also helpful in post-classification social analysis, such as for finding out the target communities in hate-speech, or for understanding the arguments or concerns against the intake of vaccines. Also it is observed that a post may express multiple notions of misinformation, hate, sentiment, and the like. Thus, the task of determining (one or multiple) labels for a given piece of text, along with the *text snippets explaining the rationale behind each of the identified labels* is a challenging *multi-label, multi-rationale* classification task, which is still nascent in the literature.

While *transformer*-based encoder-decoder generative models such as BART and T5 are well suited for the task, in this work we show how a relatively simpler *encoder-only* discriminative question-answering (QA) model can be effectively trained using *simple template-based questions* to accomplish the task. We thus propose *MuLX-QA* and demonstrate its utility in producing (label, rationale span) pairs in two different settings: *multi-class* (on the *HateXplain* dataset related to hate speech on social media), and *multi-label* (on the *CAVES* dataset related to COVID-19 anti-vaccine concerns). *MuLX-QA outperforms heavier generative models* in both settings. We also demonstrate the relative advantage of our proposed model MuLX-QA over strong baselines when trained with limited data. We perform several ablation studies, and experiments to better understand the effect of training MuLX-QA with different question prompts, and draw interesting inferences. Additionally, we show that MuLX-QA is effective on social media posts in resource-poor non-English languages as well. Finally, we perform a qualitative analysis of our model predictions and compare them with those of our strongest baseline.

CCS Concepts: • **Computing methodologies** → **Information extraction**; **Neural networks**; • **Applied computing** → *Sociology*; *Health informatics;*

The first author (S. Poddar) is supported by the Prime Minister's Research Fellowship (PMRF) from the Ministry of Education, Government of India.

Authors' addresses: S. Poddar, R. Mukherjee, N. Ganguly, and S. Ghosh, Department of Computer Science and Engineering, Indian Institute of Technology, Kharagpur 721302, India; e-mails: sohampoddar@kgpian.iitkgp.ac.in, saptarshi@cse.iitkgp.ac.in; A. Samad, L3S Research Center, Leibniz University, Hannover 30167, Germany; e-mail: azlaan.samad@l3s.de.

ACM 1559-1131/2024/05-ART35
https://doi.org/10.1145/3653303

Additional Key Words and Phrases: Multi-label classification, rationale extraction, question answering model, vaccine concerns identification, hate-speech detection

ACM Reference Format:
Soham Poddar, Rajdeep Mukherjee, Azlaan Samad, Niloy Ganguly, and Saptarshi Ghosh. 2024. MuLX-QA: Classifying Multi-Labels and Extracting Rationale Spans in Social Media Posts. *ACM Trans. Web* 18, 3, Article 35 (May 2024), 26 pages. https://doi.org/10.1145/3653303

1 INTRODUCTION

Social media platforms such as Facebook, Gab, and Twitter have become crucial sources of near real-time information about almost anything happening around the world. However, various types of social stigma also proliferate on these platforms, including *hate speech* [44], *rumors* including conspiracy theories and anti-vaccine concerns in the COVID-19 era [58, 59], and so on. While prior works have focused on classifying various types of untrustworthy/harmful content from social media posts [50], most of them focus on predicting the labels without giving any rationale behind the label assignments. This lack of transparency in their decision-making process raises questions on the applicability of these models in real-world applications [14], since the label predictions made by them may not be fully trusted in the absence of corresponding rationales/explanations.

The need to explain model predictions becomes particularly crucial when we deal with detecting potentially harmful content over social media platforms, such as hate speech [4, 57], unverified theories around the intake of COVID-19 vaccines [9, 26, 80], and the like that can have far-reaching negative consequences on the masses. Social media platforms often use classification models to flag and remove such harmful content. The classifications may or may not be correct, and there have been numerous cases where non-harmful posts have been mistakenly removed just because of the presence of certain 'trigger-words.'[1] In such a context, it becomes important to provide a rationale to explain why the post was flagged, in case the author of the post decides to appeal against its flagging. Moreover, with the rise in the use of AI models affecting peoples' daily lives, laws such as the General Data Protection Regulation enforces the right to explanation,[2] thus calling for interpretable models.

It is to be noted here that while some prior works have attempted to generate rationales/explanations for labels from within the given text (to be classified) [44, 59], a few other works have tried to generate explanations from outside the input text [60, 86]. In this work, we focus on extracting the 'rationale' spans (snippets of text) from within the source text. This is important since we get to understand which portion of the text is responsible for the model predicting a certain class/label. For instance, a rationale extracted from within a social media post, predicted as harmful, can be shown to its author if he/she challenges the model prediction. In the context of social science, these rationale spans extracted from within the posts are also helpful in understanding the specific opinions of users, varying from domain to domain. For instance, while classifying hate-speech, these rationales can be further analysed to identify the target community of hate speech.

In this work, we deal with two domains/types of harmful information on the social media—(i) hate speech and (ii) anti-vaccine content—in two settings (single-label vs. multi-label), as detailed below (the domains and datasets are detailed in Section 3). The first domain is that of analyzing offensive/hate speech content from social media posts. While online hatred is unfortunately widespread these days, hate is often targeted towards specific communities. Hence, identifying the

[1] https://www.internetgovernance.org/2020/12/23/exploring-the-problems-of-content-moderation-on-social-media
[2] https://www.privacy-regulation.eu/en/r71.htm

rationales (spans from the text portraying the hate/offensive content) behind classifying a post as hateful will help in various applications such as identifying the target communities, designing *counter speech* [43] that might help to mitigate the issue, and so on. In this domain, we perform experiments on the *HateXplain* dataset [44] where each post is to be assigned a single label from the set {*Hateful*, *Offensive*, *Normal*}, together with providing a rationale for the label in the form of an extract/span from the text. Thus, this is a joint task involving a single-label (multi-class) classification task and the task of providing a rationale for the assigned label.

The second domain relates to the *concerns* among the masses against the intake of COVID-19 vaccines. Since the onset of the pandemic, the online discourse around vaccines [29] has escalated greatly, with an increasing number of people voicing their hesitancy, over social media platforms, about taking COVID-19 vaccines [8, 58]. Most prior works have been limited to classifying vaccine-related social media posts into broad categories of *Pro-Vaccine*, *Neutral,* and *Anti-Vaccine*, without investigating the *specific objection(s) towards vaccines* that are mentioned in the posts, such as the potential side-effects, suspected ineffectiveness, political reasons, and the like.[3] To bridge this gap, our prior work [59] developed the *CAVES* dataset that opened up possibilities for exploring supervised methods for the challenging task of jointly detecting anti-vax objections (possibly multiple) expressed in a given tweet, together with extracting their corresponding rationales as spans from the tweet text. In contrast to the previous domain, this is a joint task involving a *multi-label* classification task, along with the task of providing a rationale for every label assigned to a post. Extracting the rationales is essential to understand the specific objections of the users, so that they can be given suitable counter-arguments (tailored to their specific objections) to nudge them towards vaccination. Rationale extraction is also beneficial when some part of the tweet contains some genuine concern about the vaccine while the other part talks about conspiracies.

For both the domains stated above, the task we address is that of *(label, rationale span) tuple prediction in a multi-label setting,* that was introduced in our prior work [59] but has been explored very little in the literature. The task is not only novel in the context of ML/NLP, but especially challenging since it requires extracting *rationales/explanations in a multi-label setting, where a separate rationale/explanation is to be provided for each of the predicted labels for a particular input text.* Though providing explanations for label prediction has been studied extensively [41, 45, 65, 69], most of the prior works deal with explanations in a single-label setting. To our knowledge, Mullenbach et al. [51] is the only prior work that attempted to provide explanations in a multi-label setting; however, the explanation prediction part was *unsupervised*, as no dataset existed at that time containing separate explanations for each label associated with a piece of text. Prior studies have also deliberated the possibility of explaining model predictions using intermediate representations or attention weights [28, 44, 83]. Motivated by recent studies [53], we however, formulate the task as a span extraction problem where the model is trained to jointly extract a sub-string of the input text as a natural language explanation behind the corresponding label prediction.

We thus propose the **MuLX-QA** (**Mu**lti-**L**abel e**X**plainable classifer using **Q**uestion **A**nswering) framework for the (label, rationale span) tuple prediction task described above. *MuLX-QA* uses a transformer-based framework as its backbone that is trained with carefully designed (but simple) *prompt-based* questions to extract sub-strings of the input text as rationales or explanations behind the label predictions. The training uses a contrastive method, that is, given an input text and its true labels (possibly multiple), positive and contrastive/negative questions (as shown later in Table 5) are used to train the model. Through exhaustive experiments on both CAVES as well as HateXplain datasets, we demonstrate that *MuLX-QA*, despite being a 'simpler' *encoder-only* discriminative

[3]We use the term 'a nti-vaccine (anti-vax) concern' to refer to a specific objection towards vaccination as expressed by the author of a social media post, such as the potential side-effects, suspected ineffectiveness, political reasons, and so on.

architecture, comprehensively outperforms more complex and computationally expensive encoder-decoder generative models based on BART [34] or T5 [61]. Not only do we achieve state-of-the-art results on both datasets (refer to Section 5.4), we also exhibit the robustness of *MuLX-QA* in different scenarios over strong encoder-decoder baselines (in Section 6.2). We also perform a qualitative comparison of the results given by MuLX-QA and the baselines in Section 6.3.

Limitations of Prior Work. We focus on the task of explainable multi-label classification *with separate rationales (explanations) to be extracted for each predicted label.* This task was demonstrated to be challenging in our earlier work [59], and there has been very little research on this task. Prior works have almost always considered explainable classification in a single-label setting. The only relevant work to our knowledge is Mullenbach et al. [51], which is a CNN-based model (named *CAML*) using Word2Vec embeddings, and is outmatched by modern Transformer-based methods. Importantly, though Mullenbach et al. [51] provided explanations in a multi-label setting, their model was unsupervised in the explanation generation portion. To our knowledge, our work is the first which proposes to train multi-label classification models in a supervised setting.

Also, previous studies have leveraged *transformers*-based **question-answering (QA)** models to formulate tasks such as NER [36], entity-relation extraction [35], and summarization [46] as a machine reading comprehension (or extractive QA) task. However, to the best of our knowledge, no prior work has adapted QA models for the task of *explainable classification.*

Novelty and Contributions of This Work. In this work, we propose a novel QA-based model that has not been previously explored for multi-label explainable text classification. Though it is well known that a QA-based method can extract spans from text, the novelty of our approach lies in (i) training it suitably with *contrastive examples*, which enables it to predict multiple labels, as well as the absence/non-association of a label for a given text, (ii) suitable questioning/prompting and output formatting to jointly extract (label, rationale) tuples, and (iii) making use of simple and generic templates to frame the questions, which can be easily extended to new datasets by incorporating their respective metadata information. Our proposed model MuLX-QA outperforms more complex and heavier encoder-decoder models despite being an encoder-only model in several settings, as we show later in this article.

