Exploration of Contrastive Learning Strategies toward more Robust Stance Detection Systems

by

Udhaya Kumar Rajendran

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

Department of Computer Science
Faculty of Science and Environmental Studies
Lakehead University

Abstract

Stance Detection, in general, is the task of identifying the author's position on controversial topics. In Natural Language Processing, Stance Detection extracts the author's attitude from the text written toward an issue to determine whether the author supports the issue or is against the issue. The studies analyzing public opinion on social media, especially in relation to political and social concerns, heavily rely on Stance Detection. The linguistics of social media texts and articles are often unstructured. Hence, the Stance Detection systems needed to be robust when identifying the position or stance of an author on a topic. This thesis seeks to contribute to the ongoing research on Stance Detection. This research proposes a Contrastive Learning approach to achieve the goal of learning sentence representations leading to more robust Stance Detection systems. Further, this thesis explores the possibility of extending the proposed methodology to detect stances from unlabeled or unannotated data. The stance of an author towards a topic can be implicit (through reasoning) or explicit; The proposed method learns the sentence representations in a contrastive fashion to learn the sentence-level meaning. The Contrastive Learning of sentence representations results in bringing similar examples in the Sentence Representation space belonging to the same stance close to each other, whereas the dissimilar examples are far apart. The proposed method also accommodates the token-level meaning by combining the Masked Language Modeling objective (similar to BERT pretraining) with the Contrastive Learning objective. The performance of the proposed models outperforms the baseline model (a pretrained model finetuned directly on the stance datasets). Moreover, the proposed models are more robust to the different adversarial perturbations in the test data compared to the baseline model. Further, to learn sentence representations from the unlabeled dataset, a clustering algorithm is used to partition the examples into two groups to provide pseudo-labels for the examples to use in the Contrastive Learning framework. The model trained with the proposed methodology on pseudo-labeled data is still robust and achieves similar performances to the model trained with the labeled data. Further analysis of the results suggests that the proposed methodology performs better than the baseline model for the smaller-sized and imbalanced (class ratio) datasets.

Acknowledgements

I would like to express my warmest thanks and sincere gratitude to my research supervisor Dr. Amine Trabelsi. I am extremely grateful for the opportunity he has offered me. His vision, sincerity, and motivation have inspired me to battle through the most challenging days during my study and research. It was a great privilege to work and study under his guidance.

I am incredibly grateful to my parents for their love, prayers, and sacrifices for educating me and letting me live my ’Rags to the Riches’ dream. I extend my special thanks to my brother Jagadeesan and my sister-in-law Sujitha for constantly believing in me and being my cheerleaders throughout this journey. I also want to thank my friends for their unwavering support and motivation during my study and research tenure. I must thank my friend Madhumitha who impacted and inspired me throughout this journey

Finally, I thank God, the Almighty, for blessing me with the strength and courage to complete my study and research successfully.

Table of Contents

List of Tables

List of Figures

Abbreviations

ALBERT A Lite BERT.

ARC Argument Reasoning Corpus.

BERT Bidirectional Encoder Representations from Transformers.

BOW Bag Of Words.

CNN Convolutional Neural Network.

FNC Fake News Challenge.

GLUE General Language Understanding Evaluation.

GPU Graphical Processing Unit.

MDL Multi Dataset Learning.

MLM Masked Language Modelling.

MLP Multilayer Perceptron.

NLP Natural Language Processing.

NSP Next Sentence Prediction.

RoBERTa Robustly Optimized BERT Pretraining Approach.

SDL Single Dataset Learning.

SVM Support Vector Machine.