Our work therefore makes the following contributions: (1) We design MuLX-QA, a novel approach for using a QA model for extracting labels and explanations jointly in a multi-label setting.[4] (2) We benchmark MuLX-QA on two challenging datasets containing different types of misinformation prevalent on social media, and our model outperforms several strong state-of-the-art baselines, including heavier encoder-decoder models, on both the datasets. (3) We also conduct several analyses on MuLX-QA, including ablation studies, and experiments to understand its behaviour in different settings, such as how its performance varies with the number of contrastive/negative questions, with different question prompts, and with the training data size.

Note that, we introduced the CAVES dataset in our prior work [59] along with the ‘explainable multi-label classification’ task, i.e., extracting multiple tuples of (label, rationale) together. We also benchmarked some standard Transformer-based models on the CAVES dataset in [59]. This work builds upon our prior work in three key ways, as follows: First, in this work, we perform our experiments not only on CAVES but also on HateXplain [44] containing a different type of harmful content (hate speech). Second, we propose MuLX-QA which achieves state-of-the art results on both CAVES as well as HateXplain. We also conduct several types of analyses with our proposed model. Third, we compare our proposed model with even stronger encoder-decoder baselines (Unified-BART and Paraphrase), compared to the baselines used in [59].

[4]Implementation of our model is available at https://github.com/sohampoddar26/MuLX-QA.

To summarise, we motivate the societal importance of explaining label predictions through rationale span extraction, especially when dealing with potentially harmful content over social media platforms, and propose a novel QA-based framework for the challenging task of multi-label explainable classification where separate explanations are to be extracted for each predicted label.

The rest of this article is structured as follows: Section 2 describes the related works, followed by description of the datasets and the tuple-prediction task in Section 3. Our proposed methodology is then detailed in Section 4. Section 5 presents the experimental results on the two datasets and Section 6 presents various analyses of our model and its predictions. Section 7 concludes this article.

2 RELATED WORKS

In this section, we briefly discuss some prior works on multi-label classification and explanation prediction, and how these problems have been applied to social media data. We also discuss how QA models have been used in the literature.

Multi-Label Classification has been studied for years [75, 93]; the reader is referred to the surveys [38, 74] for more details. Different paradigms of models have been tried for performing this task such as hierarchical networks [81], sequence generation [89], transformers [21], multi-task learning [48], and zero-shot learning [33]. Multi-label classification has also been applied extensively to various domain-specific applications, such as on image data [15, 16, 91], in the medical domain [21, 51], disaster mitigation domain [3, 64] and detection of emotions/sentiments [7, 48, 68]. Models that predict labels for multi-lingual tweets have also been developed [64]. There have also been some works on code-mixed social media data that contains a mix of English with other languages like Hindi, Tamil, Telegu, and the like. Methods for tasks such as hate speech detection [67, 70] and sentiment analysis [13, 24] have been developed for such code-mixed data.

In the broad domain of social media, there exist several tasks that boil down to the multi-label classification problem. For example, in the hate-speech domain, the target/category classification is a multi-label classification problem [27, 90]. The SemEval-2018 Task on multi-emotion classification [47] enables classification of 11 different types of emotion categories from tweets [6, 48]. Some prior work also perform multi-label classification to identify users' interests from Reddit data [22]. Finally, our prior work [59] developed the CAVES dataset to enable multi-label classification of tweets into their concerns towards vaccines.

Rationale/Explanation Extraction. Many research studies have voiced concerns about deep learning models being black-boxes with lack of transparency in the outputs they produce. Hence, there are many attempts towards developing methods that provide rationales behind the predictions made by such models [41, 45, 65, 69]. Some of the early and popular methods have been LIME [65] and SHAP [41] which can be used to provide explanations for any classifier. In the text domain, explanations are often in the form of spans of the input text given to the models. These have been extracted both using generative models [37] and discriminative models using attention weights [51] or by sequence labelling [96]. Generating explanations for image data has also been studied where certain objects in images are being identified as explanations [77, 95], including ones that have been adapted for COVID-19 diagnosis [85]. There exist some other techniques that are used to generate explanations for textual data. For instance, explanations can be generated by methods that perform keyword extraction [5, 11], and by multi-task models that perform classification and keyword extraction simultaneously [73]. There also exist methods that extract text spans by predicting their starting and ending indices [21]. Finally, Aspect-based Sentiment Analysis models [49, 88, 94] can be modified to perform classification while providing explanations.

There have been several studies on detecting depression and suicide risk from social media data, and due to the nature of the task, it is imperative that explanations be provided along with the classes to prevent mis-diagnosis [2, 32, 54, 98]. Moreover, fake news detection from social media is another domain where explanations help build trust in predictions from the deep learning models [18, 30, 40]. Finally, as discussed before, hate-speech detection also requires explanations to be given, and several works exist to accomplish this task of explainable classification [31, 42, 44, 87]. There exist a few datasets (for explainable classification) that provide human annotated explanations with text data [20, 44, 82, 97]. However, none of these datasets deal with a multi-label scenario. The CAVES dataset developed in our prior work [59] uniquely provides a human-annotated explanation for every label associated with an anti-vaccine tweet.

Question Answering (QA) Models. The task of QA deals with extracting an answer to a question from a given passage, and has garnered sufficient interest from the community; see [56] and [71] for surveys. The primary dataset that is the benchmark for QA models is the SQuAD dataset [63] which contains 100k questions curated from Wikipedia articles. This dataset has been analysed by some works [66], and an updated version, SQuAD 2.0, has also been released [62]. Methods of several paradigms have been explored for this task, including LSTM with pointer networks [78], hierarchical attention networks [79], and Transformers [10]. The task has also been explored in a multi-lingual setting by translating the SQuAD dataset to other languages such as Spanish [12] and Persian [1].

3 DATASETS AND TASK

We use two datasets containing social media posts with two different types of untrustworthy/ harmful information—(1) the *HateXplain* dataset [44] containing hate-speech posts (from Twitter and Gab), and (2) the *CAVES* dataset [59] consisting of anti-vaccine (anti-vax) tweets that express concerns towards COVID-19 vaccines. This section describes the datasets, and the task we tackle.

3.1 HateXplain Dataset

The HateXplain dataset (provided in [44]) contains posts from X (formerly known as Twitter) and Gab, that are categorized as 'hate-speech', 'offensive', or 'normal'.[5] *Hate-speech* is any speech that attacks certain individuals or groups based on their characteristics (such as race, religion, or gender), while *offensive-speech* is one that contains derogatory terms even though it is not directly attacking any individual/group [17, 23]. The *normal* class represents posts that do not belong to the above two categories. Each post in the dataset has been categorized into *exactly one* of these three classes. For each post, a rationale/explanation corresponding to the class label is given, explaining which part of the post led to it getting labeled as 'hate-speech' / 'offensive'. It must be noted that the 'normal' class has no marked explanation since there is no hateful/offensive content in these posts.

The dataset contains 19,229 posts that were labelled with a majority class by crowdsourced workers, with 30.9% posts being 'hate-speech' and 28.5% posts being 'offensive'. Some examples of each of these classes along with the rationale spans are given in Table 1. The dataset was split into 80% train, 10% validation, and 10% test sets.

3.2 CAVES Dataset

This dataset contains 9,921 anti-vaccine tweets labelled with specific concerns/objections that the user (author of the tweet) expresses against the use of COVID-19 vaccines (e.g., ineffectiveness, side effects, etc.) [59]. The dataset has 12 different classes, as detailed in Table 2, with 11 of them

[5]Dataset available at https://github.com/hate-alert/HateXplain.

Table 1. Examples of Tweets with their Labels and Explanations from the HateXplain Dataset

Excerpt of Post	**Label**
dad should have told the *muzrat whore to fuck off* and went in anyway	*hateful*
i am black with all white friends *taco man*	*offensive*
dont think im getting my baby them white 9 he has two white j and nikes not even touched	*normal*

The explanations for different labels are highlighted in *blue*. *Note: The examples contain offensive content due to the nature of the dataset.*

Table 2. The Different Classes/Labels (Concerns or Objections Towards Vaccines) in the CAVES Dataset [59] Along with their Descriptions and Distribution

Classes	**Description**	**%**
conspiracy	Belief in deeper conspiracies, not just money-making (e.g., vaccines are being used to track people, COVID is a hoax).	4.9%
country	Disapproval of the country where it was developed/manufactured.	2.0%
ineffective	Vaccines are not effective enough to prevent the disease.	16.9%
ingredients	Undesirable ingredients or technology used in the vaccines.	4.4%
mandatory	Vaccines should not be made mandatory.	7.9%
pharma	Big Pharmaceutical companies only care about money-making, or have a controversial history.	12.8%
political	Governments/politicians are pushing their own agenda though the vaccines.	6.3%
religious	Unwilling to get vaccinated due to religious reasons	0.6%
rushed	Vaccines have not been tested properly or that the published data is not accurate.	14.9%
side-effect	Side-effects of the vaccines, including deaths caused.	38.4%
unnecessary	Vaccines are unnecessary or alternate cures are better.	7.3%
none	No specific reason stated in the tweet or some reason different from the other given classes.	6.3%

Note that the percentages do NOT add up to 100% since a single tweet can have multiple concerns/objections.

representing actual objections/concerns, while the last one called *'None'* representing "no specific concern".[6] The distribution of classes is also given in the last column of Table 2.

Since a tweet can contain one or more anti-vax concerns, each tweet is labelled with single/multiple labels (minimum one, maximum three) expressing specific objections/concerns against the intake of COVID-19 vaccines. About 20.0% of the tweets in the dataset have more than one label whereas the remaining tweets have exactly one label. Additionally, for each of the labels associated with a tweet, the CAVES dataset contains a separate rationale in the form of a phrase/span appearing in the tweet-text. We have reported a few examples of tweets along with their labels and explanations in Table 3. Note that the *'None'* class is an exclusive class—it is not present in conjunction with any other classes, and tweets labeled 'None' have no marked explanations.

3.3 Tuple Prediction Task

In this work, we address the task of *extracting (label, rationale) tuples in a single/multi-label setting*. Given the joint task, our objective is twofold—(1) to identify single/multiple labels associated with

[6]Dataset available at https://github.com/sohampoddar26/caves-data.

Table 3. Examples of Tweets with Their Labels and Explanations, from the CAVES Dataset

Excerpt of Tweet	**Labels**
STOP TAKING *TOXIC VAX* and expose COVID hoax and murders with morphine and ventillators. **there is No covid**!	*ingredients*, conspiracy, **unnecessary**
Please don't push vaccine on us make it voluntary. We don't trust anything to do with *Bill Gates* pushing their agenda of **vaccine chips**!!	*pharma*, mandatory, **ingredients**
The reason insurance companies won't pay out if you experience the inevitable *adverse reactions, including death* is because it is an Experimental Vaccine	*side-effect*, rushed
Would you want the Russian vaccine? If not, you shouldn't want one that's been *pushed through for political reasons* either.	*political*, country
Catholic leaders are advising Catholics that the COVID-19 vaccine from Johnson & Johnson is 'morally compromised'	*religious*
I'm NOT taking your damn vaccine. Keep it out of my veins!	*none*

The explanations for different labels are highlighted in *blue*, red and **brown**.

the given text, and (2) to extract a span from the text, one for each predicted label, that explains the rationale behind the corresponding label prediction.

For the CAVES dataset, this task translates to identifying possibly multiple anti-vax concerns expressed in a Twitter post, together with their corresponding rationales in the form of extractive spans from the post's text. For the HateXplain dataset, we extract the hate-speech sentiment of the Twitter/Gab post along with its corresponding rationale. The idea of training suitable models to automatically extract labels jointly with their corresponding explanations in a *multi-label* setting makes the task more interesting as well as challenging, as was demonstrated in our prior work [59]. However, this task has been rarely explored in the literature.

4 PROPOSED METHODOLOGY

In this section, we describe our proposed methodology, where we map the (label, rationale) tuple-prediction task to a standard QA model. The final framework is called **MuLX-QA**. Different notations used in this section have been summarized in Table 4.

4.1 Overview of Our Approach

Different from prior works, we design a Question-Answering (QA)-based approach for the joint task of predicting multiple labels (for a given text) together with their corresponding rationale spans from the given text. Our intuition behind exploring a QA-approach is that the target rationale span can be thought of as an 'answer' to a 'question' about the corresponding class, such as "why is this text associated with the class $<class_k>$?" or "why is $<class_k>$ applicable to this text?".

Hence, we propose *MuLX-QA*, a novel Question-Answering framework trained with both positive and contrastive questions. Figure 1 shows a pictorial overview of our model. We start with a standard transformer-based extractive QA framework that can extract a phrase/span from the input text, given a question. More specifically, the model, as illustrated in Figure 1, consists of a transformer-based encoder *(RoBERTa)*, followed by two classification heads respectively determining the probability of each input text token to be (i) the start, and (ii) the end of the span that can provide an answer to the question asked. For a question of the type stated above, this answer span will be the rationale for the predicted label.

Table 4. List of Different Notations Used in Section 4

Notation	Description
	Symbols
$<text_i>$	The ith text input
$<class_k>$	The kth class
L	Number of classes
n	Number of tokens in the text
S_{ik}	Starting index of target span for $<text_i>$ and $<class_k>$
E_{ik}	Ending index of target span for $<text_i>$ and $<class_k>$
$\mathcal{N}$	Negative sample rate
$\mathcal{I}$	Final input to the model consisting of $<text_i>$, special tokens and a question
l	Number of tokens in input to the model $\mathcal{I}$
$\mathcal{E}$	Transformers based text encoder
d	Output embedding dimension of $\mathcal{E}$
$\mathcal{H}$	Contextualized embeddings for every token in $\mathcal{I}$
$\mathcal{H}_c$	Contextualized embeddings for first $n+2$ tokens in $\mathcal{I}$
W_{start}	Trainable weights for the span start classification layer
W_{end}	Trainable weights for the span end classification layer
P_{start}	Vector of probabilities for each token in the input to be span start
P_{end}	Vector of probabilities for each token in the input to be span end
n_{best}	Number of best tokens to consider from P_{start} and P_{end} for Algorithm 1
	Special Input Tokens
<s>	Start of sequence
</s>	End of sequence
<unk>	Unknown

Thus a standard QA model can help achieve our goal to find out the rationale behind the text being associated with a certain label. However, we want to simultaneously identify the label(s) as well, apart from extracting the rationale span(s) for each identified label(s). Accordingly, we propose ***a novel strategy to train such a framework with contrastive questions*** to determine the non-existence of certain labels. Specifically, we append a special <unk> token to the input text. The model is then trained to extract the <unk> token for labels not associated with the text while extracting a normal span for the labels present.

The rest of the section describes how we formulate the tuple prediction task as a QA-problem, and then explains our proposed framework and training/inference strategy in detail.

4.2 Formulating the Tuple Prediction Task as QA

Let $<text_i> = w_{i1}, w_{i2}, \ldots, w_{in}$ be the source text of length n, that we want to classify into L classes. We prepend the input text with a 'begin of sequence' (<s>) token, and append it with a special <unk> token at the end. The question that we ask in order to determine the presence/absence of a given class $<class_k>$ in the text has the following *generic template* "*Why* $<class_k>$*?*". We separate the modified input text and the question with a 'end of sequence' (</s>) token. Finally, the input given to the model, $\mathcal{I}$, takes the format—'`<s>` $<text_i>$ `<unk> </s> Why` $<class_k>$`? </s>`' .

Here $k \in [1, L]$, and <unk> represents *"unknown"*. Note that while our *question templates remain generic*, we use the dataset metadata to frame our questions for the two datasets as shown in Table 5. We try different variations of question prompts, which will be discussed later in Section 6.2.

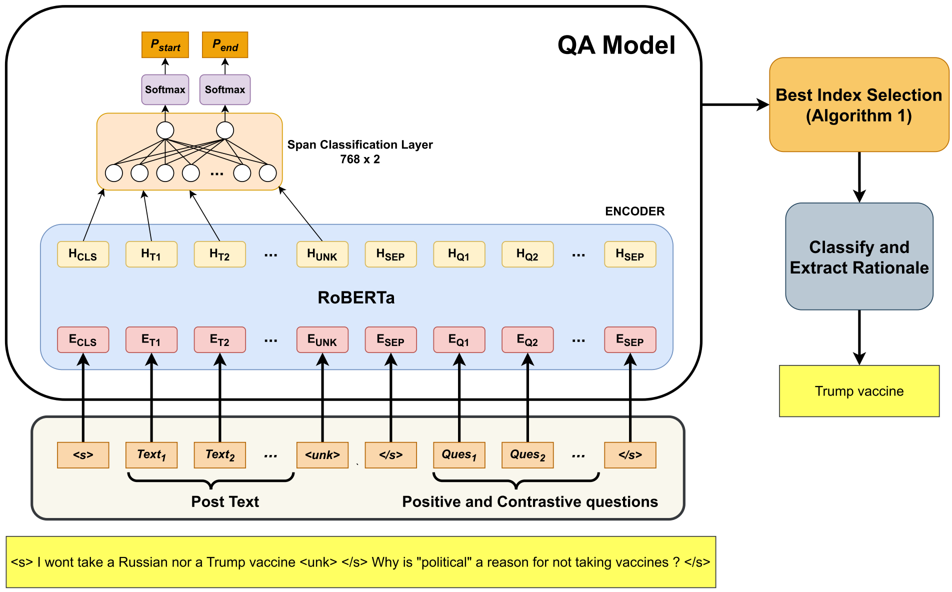


Fig. 1. The proposed MuLX-QA model. The tokens of the input text and the question (framed using the class labels) are fed into the RoBERTa encoder, which converts them into contextualized embeddings. Embeddings corresponding to the input text are then fed to the classification head on top to predict the token probabilities of being the start and end of the rationale span. Examples of inputs are given in Table 5.

Table 5. Examples of Inputs and Target Spans used to Train MuLX-QA for the Explainable Classification

Text & Ground truth label(s)	**Question**	**Target Span**
HateXplain Dataset (Here, $\mathcal{N}$ = 2)		
<s> if do not fuck with indie or house music idk *your a faggot* **<unk>** [Offensive]	Why is the text offensive ?	*your a faggot*
	Why is the text hateful ?	**<unk>**
	Why is the text normal ?	**<unk>**
CAVES Dataset (Here, $\mathcal{N}$ = 3)		
<s> I wont take a *Russian* nor a *Trump vaccine* **<unk>** [Political, Country]	Why is "political" a reason for not taking vaccines ?	*Trump vaccine*
	Why is "country" a reason for not taking vaccines ?	*Russian*
	Why is "side-effect" a reason for not taking vaccines ?	**<unk>**
	Why is "ineffective" a reason for not taking vaccines ?	**<unk>**
	Why is "none" a reason for not taking vaccines ?	**<unk>**

The positive questions are formed from the ground truth targets, while $\mathcal{N}$ contrastive/negative questions are randomly formed from those absent from the ground truth.

The model is trained to infer the context from the given tweet text $<text_i>$ and determine if $<class_k>$ is associated with the tweet by extracting a span (S_{ik}, E_{ik}) as the output, represented as a tuple of start (S_{ik}) and end (E_{ik}) indices with respect to $<text_i>$. The indices serve both as the *rationale span* and an of $<class_k>$ in $<text>$. Accordingly, (S_{ik}, E_{ik}) can take one of the following three combinations of values:

— $(n+1, n+1)$ representing the <unk> token, thereby signifying that $<class_k>$ *is absent.*
— (s, e), where $1 \leq s \leq e \leq n$ representing a valid explanation ranging from the sth to eth token in the text, thereby also signifying the *presence of* $<class_k>$.
— $(0, 0)$ representing the <s> token, if no explanation is present (e.g., for the *'None'* class in CAVES dataset or the *'Normal'* class in HateXplain).

Table 5 shows examples of positive and negative/contrastive questions and corresponding answers for a given text from each dataset.

4.3 Generating Inputs for Training and Testing

We now highlight the novelty of our training strategy that enables MuLX-QA to jointly extract labels together with their corresponding target spans, for a given text. For each text in the training and validation sets, first we construct *positive* questions corresponding to each ground truth label associated with the text. Here, the target answer spans are the corresponding ground truth rationales provided in the dataset.

Next, for the given text, we randomly sample $\mathcal{N}$ 'negative' classes among those that are *not* part of the ground truth label set (for the given text). These negative classes are used to form our *contrastive/negative* questions. For these questions, <unk> (representing *unknown*) is set as the target answer span, since the corresponding classes are not associated with the given text. While the model is trained to predict explanation spans, training with contrastive questions indirectly gives our model the ability to predict the labels as well (if the model predicts <unk> as the explanation, the corresponding label with which the question was formed/asked is not present). Table 5 shows examples of both positive and negative questions from the HateXplain and CAVES datasets.

Additionally, during training, for every tweet in the CAVES dataset that is *not* labeled with the exclusive "**none**" class (please note that "**none**" does not co-exist with any other class), we always include 'none' as one of the $\mathcal{N}$ classes for framing the negative/contrastive questions. We refer to the negative questions formed using the 'none' class as '*exclusive-class questions*', an example of which is shown in the last row for the second example (from the CAVES dataset) in Table 5.

Note that, in Table 5, we demonstrate the examples with $\mathcal{N} = 2$ for HateXplain and $\mathcal{N} = 3$ for the CAVES datasets, where $\mathcal{N}$ is the number of contrastive/negative questions. While we perform our experiments with different values of $\mathcal{N}$ (detailed later in Section 6.2), it was empirically set to 2 for HateXplain and 5 for CAVES for our final experiments.

During the test/inference phase, for each text in the test set, we frame one question for each of the L classes in the dataset (12 for CAVES and 3 for HateXplain) to determine their presence/absence with regards to the text. For a given question, we say that the corresponding class is present if the trained model does *not* predict the <unk> token. In such a case, the predicted answer span is considered as the rationale behind the existence of the corresponding label. The final output consists of all such pairs/tuples of (label, target span).