Chapter 1

Introduction

1.1 Overview

Stance Detection is an emerging and active research area in Natural Language Processing. It is a sub-task of opinion mining that identifies the stance of the author of the text toward arguable topics. Opinion mining is an umbrella term that covers the extraction and analysis of opinions and emotions expressed in the text toward the topic of discussion in blog posts, online reviews, and debate forums. Social media, especially debate forums and opinion polls, are flooded with opinions and discussions on several controversial topics. A controversial topic divides people into two groups with different views (support/against) on the topic of discussion. Some popular, controversial topics include the Legalization of Abortion, Concern about Climate Change, Gay Marriage, Obama, the Legalization of Marijuana, Feminism, Atheism, etc. The term Stance Detection is often confused with Sentiment Analysis; Sentiment Analysis focuses only on the text's emotional polarity (positive or negative). For example, the customer review or feedback for a product indicates whether the product is good or bad, and the Sentiment Analysis system considers the words in the review or feedback to predict if the given feedback is positive or negative. However, Stance Detection focuses on the author's stance and detects if the author is in favor of the topic or against the topic of discussion. For example, the author's text 'Fetus is not human', on the topic 'Legalization of Abortion', indicates that the author is supporting the

Topic: Legalization of Abortion
Example: A fetus has rights too! Make your voice heard Stance: AGAINST
Example: Fetus is not human Stance: SUPPORT
Example (with errors in the text): A fetus has rights to! Make your voice heard Stance: AGAINST
Example (words replaced with their synonyms): A bunch of cells is not human Stance: SUPPORT

Table 1.1: Examples of opposing views on Legalization of Abortion topic

Legalization of Abortion. It is not necessary that when the author's stance is in favor of the topic of discussion, the sentiment of the text is positive. The piece of text just indicates the stance of the author on a given topic.

If we carefully notice the text 'Fetus is not human', the topic of discussion, 'Legalization of Abortion', is nowhere in the text. This indicates that for the Stance Detection task, the author of the text can convey their stance explicitly or infer the stance implicitly without having the topic of discussion mentioned in the text. The author's text may or may not contain the discussion topic, making the Stance Detection task more complicated. Also, spelling errors, missing words, repetition of words, and other commonly occurring errors in the text should be accommodated when predicting the author's stance. In this thesis, we aim to improve the automated Stance Detection from a given piece of text on controversial topics and to make the Stance Detection system more robust to the adversarial perturbations by accommodating the errors in the text when detecting the stance. Table 1.1 presents examples of opposing stances regarding the topic of the Legalization of Abortion.

1.2 Objective

The advancement in the machine learning area with the advent of neural networks and deep learning advanced the field of Natural Language Processing [1]. The main objective of this thesis is to learn robust representations of sentences expressed in social media discussions regarding contentious issues. By robust representations, we mean representations that will lead to more robust models to changes and variations in input data. The representations will be used in the Stance Detection task. Furthermore, we attempt to learn the representations of sentences from the stance examples, which are not labeled or annotated. We primarily concentrate on binary stances (e.g., support/against) social media English texts, such as tweets, news comments, and discussion forums.

We used the Contrastive Learning approach to construct more robust sentence representations for the Stance Detection task. Given an example (let's say anchor), the Contrastive Learning technique identifies a similar and dissimilar example. It keeps the similar example closer to the anchor example and drives the dissimilar example away from the anchor example in the representation space. We build similar (positive) and dissimilar (negative) examples for Contrastive Learning by considering the stance label of the examples. The examples with the same stance labels are similar, and the examples with different stance labels are dissimilar. We mainly explored different strategies for building positive and negative examples for an anchor example to learn the sentence representations in a contrastive fashion. We select the strategies to allow the Contrastive Learning framework to learn from 1) random positive and negative examples for an anchor example and 2) selected positive and negative examples for an anchor example; the selection is based on whether the example is closer or far from the anchor example. This strategy aims to identify and learn from the examples that are hard to distinguish in the representation space from the anchor example.

We see that Stance Detection identifies the author's stance toward a topic. We have already seen a stance example for the topic 'Legalization of Abortion'. There are widespread issues like 'Climate Change is a real concern', 'Atheism', 'Feminism', etc., where people on social media are divided into groups and hold a stance discussing the issue topics. The datasets we identified for our experiments contain multiple topics; for example, the dataset SemEval2016 [2] has five different topics such as 'Abortion', 'Atheism', 'Climate Change', 'Feminism', and 'Hillary Clinton'. We draw another dimension in our experimentation setups by 1) learning the representations from all the examples of all the topics in a dataset and 2) learning the representations for stances for each topic (e.g., considering the topic 'Abortion'as a dataset for learning representations). This arrangement intends to explore and experiment with the behavior of different strategies on positive and negative examples selection for an anchor example.