4.4 Proposed Model Architecture

Let us assume that the input sequence to the encoder, $\mathcal{I}$ consists of $l = n + m + 4$ tokens, where n is the number of tokens in the context (text to be classified), m is the number of tokens in the question, and the remaining four tokens consist of the <s>, the <unk>, and the two </s> tokens as described earlier. Our proposed framework, as illustrated in Figure 1, can be decomposed into two parts—a transformer-based encoder, followed by two classification heads performing the span extraction. Since *transformers* are extensively used in recent literature, the readers can refer to the original works [19, 76] for further details on the transformer architecture and pre-training strategies.

Let the encoder be a function E that converts the input sequence $\mathcal{I}$ into a sequence of contextualized vector embeddings $\mathcal{H} \in \mathbb{R}^{l \times d}$, where d is the output embedding dimension, and $\mathcal{H} = E(\mathcal{I})$. As discussed earlier in this section, the model is trained to extract an answer span by predicting its start and end tokens. This is done with the help of two feed-forward layers with trainable weight vectors $W_{start}, W_{end} \in \mathbb{R}^{d \times 1}$. Since the start and end token indices can only take values from 0 (representing the <s> token) to $n + 1$ (representing the <unk> token), we only consider a subset

ALGORITHM 1: Compute the best possible starting and ending indices for the answer, given the input sets of probability vectors.)

```
function getBestIndex(P_start, P_end)
    X_start ← argSortDescending(P_start)
    X_end ← argSortDescending(P_end)
    valid ← hashmap()
    for i_start ∈ X_start[1 : n_best] do
        for i_end ∈ X_end[1 : n_best] do
            if (i_start < i_end) then
                score = P_start[i_start] + P_end[i_end]
                valid[(i_start, i_end)] ← score
            end if
        end for
    end for
    return (i_start, i_end) ← argMax(valid)
end function
```

(the first $n + 1$ context vectors) of $\mathcal{H}$, and call it $\mathcal{H}_c \in \mathbb{R}^{(n+2)\times d}$. The normalized probabilities of the source text tokens to be the start and end of the rationale span, $P_{start}, P_{end} \in [0,1]^{(n+2)\times 1}$, are obtained as follows:

$$P_{start} = \mathtt{softmax}(\mathcal{H}_c \cdot W_{start})$$
$$P_{end} = \mathtt{softmax}(\mathcal{H}_c \cdot W_{end})$$

Here, the softmax function is defined on a vector Z as follows:

$$\mathtt{softmax}(z_i) = \frac{e^{z_i}}{\sum_{z_j \in Z} \ e^{z_j}}, \forall z_i \in Z.$$

During training, we calculate the cross-entropy losses between the predicted probabilities and the ground truth span indices gt_{start}, and gt_{end}, as follows:

$$\mathcal{L}_{start} = -\sum_{i=0}^{n+1} \mathbb{1}(gt_{start} \equiv i) \cdot \log P_{start}[i]$$
$$\mathcal{L}_{end} = -\sum_{i=0}^{n+1} \mathbb{1}(gt_{end} \equiv i) \cdot \log P_{end}[i]$$

where $\mathbb{1}(\cdot)$ represents an indicator function that returns 1 if the condition is true and 0 otherwise. The model is trained by optimizing the joint loss $\mathcal{L} = \mathcal{L}_{start} + \mathcal{L}_{end}$.

During inference, our objective is to obtain the predicted indices (pr_{start}, pr_{end}) that correspond to the extracted answer span. However, simply taking these to be the `argmax` of the probability vectors P_{start} and P_{end} to get the indices might lead to an issue when $pr_{start} > pr_{end}$. We therefore use an inference algorithm (detailed in Algorithm 1), where we consider the top n_{best} probabilities in the start and end probability vectors, and consider all their combinations to get the pair that yields the highest sum, while respecting the constraint mentioned above. Here, n_{best} is a hyper-parameter used to select only the top logits returned by `argSortDecending`. This was done to limit the search space for the best answer to reduce inference time. Thus, we check only up to n_{best}^2 pairs of indices, and not all $(n+1)^2$ possibilities. n_{best} was empirically set to 20 during experiments.

This section detailed the proposed MuLX-QA framework. In the next section, we compare the performance of MuLX-QA on the two considered datasets with that of several strong baselines.

5 EXPERIMENTS AND RESULTS

In this section, we discuss our experimental setup followed by a description of our baselines. We then compare the results of our proposed model against the baselines on the two datasets.

5.1 Experimental Setup

The CAVES dataset (9,921 tweets) was originally split by iterative stratified sampling into train (70%), validation (10%) and test (20%) sets. The HateXplain dataset (19,229 posts) was also originally split into train (80%), validation (10%) and test (10%) sets. We used these existing splits to train, validate and test all the models.

For training MuLX-QA, we leveraged transformers-based QA pipelines from the Huggingface library [84], and experimented with pre-trained BERT [19], and RoBERTa [39] as the encoders. We observed better results with RoBERTa, and hence used it as the encoder to train MuLX-QA.

We enforced a maximum sequence length of 128—any input texts longer than 128 tokens were truncated. During training, we used a batch size of 64 and a learning rate of 1e-5. We trained all our models for eight epochs and saved the instance which achieved the best Macro-F1 performance on the validation set. This saved model instance was used to evaluate the test set. The metrics used to evaluate our models on various tasks are described below.

5.2 Metrics for the Tuple Prediction Task

For the task of predicting (label, explanation) tuples, we calculate binary F1-scores by considering tuples as separate entities. For a given text, we consider a predicted label-explanation tuple to be a match (true positive) if and only if (i) the predicted label is present in the gold standard set of labels for the given text, and (ii) the predicted explanation has an **intersection-over-union (IOU)** overlap of at least 50% (IOU $\geq$ 0.5) with the corresponding gold standard explanation (similar to [20]). We calculate the IOU between the predicted and gold-standard explanations at the word-level, after removing punctuations and articles. More specifically, we consider the predicted and gold-standard explanations as bags/sets of words (after removing punctuations and articles) and then compute the union and intersection between these two bags/sets of words.

For a given text, we calculate the #predicted tuples, #gold-standard tuples, and #correct (matching) tuples, and then calculate the Precision (#correct / #predicted), Recall (#correct / #gold-standard), and the F1-scores (harmonic mean of precision and recall). We refer to these metrics as *Tuple-Pre*, *Tuple-Rec*, and *Tuple-F1,* respectively [59].

5.3 Baselines

To compare our proposed model, we consider several baselines, including encoder-only discriminative models as well as encoder-decoder-based generative models.

Encoder-Only Discriminative Models. We consider several methods that predict multiple labels for a given text along with their corresponding rationales, in the form of tuples. First, we try the **'*Rational Label*'** model provided by the authors of *HateXplain* [44]. It consists of a BERT-based encoder along with classification layers for predicting the class and its corresponding explanation from a given text. Though it can only predict a single label with a single rationale, we applied it to both datasets to examine its performance on an explainable multi-label dataset (*CAVES*). It contains a token classification layer which predicts if a token is part of the rationale or not. For training on the CAVES dataset, we converted each tweet with multiple labels into multiple data points with the same tweet-text but with a different label. During validation and testing, they were evaluated similar to the other models.

The ‘***Multi-task***’ model, introduced in our prior work [59], contains a shared COVID-Twitter-BERT-v2 [52] encoder (which is a BERT-Large encoder pre-trained on COVID-related tweets) with two classification layers on top to separately predict the labels and generate the rationale spans for each of the classes. The beginning-of-sequence token (`[CLS]`) embedding from the encoder output is fed to a multi-label classification layer (linear fully-connected network) to get the logits corresponding to each of the classes. This is followed by sigmoid operations on each of the logits to get the probability of each class being present. The classes with a probability score ≥ 0.5 are considered to be the predicted labels. For the explanations, the second token classification layer is trained using a sequence labelling approach separately for each of the classes. This is done by passing the token embeddings from the encoder output of all the words to a linear layer that predicts if that token is part of the rationale for each of the classes. Here, the goal is to determine which words in the text can be part of the rationale for the given label. Finally, the explanations corresponding to the predicted labels are considered.

We also try a variation of the ‘***Multi-task***’ model which includes a **Recurrent Neural Network (RNN)** layer, specifically a **Gated Recurrent Unit (GRU)**, before the rationale classification layer. The encoder output embedding for each token is fed to the GRU layer, whose outputs are then passed to the linear classification layer as before.

As another baseline, we use the modified ‘***ExPred***’ **[96]** model for the multi-label setting as given in our prior work [59]. This model is similar to the *Multi-task* model and is used to generate rationale spans, with label prediction modelled as an auxiliary task. The predicted explanations are then multiplied by the original encoder embeddings and are then fed into another multi-label classification layer to predict the final labels.

Encoder-Decoder Generative Models. Among the encoder-decoder **generative** models, we leverage two (suitably modified) models, one based on BART [34] and the other based on T5 [61].

We experiment with the ‘***Paraphrase***’ **[94]** model, which uses a T5 encoder-decoder trained to predict pairs of label-rationale tuples by generating a template-based output. The target output for a given data point is constructed according to the template “$<class_1>$ because $<rationale_1>$ $[SSEP] \cdots [SSEP]$ $<class_n>$ because $<rationale_n>$”. Given an input text, the T5 model is then trained to generate this template output in an auto-regressive manner.

Finally, the ‘***Unified-BART***’ **[88]** model consists of a BART encoder-decoder trained to predict possibly multiple pairs of labels and rationales using a generative framework. First, all the class labels are appended to the end of the input text. The target output for the given text is then formed by converting the associated labels and corresponding rationales into a sequence of triplets. Each triplet consists of three indices, the start and end token positions in the input text representing the rationale span, and the target label index corresponding to the position of the class label after the input text. The probabilities of the generated outputs are calculated using various equations involving the input token embeddings, the BART-encoder hidden embeddings, the class token list embeddings, and the BART-decoder output (please refer to [88] for more details). Finally, beam search is used to decode the probabilities into the target sequence of index triplets.

5.4 Comparative Results

In this section, we discuss the performance of our proposed model MuLX-QA on the test sets of the two datasets, in terms of Tuple-metrics defined above, and compare it with the baselines. Table 6 shows the results of all models on both the *HateXplain* dataset and the *CAVES* dataset.

Table 6. Comparison of Models on the Two Datasets

	HateXplain Dataset			*CAVES Dataset*		
Model	**T-Pre**	**T-Rec**	**T-F1**	**T-Pre**	**T-Rec**	**T-F1**
Encoder-Only Discriminative Baselines						
Rational Label [44] *(single label)*	0.1266	0.1212	0.1238	0.0642	0.0521	0.0576
ExPred [96]	0.1386	0.1185	0.1278	0.1944	0.1535	0.1716
Multi-Task [59]	0.1815	0.1757	0.1786	0.3952	0.3961	0.3957
Multi-Task (w/ GRU) [59]	0.2229	0.2199	0.2214	0.3304	0.3383	0.3343
Encoder-Decoder Generative Baselines						
Paraphrase (T5) [94]	<u>0.5322</u>	<u>0.5322</u>	<u>0.5322</u>	<u>0.4303</u>	0.4041	<u>0.4168</u>
Unified-BART [88]	0.5232	0.5210	0.5221	0.4132	<u>0.4187</u>	0.4159
Proposed model						
MuLX-QA ($\mathcal{N}$ = 2)	**0.5463**	**0.5852**	**0.5651**	0.3179	**0.5074**	0.3909
MuLX-QA ($\mathcal{N}$ = 5)	-	-	-	0.4175	0.4914	**0.4514**
MuLX-QA ($\mathcal{N}$ = 9)	-	-	-	**0.4506**	0.4438	0.4506
MuLX-QA with CT-BERT encoder ($\mathcal{N}(HateXplain)$ = 2, $\mathcal{N}_{(CAVES)}$ = 5)	0.5498	0.5884	0.5684	0.4616	0.5205	0.4893

MuLX-QA performs the best on both datasets. The best method in each column is highlighted in bold, and second best method is <u>underlined</u>. Note that the HateXplain dataset has only 3 classes, hence the number of negative/contrastive questions $\mathcal{N} < 3$. All results of MuLX-QA except the last row are with the RoBERTa encoder, while the last row shows results with the CT-BERT encoder that is specifically pre-trained on tweets related to COVID-19.

Results on the HateXplain Dataset [Single Label per post]. Among the encoder-only baselines the *Multi-task* model with GRU performs the best with a Tuple-F1 of 0.2214. The encoder-decoder generative models perform much better with the *Paraphrase* method achieving Tuple-F1 score of 0.5322. The *Paraphrase* model has slightly higher scores than *Unified-BART*. Our proposed model ***MuLX-QA*** performs the best on all three metrics even though it is an encoder-only discriminative model, with a Tuple-F1 of 0.5651 (**6.2% improvement over *Paraphrase***).

Results on the CAVES Dataset [Single/Multiple Labels per post]. Table 6 also compares the performance of different models on the CAVES dataset. The *Multi-Task* models perform quite well, especially the non-GRU version, with a Tuple-F1 of 0.3957. The *Rational Label* model performs poorly on the *CAVES* dataset since it can only predict a single label. The encoder-decoder models achieve better results with *Paraphrase* achieving a Tuple-F1 score of 0.4168. However, all the other models are outperformed by our encoder-only *MuLX-QA ($\mathcal{N}$ = 5)* with a Tuple-F1 score of 0.4514 (**8.3% improvement over *Paraphrase***). Note that it is possible to get a higher Tuple-precision score for MuLX-QA, if we increase the value of $\mathcal{N} \geq 6$ (further details in Section 6.2).

Statistical Significance Testing. The predictions of the proposed model MuLX-QA are statistically significantly better ($p < 0.05$) than those of the best-performing baselines for both datasets, as per *McNemar's chi-square test.* We chose this test to compare the tuples from two classifiers since the test is applicable on paired nominal data [72].

Further Improvement with the Use of Domain-Specific Encoders. All results of MuLX-QA in Table 6, apart from the last row, are with a standard RoBERTa encoder, as stated earlier in this section. However, MuLX-QA can achieve better performances with the use of domain-specific encoders suited to the domain of the datasets. To demonstrate this, we use COVID-Twitter-BERT-v2 [52]

Table 7. Results of Different Ablation Tests on the MuLX-QA Model

Model	Tuple-Pre	Tuple-Rec	Tuple-F1
Best Baseline (Paraphrase)	0.4303	0.4041	0.4168
Original MuLX-QA	**0.4175**	**0.4914**	**0.4514**
MuLX-QA without exclusive-class questions	0.4508	0.4269	0.4385
MuLX-QA without negative questions	0.0604	0.5883	0.1096

(abbreviated as CT-BERT) which is a BERT-Large encoder pre-trained on COVID-related tweets, as the encoder. We tested the performance of our model with the CT-BERT encoder on both the datasets, and the results are given in the last row of Table 6. We observe that using a domain-specific encoder can help improve scores on the CAVES dataset in all three metrics, with a best Tuple-F1 score of 0.4893. We also observe slightly better scores on the HateXplain dataset, since the CT-BERT encoder is twice as large as the RoBERTa encoder. While we achieve better scores for the task at hand with the use of domain-specific pre-trained encoders, in the next section, we analyze the performance of MuLX-QA with the originally used RoBERTa encoder, since this version of the model is directly comparable to our baselines. Further exploration on the effects of using domain-specific encoders is left as a future work.

6 ANALYSIS

In this section, we analyse the performance of our model MuLX-QA in various scenarios.

We also perform a qualitative analysis of the predictions made by our model on the CAVES dataset and compare it with that of the best baseline (*Paraphrase*).

6.1 Ablation Tests Over MuLX-QA

First, we perform an ablation study on two aspects of MuLX-QA—(i) removing the 'exclusive-class questions', and (ii) removing all the negative questions. We perform these experiments only on CAVES, since this is the only dataset on which we use 'exclusive-class questions' to train the model.

Table 7 shows the performance of the original MuLX-QA, and the same model after these modifications are done, on the CAVES validation set. For comparison, we also list the performance of the best baseline (*Paraphrase*). We observe that removing the 'exclusive-class questions' reduces the Tuple-recall while leading to some increase in the Tuple-precision, overall dropping the Tuple-F1 scores to 0.4385 (from 0.4514 of the original MuLX-QA). The performance is however still better than *Paraphrase*, which has a Tuple-F1 score of 0.4168. If we remove the negative questions (and hence the guidance that the model needs to predict the <unk> token for labels that should not be associated with a given text), MuLX-QA loses its ability to correctly predict labels. Now, it predicts a text span as rationale for questions asked with any of the labels. Consequently, it performs very poorly on the Tuple-Precision score.

6.2 Performance of MuLX-QA in Various Scenarios

In this section, we analyse the performance of our model in different scenarios—modifying the model architecture, varying $\mathcal{N}$, input prompts, and the training dataset size. We also examine the applicability of our model over non-English data.

Effect of $\mathcal{N}$: The negative sample rate ($\mathcal{N}$) is an important hyper-parameter of MuLX-QA, which determines how many negative/contrastive questions per tweet to be included in the set of input data points (described earlier in Section 4.3). In case of the HateXplain dataset, since there are only

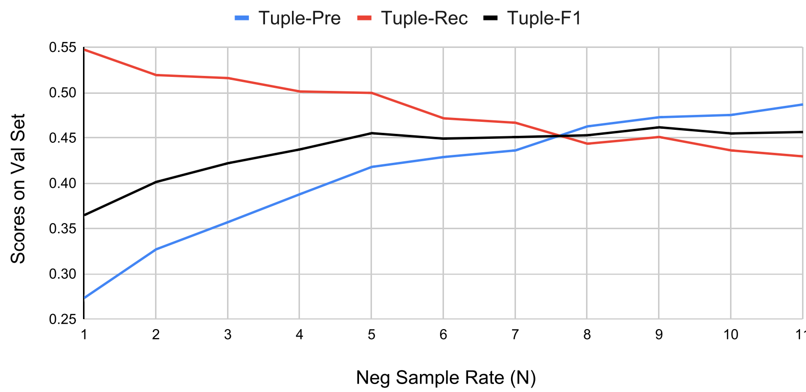


Fig. 2. Performance of MuLX-QA on varying $\mathcal{N}$ in terms of tuple-metrics on the validation set of CAVES.

three classes, $\mathcal{N}$ varies between 1 and 2 only and we get better results with $\mathcal{N} = 2$. However, in case of CAVES, $\mathcal{N}$ can vary between 1 and 11, and we observe some interesting trends.

We trained MuLX-QA on the training dataset for six epochs with different $\mathcal{N}$. We then plot the performance on the validation set of the CAVES dataset in Figure 2. We observe that on increasing the value of $\mathcal{N}$, the Tuple-Precision increases while the Tuple-Recall decreases steadily. This could be intuitively explained as that the model gets more precise with more negative examples being asked, as it becomes more aware of which labels should not be associated with a given tweet.

The Tuple-F1 rises steadily till $\mathcal{N} = 5$, after which it evens out and we see only minor change after that. The maximum score of 0.4617 (on the CAVES validation set) is obtained at $\mathcal{N} = 9$. However, it should be noted that increasing the value of $\mathcal{N}$ also increases the size of the training data, which in turn increases the training time linearly. To achieve a balance between the training time and the performance, we selected $\mathcal{N} = 5$ for reporting our final results on the CAVES dataset, for which the score on the validation set is 0.4552, but for which the model can be trained nearly twice as faster than when using $\mathcal{N} = 9$. We also notice that, on the *test set*, $\mathcal{N} = 9$ suffers a slight decrease in Tuple-F1 score (0.4506) than that with $\mathcal{N} = 5$ (0.4514). To summarize, we observe that increasing the value of $\mathcal{N}$ only increases performance up to a certain point, while also increasing the training time required by the model. Thus, it is better to increase the value only till there is a significant increase in performance.

Effect of Different Question Prompts. The use of generic and simple templates to frame our questions for training MuLX-QA *makes our soution generalizable* to other datasets/domains. The choice of words/prompts or the use of dataset metadata information (e.g., reasons for not taking vaccines or reasons for hate-speech) to fill the templates can however impact the overall model performance. To understand this effect, and to systematically obtain our best prompts, we compare, in Figure 3, the relative performance of MuLX-QA when trained with different types of questions. The different prompts used to frame these questions are also reported in Figure 3. Questions range from being very simple (using only the <label> information) to being more complex and longer (adding more natural language context or using dataset metadata information).

It is interesting to note that most of the prompts work well on both the datasets, with all corresponding model versions outperforming the best baseline model, *Paraphrase*. However, MuLX-QA trained with *'Original'* questions (following a generic template 'Why is the text <label>?' for HateXplain, and 'Why is <label> a reason for not taking vaccines?' for CAVES, as reported in Table 5), achieve marginally better scores. These questions, although being generic, are not only longer but also use the dataset metadata information (for CAVES) to give more context to the model to