Given a sentence in a specific issue (e.g., 'fetus is not human', on legalization of abortion), our objective is two-fold. First, we will produce representations of sentences that will lead to trained models less sensitive to perturbations, hence improving stance-related tasks. Second, we will explore building the representations assuming having access to labeled as well as unlabeled examples indicating the stance of the sentence. The annotation-independence approach will accelerate the construction of the representations for different domains involving stance content (e.g., Covid-19 vaccination). Moreover, it will not rely on domain-specific features or labels; hence it will be highly applicable to emergent issues.

1.3 Key Concepts

Below we present the key concepts used in this thesis to learn robust sentence representations for the Stance Detection tasks.

1. **Text Sentence Representation Learning with Contrastive Pairs**

   A sequence of text contributes to the sentence. Learning the semantics of the entire sentence becomes essential in the Stance Detection task. We use a Contrastive Learning approach [3] to achieve our goal of learning sentence representations. As discussed in Section 1.2, the Contrastive Learning technique segregates the similar and dissimilar examples, i.e., the examples belonging to different stances are separated apart in the representation space.

2. **Combining Contrastive Learning Approach with Word Representation Learning**
   The word representations learned based on the surrounding words gives the system the ability to learn the context of the word. The combination of sentence representation learning and word representation learning is effective in making the system understand the semantics of the sentence and the context of the words in the sentence simultaneously.

1.4 Contributions

Below we present our contributions based on the key concepts explained in Section 1.3. Our code is publicly available in the Github repository [1].

1. We develop an approach of using a Contrastive Learning framework with different positive and negative pairs building strategies (explained in detail in Section 3.2) to learn more robust sentence representations to use in the Stance Detection task. To the best of our knowledge, this work is the first to employ a Contrastive Learning framework to learn sentence representations for the Stance Detection task.

2. We create a pipeline framework to learn sentence representations from unannotated text sentence examples using the Contrastive Learning approach. We first cluster the unannotated examples into two groups of contrastive stances

[1] https://github.com/rajendranu4/stance-detection

and then use the generated stance labels in the Contrastive Learning framework to learn representations for the examples.

1.5 Outline

Chapter 2 of this work contains the literature review of the closely related work in Stance Detection, Contrastive Learning, and Robustness of Natural Language Processing systems. Chapter 3 introduces and explains the methodology and the framework used to learn sentence representations from annotated and unannotated text sentence examples for the Stance Detection tasks. Also, the methodology behind the clustering of stance examples is explained in Chapter 3. Chapter 4 lists and explains all the datasets used in this work. The different experimental setups and the experiments carried out in this work are explained in Chapter 4. Chapter 5 reports the results of all the experiments carried out, as explained in Chapter 4, with different experiment setups, such as experiments with or without annotations and mixed or individual topics, etc. We conclude the thesis in Chapter 6 and mention the challenges that are not tackled in the presented work and which can be explored in the future.

Chapter 2

Related Work

2.1 Background

Stance Detection is the task of identifying from the text if the author is in 'favor of', or 'against', or 'neutral', towards a target topic. Detecting an author's stance with a given piece of text against a specific topic becomes essential in downstream applications like fake news detection, claim validation or argument search. Sentiment Analysis of a post considers the words used in the text sentences to classify the post's sentiment toward a discussion. However, in the Stance Detection problem, the stance of the author with the post is targeted toward a topic or an entity; hence, considering only the sentiment of the words will not help in predicting the author's stance on the related topic.