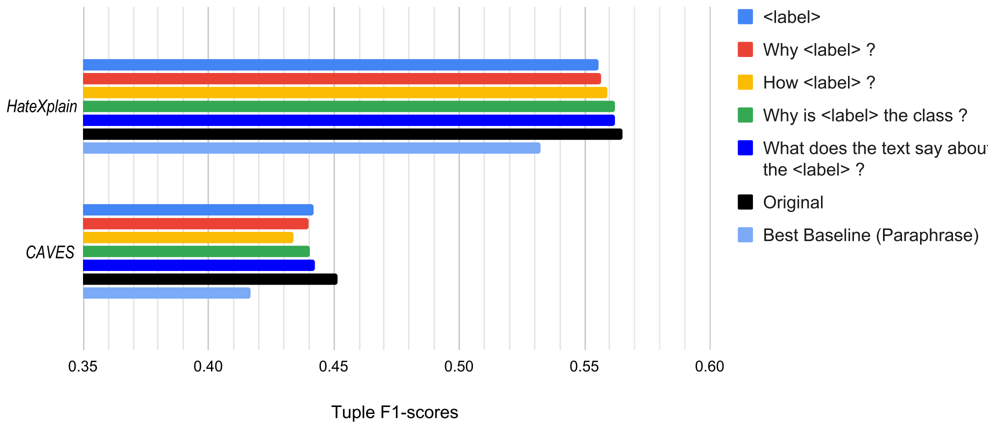


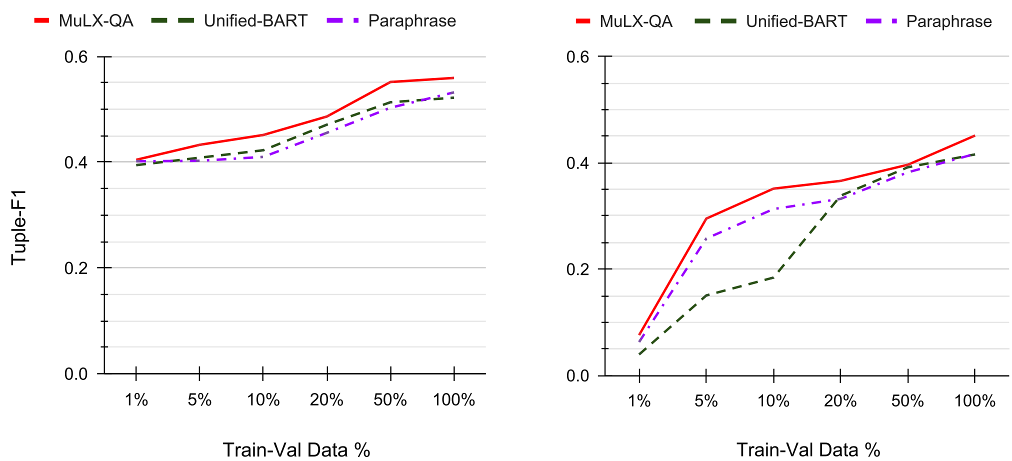
Fig. 3. Performance of different questions in terms of Tuple-F1 scores compared to the best baseline, *Paraphrase*. "Original" refers to the questions described earlier in Table 5.

condition its outputs on. At the same time, it is encouraging to observe that the scores with shorter prompts on both the datasets are almost at par with the best obtained results. Hence, while applying MuLX-QA on newer datasets, framing short questions just by using the class <label> (such as 'Why <label>?') is usually a good starting point. Further improvement may be achieved by incorporating more domain-specific signals in the questions.

Effect of Training Dataset Size. Finally, we also try to understand the effect of training data size on the performance of MuLX-QA, when compared to the encoder-decoder generative models—*Paraphrase* and *Unified-BART*. To this end, we trained these three models on various (randomly selected) fractions of the training (and validation) sets—1%, 5%, 10%, 20%, 50%, and 100%, and evaluated them on the *original test set data*. For each setting, the models were trained the same way as described previously (refer to Section 5.1). These experiments help us to understand how MuLX-QA fares against strong generative baselines in limited data settings.

The Tuple-F1 scores of the three models obtained on the two datasets have been plotted in Figure 4. MuLX-QA performs the best in all scenarios, which demonstrates the robustness of our model across varying sizes of available training data. The two encoder-decoder models perform mostly similarly. However, on CAVES, we observe that the *Unified-BART* model performs much worse than MuLX-QA and *Paraphrase* with 5% and 10% data sizes. This gap is however less pronounced on the HateXplain dataset. A possible explanation of this difference could be that the pre-training strategy of T5 (backbone of *Paraphrase*) is more suitable to handle text-to-text tasks; as a result, even with less data to train on, *Paraphrase* performs better than *Unified-BART*. Also, BART (backbone of *Unified-BART*) is pre-trained for sequence-to-sequence tasks. *Unified-BART*, trained to generate sequence of indices instead of sequence of words, seems to need more data to understand the mapping between tokens/class labels and their corresponding indices. Lastly, the number of labels is 12 in case of CAVES, whereas HateXplain has only 3 labels, which makes the task easier on the latter. Also, HateXplain has twice the amount of training data as the CAVES dataset.

Application on Other Languages. As discussed in Section 2, there is now an increasing amount of work on social media text in non-English languages. We therefore wanted to analyse the performance of our model on some language that is completely different from English. To this end, we select the ViHOS dataset [25] which contains texts in *Vietnamese* having two classes



(a) HateXplain (b) CAVES

Fig. 4. Tuple-F1 scores of models with limited training data. Scores are obtained on the original test sets.

Table 8. Performance of the Proposed Model and the Best Baseline Model on the “ViHOS” [25] Dataset on Hate-speech in Vietnamese Language

Model	Tuple-Pre	Tuple-Rec	Tuple-F1
Best Baseline (Paraphrase)	0.6447	0.6447	0.6447
MuLX-QA	0.7661	0.7523	0.7591

(hateful/not-hateful), with hateful spans being marked for the hateful class. This dataset contains about 11K samples split into train-validation-test subsets in the ratio 80:10:10. In about half the samples, there exist hateful spans. Though this dataset has only two possible classes depending on the presence/absence of spans, we can map it to our tuple-prediction task.

We applied MuLX-QA on the ViHOS dataset with two minor changes. First, we used the PhoBERT-base-v2 encoder [55] (instead of RoBERTa), which is a RoBERTa encoder pre-trained on Vietnamese texts. Second, we asked it the questions “Tai sao ghét ?” (‘Why hate?’ in Vietnamese) and “Tai sao không ghét ?” (‘Why not hate?’ in Vietnamese), respectively, for hateful and not hateful classes. We then trained the model as before, on the training set of ViHOS, and used the validation set to find the best checkpoint. Our model MuLX-QA achieved a Tuple-F1 score of 0.7591 on the test set of ViHOS, as shown in Table 8. For comparison, we also considered the Paraphrase model, which was the best baseline as per our previous analysis, using T5-vietnamese[7] (instead of T5). We trained it on the ViHOS train dataset and applied it on the ViHOS test dataset. Table 8 shows the results of the Paraphrase model as well. It is seen that our proposed model performs much better than the Paraphrase model on this dataset as well, which demonstrates the utitlity of our proposed MuLX-QA model for non-English data as well.

6.3 Qualitative Comparison of MuLX-QA with *Paraphrase*, our Strongest Baseline

In this section, we qualitatively analyse some predictions made by MuLX-QA and our strongest baseline, *Paraphrase*. We also compare their model complexities.

[7]https://huggingface.co/sangcamap/t5_vietnamese_qr

Table 9. Qualitative Comparison of Predictions Made by MuLX-QA and *Paraphrase*

#	Tweet Excerpt	Ground Truth Tuple	MuLX-QA Tuple	Paraphrase Tuple
HateXplain Dataset				
H1	bring out the uneducated hillbillies and white supremacists to our sign meeting duh i work oil duh i like truck duh me hate muslim duh i can not read	(normal, -)	(normal, -)	(hateful, bring out the uneducated hillbillies ··· duh me hate muslim duh i can not read)
H2	any wigger who calls white missionaries who save the non white children in africa india ect a race traitor is an actual h0m0	(offensive, wigger h0m0)	(offensive, wigger)	(offensive, wigger white missionaries who save the lives of non white children in africa)
CAVES Dataset				
C1	whistleblower reveals emails discussing covid vax we want to avoid having the info on the fetal cells floating out there	(ingredients, fetal cells)	(ingredients, info on the fetal cells)	(side-effect, Fetal Cells Floating Out There)
C2	you can always re infect with the same cold, there will never be a vaccine this thing touted by pfizer is probably a weak infection agent	(ineffective, re infect with the same cold)	(ineffective, pfizer is probably a weak infection agent)	(ineffective, never be a vaccine)

Manual Analysis of Predictions. As seen in the previous section, MuLX-QA is able to predict the correct labels and corresponding rationale spans (with respect to the gold standard annotations) in many cases, quantitatively outperforming strong baselines in the process. Now, we qualitatively analyze where our model is going wrong, and where *MuLX-QA* is performing better than *Paraphrase.*

Table 9 shows, for two test set tweets/posts from each dataset (H1, H2 from HateXplain, and C1, C2 from CAVES), the ground truth tuple (label and rationale), and the tuples predicted respectively by MuLX-QA and *Paraphrase.* First, we notice that in quite a few cases, the predicted rationale spans (by both models) are either too short or too long compared to the ground truth rationales. Hence, the *IOU* falls slightly below 0.5 (the minimum requirement to be declared as overlap according to our metrics, as described in Section 5.1), and hence the predicted rationales do *not* 'match' the ground truth according to the metrics. However, we observe that *Paraphrase* more frequently tends to extract much longer rationale spans than the ground truth, compared to MuLX-QA. H1 & H2 are two such examples. We also notice cases where *Paraphrase* completely misses the ground truth rationales, possibly because it concentrates more on discriminative words (such as 'hate muslims' in H1 and 'floating' in C1. As a result, it tends to predict wrong labels more often than MuLX-QA.

When we check tweets where both the models fail according to our tuple-metrics, MuLX-QA apparently outputs much more relevant spans. For example, for C2 in Table 9, even though the rationale span predicted by MuLX-QA does not match the ground truth rationale, the predicted rationale is still related to the 'ineffective' class, unlike the prediction of the *Paraphrase* model which extracts something completely irrelevant.

Complexity Analysis. As we saw earlier in Section 5.4, our proposed model MuLX-QA outperforms heavier encoder-decoder generative models, even though it is an encoder-only model. Specifically, ***Paraphrase*** uses T5-base encoder-decoder which has **220M** (million) parameters, and ***Unified-BART*** uses BART-large encoder-decoder which has **406M** parameters. In comparison, **MuLX-QA** uses a RoBERTa-base encoder which has only **125M** parameters, thus resulting in a much smaller memory footprint, and requiring less GPU-memory to train.

Next, we analyse the time-complexity of the models. If we feed a sequence of n tokens as input to a transformer encoder with hidden dimension d, the time complexity is $O(n^2d + nd^2)$ [76, 92]. Similarly, a transformer decoder running k auto-regressive steps (generating k tokens) has a time complexity of $O((n^2d+nd^2)\cdot k)$. For encoder-decoder models such as BART and T5, the overall time complexity amounts to $O((n^2d+nd^2)\cdot(1+k))$. For MuLX-QA, the QA-model has a time complexity of $O(n^2d + nd^2 + nd)$ which can be approximated as $O(n^2d + nd^2)$. However, we ask a question to the model $\mathcal{N}$ times, which brings the complexity to $O((n^2d + nd^2) \cdot \mathcal{N})$. Thus, our model usually takes less time to run if we use relatively low value of $\mathcal{N}$.

As a future work, we envision to explore better ways to reduce the inference time—such as using a separate lightweight classifier to predict potential classes with higher recall, followed by utilizing MuLX-QA to predict the answers only for those classes.

7 CONCLUSION

In this work, we focus on the task of explainable multi-label classification on two challenging datasets related to two different types of untrustworthy/harmful content prevalent in social media—hate speech, and vaccine misinformation. Our proposed Question-Answering model **MuLX-QA**, using simple and generic question prompts, outperforms several strong baselines, including state-of-the-art encoder-decoder generative models on both the datasets. The implementation of our model will be made publicly available upon acceptance of this article. As a future work, we propose to apply MuLX-QA in other domains (apart from social media posts) where explanations are important for classification. We also propose to use these models to gain insights about real world trends on social media in the respective domains.