2.1.1 Language Models

The relationship between the words in the sentence of a Stance Detection task is crucial; hence, keeping track of it becomes essential. The transformers make this possible by having the attention mechanisms [4] to keep a record of the relations between words of a text sentence both in the forward and backward direction. The architecture of the transformers includes an encoder-decoder structure where the encoder maps the input sentence to a sequence of continuous representation, and the decoder decodes the output of the encoder to generate an output sequence. A representation is a vector of real values intended to encode the semantics of a sequence of words. Retaining the meaning of a word relative to the surrounding words will help better predict the next word for a given set of words in sequence. This task of determining the probability of a sequence of words to be followed, given a sequence of words, is called Language Modelling. This probabilistic model will help predict which word will more probably appear next, given a sequence of words.

The machine learning algorithm is provided with data to identify patterns and learn optimal values for all the relevant attributes. The process of acquiring knowledge and identifying specific patterns from the training data is called model training. Similar to humans, machine learning models can be made to reuse their old knowledge and transfer it to learn and comprehend the new knowledge that can be applied to a variety of new tasks. The process of training a model to build knowledge that can be used in other tasks is called pretraining. The process of retraining the pretrained model for a dedicated task is called finetuning. The transformer-based language representation models such as BERT (Bidirectional Encoder Representations from Transformers) [5] is pretrained with a vast corpus BooksCorpus (800M words) [6] and English Wikipedia (2,500M words). The particularity of BERT over the other language models is that BERT is trained bidirectionally, i.e., taking into account the order of words in the sequence from left to right and from right to left. The goal is to have a more profound sense of the context of the language. During the pretraining of the BERT model, two strategies are used to have better contextual learning. The first strategy is Masked Language Modelling (MLM) masks a percentage of words in a sentence and allows the model to predict the masked words given the context of the surrounding words. The second strategy is at the sentence level, Next Sentence Prediction (NSP). Given a pair of sentences, NSP predicts if the second sentence is the sentence that follows the first sentence in the original document. The pretrained BERT over a huge corpus with the MLM and NSP objective can be finetuned for a range of language tasks such as Classification tasks like Sentiment Analysis, Question-Answering tasks, Named

Entity Recognition, etc.

There were variants introduced in BERT, such as DistilBERT [7], ALBERT [8], and RoBERTa [9], which can be distinguished from BERT in terms of the size of the architecture and the methods used during pretraining. The BERT architecture has 12 layers and 110M parameters for the base version and 24 layers and 340M parameters for the large version. DistilBERT is the distilled version of BERT with six layers and 66M parameters. RoBERTa (Robustly Optimized BERT Pretraining Approach) was developed by implementing a dynamic masking method to enhance the training phase. With the dynamic masking strategy, the input sequence is duplicated ten times to have ten different ways of masking in contrast to BERT, which has masking done only once during the data preprocessing stage. The architecture of the RoBERTa model is even more complex than BERT, with 125M and 355M parameters for the base and large versions, respectively.

2.1.2 Contrastive Learning

The choice and quality of the data representation, or features, in the data used to train a machine learning system directly impact its performance. The process of learning a parametric mapping from a raw input data domain to a feature vector or tensor in the hopes of capturing and extracting more abstract and valuable notions that can increase performance on a variety of downstream tasks is referred to as representation learning. Contrastive representation learning can be thought of as learning through comparison. In contrast to the discriminative model, which learns assignments to some (pseudo) labels, and the generative model, which reconstructs the input sample, Contrastive Learning learns the representation by comparing the input samples. In Contrastive Learning, learning is carried out by comparing different samples instead of learning the signals individually from individual data samples. In the case of Natural Language Processing, comparisons are made between positive pairs of 'semantically similar' inputs and negative pairs of 'semantically different' inputs. The goal of Contrastive Learning is straightforward. Representations of 'similar' patterns need to be mapped closer to each other, while representations of 'different' patterns need to be farther away in the embedded space (a vector space representing the input samples). Therefore, by contrasting the positive and negative pair samples, the positive pair representation is pulled together, and the negative pair representation is pushed far away.