ACKNOWLEDGMENTS

The authors acknowledge the anonymous reviewers whose comments greatly helped to improve the article.

REFERENCES

[1] Negin Abadani, Jamshid Mozafari, Afsaneh Fatemi, Mohammd Ali Nematbakhsh, and Arefeh Kazemi. 2021. ParSQuAD: Machine translated squad dataset for persian question answering. In *Proceedings of the 2021 7th International Conference on Web Research (ICWR).* IEEE, 163–168.

[2] V. Adarsh, P. Arun Kumar, V. Lavanya, and G. R. Gangadharan. 2023. Fair and explainable depression detection in social media. *Information Processing & Management* 60, 1 (2023), 103168.

[3] Alan Aipe, N. Mukuntha, Asif Ekbal, and Sadao Kurohashi. 2018. Deep learning approach towards multi-label classification of crisis related tweets. In *Proceedings of the 15th ISCRAM Conference.*

[4] Fatimah Alkomah and Xiaogang Ma. 2022. A literature review of textual hate speech detection methods and datasets. *Information* 13, 6 (2022), 273.

[5] Rabah Alzaidy, Cornelia Caragea, and C. Lee Giles. 2019. Bi-LSTM-CRF sequence labeling for keyphrase extraction from scholarly documents. In *The Web Conf erence* 2551–2557.

[6] Iqra Ameer, Noman Ashraf, Grigori Sidorov, and Helena Gómez Adorno. 2020. Multi-label emotion classification using content-based features in twitter. *Computación y Sistemas* 24, 3 (2020), 1159–1164.
[7] Ferdaous Benrouba and Rachid Boudour. 2023. Emotional sentiment analysis of social media content for mental health safety. *Social Network Analysis and Mining* 13, 1 (2023), 17.
[8] Erika Bonnevie, Allison Gallegos-Jeffrey, Jaclyn Goldbarg, Brian Byrd, and Joseph Smyser. 2020. Quantifying the rise of vaccine opposition on twitter during the COVID-19 pandemic. *Journal of Communication in Healthcare* (2020), 1–8.
[9] Talha Burki. 2020. The online anti-vaccine movement in the age of COVID-19. *The Lancet Digital Health* 2, 10 (2020), e504–e505.
[10] Sabur Butt, Noman Ashraf, Muhammad Hammad Fahim Siddiqui, Grigori Sidorov, and Alexander Gelbukh. 2021. Transformer-based extractive social media question answering on TweetQA. *Computación y Sistemas* 25, 1 (2021), 23–32.
[11] Ricardo Campos, Vítor Mangaravite, Arian Pasquali, Alípio Mário Jorge, Célia Nunes, and Adam Jatowt. 2018. Yake! collection-independent automatic keyword extractor. In *Proceedings of the European Conference on Information Retrieval*. Springer, 806–810.
[12] Casimiro Pio Carrino, Marta R. Costa-jussà, and José A.R. Fonollosa. 2019. Automatic spanish translation of the squad dataset for multilingual question answering. *arXiv preprint arXiv:1912.05200* (2019).
[13] Bharathi Raja Chakravarthi, Vigneshwaran Muralidaran, Ruba Priyadharshini, and John P. McCrae. 2020. Corpus creation for sentiment analysis in code-mixed Tamil-English text. *arXiv preprint arXiv:2006.00206* (2020).
[14] Hanjie Chen, Guangtao Zheng, and Yangfeng Ji. 2020. Generating hierarchical explanations on text classification via feature interaction detection. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Online, 5578–5593. https://doi.org/10.18653/v1/2020.acl-main.494
[15] Tianshui Chen, Muxin Xu, Xiaolu Hui, Hefeng Wu, and Liang Lin. 2019. Learning semantic-specific graph representation for multi-label image recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 522–531.
[16] Zhao-Min Chen, Xiu-Shen Wei, Peng Wang, and Yanwen Guo. 2019. Multi-label image recognition with graph convolutional networks. In *Proceedings of the IEEE/CVF on CVPR*. 5177–5186.
[17] Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. In *Proceedings of the International AAAI Conference on Web and Social Media*, Vol. 11. 512–515.
[18] Giorgio De Magistris, Samuele Russo, Paolo Roma, Janusz T. Starczewski, and Christian Napoli. 2022. An explainable fake news detector based on named entity recognition and stance classification applied to COVID-19. *Information* 13, 3 (2022), 137.
[19] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).
[20] Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C. Wallace. 2020. ERASER: A benchmark to evaluate rationalized NLP models. In *Proceedings of the 58th Annual Meeting of the ACL*. 4443–4458.
[21] George-Andrei Dima, Dumitru-Clementin Cercel, and Mihai Dascalu. 2021. Transformer-based multi-task learning for adverse effect mention analysis in tweets. In *Proceedings of the 6th Social Media Mining for Health (# SMM4H) Workshop and Shared Task*. 44–51.
[22] Angel Fiallos and Karina Jimenes. 2019. Using Reddit data for multi-label text classification of Twitter users interests. In *Proceedings of the 2019 6th International Conference on eDemocracy & eGovernment (ICEDEG)*. IEEE, 324–327.
[23] Antigoni-Maria Founta, Constantinos Djouvas, Despoina Chatzakou, Ilias Leontiadis, Jeremy Blackburn, Gianluca Stringhini, Athena Vakali, Michael Sirivianos, and Nicolas Kourtellis. 2018. Large scale crowdsourcing and characterization of Twitter abusive behavior. In *Proceedings of the 12th International AAAI Conference on Web and Social Media*.
[24] Soumitra Ghosh, Amit Priyankar, Asif Ekbal, and Pushpak Bhattacharyya. 2023. Multitasking of sentiment detection and emotion recognition in code-mixed hinglish data. *Knowledge-Based Systems* 260 (2023), 110182.
[25] Phu Gia Hoang, Canh Duc Luu, Khanh Quoc Tran, Kiet Van Nguyen, and Ngan Luu-Thuy Nguyen. 2023. ViHOS: Hate speech spans detection for Vietnamese. *arXiv preprint arXiv:2301.10186* (2023).
[26] Azhar Hussain, Syed Ali, Madiha Ahmed, and Sheharyar Hussain. 2018. The anti-vaccination movement: A regression in modern medicine. *Cureus* 10, 7 (2018).
[27] Muhammad Okky Ibrohim and Indra Budi. 2019. Multi-label hate speech and abusive language detection in Indonesian Twitter. In *Proceedings of the 3rd Workshop on Abusive Language Online*. 46–57.

[28] Sarthak Jain and Byron C. Wallace. 2019. Attention is not explanation. 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)*. Association for Computational Linguistics, Minneapolis, Minnesota, 3543–3556. https://doi.org/10.18653/v1/N19-1357

[29] Neil F. Johnson, Nicolas Velásquez, Nicholas Johnson Restrepo, Rhys Leahy, Nicholas Gabriel, Sara El Oud, Minzhang Zheng, Pedro Manrique, Stefan Wuchty, and Yonatan Lupu. 2020. The online competition between pro-and anti-vaccination views. *Nature* (2020), 1–4.

[30] Gargi Joshi, Ananya Srivastava, Bhargav Yagnik, Mohammed Hasan, Zainuddin Saiyed, Lubna A. Gabralla, Ajith Abraham, Rahee Walambe, and Ketan Kotecha. 2023. Explainable misinformation detection across multiple social media platforms. *IEEE Access* 11 (2023), 23634–23646.

[31] Ayushi Kohli and V. Susheela Devi. 2023. Explainable offensive language classifier. In *Proceedings of the International Conference on Neural Information Processing*. Springer, 299–313.

[32] Abhinav Kumar, Jyoti Kumari, and Jiesth Pradhan. 2023. Explainable deep learning for mental health detection from English and Arabic social media posts. *ACM Transactions on Asian and Low-Resource Language Information Processing* (2023).

[33] Chung-Wei Lee, Wei Fang, Chih-Kuan Yeh, and Yu-Chiang Frank Wang. 2018. Multi-label zero-shot learning with structured knowledge graphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1576–1585.

[34] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 7871–7880.

[35] Xiaoya Li, Jingrong Feng, Yuxian Meng, Qinghong Han, Fei Wu, and Jiwei Li. 2020. A unified MRC framework for named entity recognition. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 5849–5859.

[36] Andy T. Liu, Wei Xiao, Henghui Zhu, Dejiao Zhang, Shang-Wen Li, and Andrew Arnold. 2022. QaNER: Prompting question answering models for few-shot named entity recognition. *arXiv preprint arXiv:2203.01543* (2022).

[37] Hui Liu, Qingyu Yin, and William Yang Wang. 2018. Towards explainable NLP: A generative explanation framework for text classification. *arXiv preprint arXiv:1811.00196* (2018).

[38] Weiwei Liu, Haobo Wang, Xiaobo Shen, and Ivor Tsang. 2021. The emerging trends of multi-label learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2021).

[39] 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. *arXiv preprint arXiv:1907.11692* (2019).

[40] Yi-Ju Lu and Cheng-Te Li. 2020. GCAN: Graph-aware co-attention networks for explainable fake news detection on social media. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 505–514.

[41] Scott M. Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems* 30 (2017).

[42] Aditya Mahajan, Divyank Shah, and Gibraan Jafar. 2021. Explainable AI approach towards toxic comment classification. In *Emerging Technologies in Data Mining and Information Security: Proceedings of IEMIS 2020, Volume 2*. Springer, 849–858.

[43] Binny Mathew, Punyajoy Saha, Hardik Tharad, Subham Rajgaria, Prajwal Singhania, Suman Kalyan Maity, Pawan Goyal, and Animesh Mukherjee. 2019. Thou shalt not hate: Countering online hate speech. In *Proceedings of the International AAAI Conference on Web and Social Media*, Vol. 13. 369–380.

[44] Binny Mathew, Punyajoy Saha, Seid Muhie Yimam, Chris Biemann, Pawan Goyal, and Animesh Mukherjee. 2020. Hatexplain: A benchmark dataset for explainable hate speech detection. *arXiv preprint arXiv:2012.10289* (2020).

[45] Sherin Mary Mathews. 2019. Explainable artificial intelligence applications in NLP, biomedical, and malware classification: A literature review. In *Intelligent Computing - Proceedings of the Computing Conference*. Springer, 1269–1292.

[46] Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. 2018. The natural language decathlon: Multitask learning as question answering. *arXiv preprint arXiv:1806.08730* (2018).

[47] Saif Mohammad, Felipe Bravo-Marquez, Mohammad Salameh, and Svetlana Kiritchenko. 2018. Semeval-2018 task 1: Affect in tweets. In *Proceedings of the 12th International Workshop on Semantic Evaluation*. 1–17.

[48] Rajdeep Mukherjee, Atharva Naik, Sriyash Poddar, Soham Dasgupta, and Niloy Ganguly. 2021. Understanding the role of affect dimensions in detecting emotions from tweets: A multi-task approach. In *Proceedings of the 44th International ACM SIGIR Conference*. 2303–2307.

[49] Rajdeep Mukherjee, Tapas Nayak, Yash Butala, Sourangshu Bhattacharya, and Pawan Goyal. 2021. PASTE: A tagging-free decoding framework using pointer networks for aspect sentiment triplet extraction. In *Proceedings of the 2021 Conference on EMNLP*. 9279–9291.

[50] Rajdeep Mukherjee, Uppada Vishnu, Hari Chandana Peruri, Sourangshu Bhattacharya, Koustav Rudra, Pawan Goyal, and Niloy Ganguly. 2022. MTLTS: A multi-task framework to obtain trustworthy summaries from crisis-related microblogs. In *Proceedings of the 15th ACM International Conference on Web Search and Data Mining* (Virtual Event) *(WSDM'22)*. ACM, New York, , 755–763. https://doi.org/10.1145/3488560.3498536

[51] James Mullenbach, Sarah Wiegreffe, Jon Duke, Jimeng Sun, and Jacob Eisenstein. 2018. Explainable prediction of medical codes from clinical text. *arXiv preprint arXiv:1802.05695* (2018).

[52] Martin Müller, Marcel Salathé, and Per E. Kummervold. 2020. COVID-Twitter-BERT: A natural language processing model to analyse COVID-19 content on Twitter. *arXiv preprint arXiv:2005.07503* (2020).

[53] Sharan Narang, Colin Raffel, Katherine Lee, Adam Roberts, Noah Fiedel, and Karishma Malkan. 2020. WT5?! Training Text-to-Text Models to Explain Their Predictions. https://doi.org/10.48550/ARXIV.2004.14546

[54] Usman Naseem, Matloob Khushi, Jinman Kim, and Adam G. Dunn. 2022. Hybrid text representation for explainable suicide risk identification on social media. *IEEE Transactions on Computational Social Systems* (2022).

[55] Dat Quoc Nguyen and Anh Tuan Nguyen. 2020. PhoBERT: Pre-trained language models for Vietnamese. In *Findings of the Association for Computational Linguistics (EMNLP 2020)*. 1037–1042.

[56] Devshree Patel, Param Raval, Ratnam Parikh, and Yesha Shastri. 2020. Comparative study of machine learning models and BERT on SQuAD. *arXiv preprint arXiv:2005.11313* (2020).

[57] María Antonia Paz, Julio Montero-Díaz, and Alicia Moreno-Delgado. 2020. Hate speech: A systematized review. *Sage Open* 10, 4 (2020), 2158244020973022.

[58] Soham Poddar, Mainack Mondal, Janardan Misra, Niloy Ganguly, and Saptarshi Ghosh. 2022. Winds of change: Impact of COVID-19 on vaccine-related opinions of Twitter users. In *Proceedings of the 16th ICWSM'22*.

[59] Soham Poddar, Azlaan Mustafa Samad, Rajdeep Mukherjee, Niloy Ganguly, and Saptarshi Ghosh. 2022. CAVES: A dataset to facilitate explainable classification and summarization of concerns towards COVID vaccines. In *Proceedings of the 45th International ACM SIGIR Conference*.

[60] Kashyap Popat, Subhabrata Mukherjee, Andrew Yates, and Gerhard Weikum. 2018. DeClarE: Debunking fake news and false claims using evidence-aware deep learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. 22–32.

[61] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research* 21, 1 (2020), 5485–5551.

[62] Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. 784–789.

[63] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. 2383–2392.

[64] Jishnu Ray Chowdhury, Cornelia Caragea, and Doina Caragea. 2020. Cross-lingual disaster-related multi-label tweet classification with manifold mixup. In *Proceedings of the 58th Annual Meeting of the ACL: Student Research Workshop*.

[65] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. " Why should I trust you?" Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1135–1144.

[66] Marc-Antoine Rondeau and Timothy J. Hazen. 2018. Systematic error analysis of the Stanford question answering dataset. In *Proceedings of the Workshop on Machine Reading for Question Answering*. 12–20.

[67] Pradeep Kumar Roy, Snehaan Bhawal, and Chinnaudayar Navaneethakrishnan Subalalitha. 2022. Hate speech and offensive language detection in Dravidian languages using deep ensemble framework. *Computer Speech & Language* 75 (2022), 101386.

[68] Tulika Saha, Apoorva Upadhyaya, Sriparna Saha, and Pushpak Bhattacharyya. 2021. A multitask multimodal ensemble model for sentiment-and emotion-aided tweet act classification. *IEEE Transactions on Computational Social Systems* (2021).

[69] Wojciech Samek, Thomas Wiegand, and Klaus-Robert Müller. 2017. Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models. *arXiv preprint arXiv:1708.08296* (2017).

[70] Shashi Shekhar, Hitendra Garg, Rohit Agrawal, Shivendra Shivani, and Bhisham Sharma. 2023. Hatred and trolling detection transliteration framework using hierarchical LSTM in code-mixed social media text. *Complex & Intelligent Systems* 9, 3 (2023), 2813–2826.

[71] Marco Antonio Calijorne Soares and Fernando Silva Parreiras. 2020. A literature review on question answering techniques, paradigms and systems. *Journal of King Saud University - Computer and Information Sciences* 32, 6 (2020), 635–646.

[72] Joshua Henrina Sundjaja, Rijen Shrestha, and Kewal Krishan. 2022. McNemar and Mann-Whitney U tests. In *StatPearls [Internet]*. StatPearls Publishing.

[73] Matthew Tang, Priyanka Gandhi, Md Ahsanul Kabir, Christopher Zou, Jordyn Blakey, and Xiao Luo. 2019. Progress notes classification and keyword extraction using attention-based deep learning models with BERT. *arXiv preprint arXiv:1910.05786* (2019).

[74] Adane Nega Tarekegn, Mario Giacobini, and Krzysztof Michalak. 2021. A review of methods for imbalanced multi-label classification. *Pattern Recognition* 118 (2021), 107965.

[75] Grigorios Tsoumakas and Ioannis Katakis. 2007. Multi-label classification: An overview. *International Journal of Data Warehousing and Mining (IJDWM)* 3, 3 (2007), 1–13.

[76] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in Neural Information Processing Systems* 30 (2017).

[77] Tom Vermeire, Dieter Brughmans, Sofie Goethals, Raphael Mazzine Barbossa de Oliveira, and David Martens. 2022. Explainable image classification with evidence counterfactual. *Pattern Analysis and Applications* (2022), 1–21.

[78] Shuohang Wang and Jing Jiang. 2016. Machine comprehension using match-lstm and answer pointer. *arXiv preprint arXiv:1608.07905* (2016).

[79] Wei Wang, Ming Yan, and Chen Wu. 2018. Multi-granularity hierarchical attention fusion networks for reading comprehension and question answering. In *Proceedings of the 56th Annual Meeting of the ACL*. 1705–1714.

[80] Yuxi Wang, Martin McKee, Aleksandra Torbica, and David Stuckler. 2019. Systematic literature review on the spread of health-related misinformation on social media. *Social Science & Medicine* 240 (2019), 112552.

[81] Jônatas Wehrmann, Ricardo Cerri, and Rodrigo C. Barros. 2018. Hierarchical multi-label classification networks. In *Proceedings of the International Conference on Machine Learning*. PMLR, 5075–5084.

[82] Sarah Wiegreffe and Ana Marasović. 2021. Teach me to explain: A review of datasets for explainable natural language processing. *arXiv preprint arXiv:2102.12060* (2021).

[83] Sarah Wiegreffe and Yuval Pinter. 2019. Attention is not not explanation. 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)*. Association for Computational Linguistics, Hong Kong, China, 11–20. https://doi.org/10.18653/v1/D19-1002

[84] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of EMNLP: System Demonstrations*.

[85] Yu-Huan Wu, Shang-Hua Gao, Jie Mei, Jun Xu, Deng-Ping Fan, Rong-Guo Zhang, and Ming-Ming Cheng. 2021. JCS: An explainable COVID-19 diagnosis system by joint classification and segmentation. *IEEE Transactions on Image Processing* 30 (2021), 3113–3126.

[86] Weizhi Xu, Junfei Wu, Qiang Liu, Shu Wu, and Liang Wang. 2022. Evidence-aware fake news detection with graph neural networks. In *Proceedings of the ACM Web Conference 2022*. 2501–2510.

[87] Sargam Yadav, Abhishek Kaushik, and Kevin McDaid. 2023. Hate speech is not free speech: Explainable machine learning for hate speech detection in code-mixed languages. In *Proceedings of the 2023 IEEE International Symposium on Technology and Society (ISTAS)*. IEEE, 1–8.

[88] Hang Yan, Junqi Dai, Tuo Ji, Xipeng Qiu, and Zheng Zhang. 2021. A unified generative framework for aspect-based sentiment analysis. In *Proceedings of the 59th Annual Meeting of the ACL and the 11th IJCNLP (Volume 1: Long Papers)*. 2416–2429.

[89] Pengcheng Yang, Xu Sun, Wei Li, Shuming Ma, Wei Wu, and Houfeng Wang. 2018. SGM: Sequence generation model for multi-label classification. *arXiv preprint arXiv:1806.04822* (2018).

[90] Mesay Gemeda Yigezu, Olga Kolesnikova, Grigori Sidorov, and Alexander Gelbukh. 2023. Transformer-based hate speech detection for multi-class and multi-label classification. In *Proceedings of IberLEF 2023 Workshop*.

[91] Jin Yuan, Shikai Chen, Yao Zhang, Zhongchao Shi, Xin Geng, Jianping Fan, and Yong Rui. 2023. Graph attention transformer network for multi-label image classification. *ACM Transactions on Multimedia Computing, Communications and Applications* 19, 4 (2023), 1–16.

[92] Biao Zhang, Deyi Xiong, and Jinsong Su. 2018. Accelerating neural transformer via an average attention network. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 1789–1798.

[93] Min-Ling Zhang and Zhi-Hua Zhou. 2013. A review on multi-label learning algorithms. *IEEE Transactions on Knowledge and Data Engineering* 26, 8 (2013), 1819–1837.

[94] Wenxuan Zhang, Yang Deng, Xin Li, Yifei Yuan, Lidong Bing, and Wai Lam. 2021. Aspect sentiment quad prediction as paraphrase generation. In *Proceedings of the 2021 Conference on EMLNP*. 9209–9219.

[95] Xiaoge Zhang, Felix T. S. Chan, and Sankaran Mahadevan. 2022. Explainable machine learning in image classification models: An uncertainty quantification perspective. *Knowledge-Based Systems* 243 (2022), 108418.

[96] Zijian Zhang, Koustav Rudra, and Avishek Anand. 2021. Explain and predict, and then predict again. In *Proceedings of the 14th ACM International Conference on WSDM*. 418–426.

[97] Linda Zhou, Andrew Caines, Ildiko Pete, and Alice Hutchings. 2023. Automated hate speech detection and span extraction in underground hacking and extremist forums. *Natural Language Engineering* 29, 5 (2023), 1247–1274.

[98] Hamad Zogan, Imran Razzak, Xianzhi Wang, Shoaib Jameel, and Guandong Xu. 2022. Explainable depression detection with multi-aspect features using a hybrid deep learning model on social media. *World Wide Web* 25, 1 (2022), 281–304.

Received 16 July 2023; revised 8 January 2024; accepted 15 March 2024