Selection of the best positive and negative pair for an anchor is crucial for Contrastive Learning. The intuition of Contrastive Learning is that the positive sample for an anchor is semantically similar and pulled towards the anchor. In contrast, the negative sample is pushed away from the anchor. The Contrastive Learning framework learns from the examples that are hard to distinguish in the representation space from the anchor example when we choose semantically similar examples or examples belonging to the same class of an anchor but far away from the anchor sentence in the embedding space [10]. These examples are known as hard positives + as mentioned in Figure 2.1. The hard negatives – for an anchor are the examples that do not belong to the same class of an anchor or are semantically dissimilar to an anchor but are closer to the anchor sample in the embedding space. Table 2.1 illustrates the selection of positive and negative examples for an anchor example for Contrastive Learning. The examples in the Table 2.1 are taken from the DebateForum dataset (see Table 4.1).

The following sections describe the early work carried out in Stance Detection and Contrastive Learning.

2.2 Stance Detection

The stance of the author can be predicted either with the text that the author has posted towards the topic or by the network connections and the metadata of the author, such as the author's connections within the social medium, the author's reactions that include likes and dislikes over a text post, etc. Hence the approach for

	Topic: Legalization of Abortion	
Category	**Example**	**Stance**
Anchor	Killing an unborn baby is murder it is still a life why does it make a difference if the baby is born or unborn it is still a baby and a life.	Against
Positive	Murder is wrong. Abortion is murder. Therefore abortion is wrong.	Against
Negative	Yea abortion should be legal!!!!!!! Even though i do not like the thought of innocent babies being killed for no reason at all. I would much rather see that happen then having children being starved and abused and neglected!	Support

Table 2.1: Selection of Positive and Negative examples for an Anchor example for Contrastive Learning

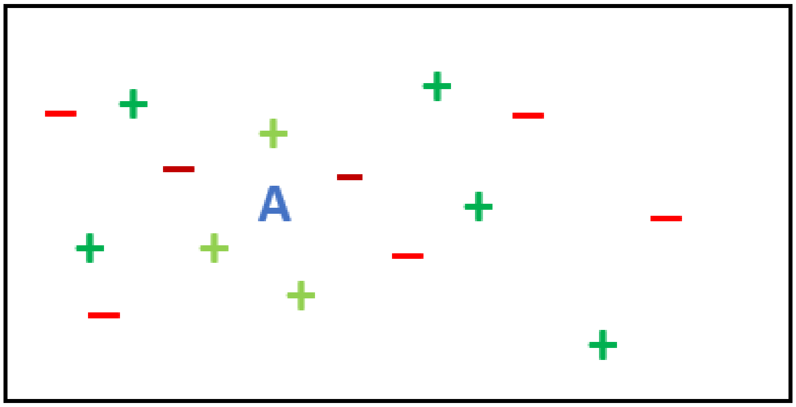


Figure 2.1: Illustration of Easy Positive +, Hard Positive +, Easy Negative – and Hard Negative – samples for an Anchor sample **A** in the embedding space.

the Stance Detection problem is sub-categorized into text-level and user-level, as the text can determine the author's stance that they post as well as the author's interactions online. Most of the works carried out in the Stance Detection area are based on supervised learning approaches, where the machine learning model is trained with the data having labeled stances.

2.2.1 Supervised Approach - Text Level

SemEval has introduced a dataset with 4870 English tweets [11] for stance towards six commonly known targets in the United States for the Stance Detection task. The dataset focuses on five different targets such as 'Atheism', 'Climate Change is a Real Concern', 'Feminist Movement', 'Hillary Clinton', and 'Legalization of Abortion'. The teams that participated in the competition used different classifiers with unigrams, n-grams, or n-grams-comb models for the Stance Detection task. The results of the competition show that the difference in the class distribution in the dataset plays a vital role in the model's performance. The instances for most of the targets are biased towards the label 'against'; hence, the F_{avg} score for the majority class is very high. Also, many teams in the competition tried introducing noisy labels in the dataset, which improved the model's performance.

A bidirectional Recurrent Neural Network [12] is used for the Stance Detection task where the input tweet is mapped to a vector on the encoder side, and stance labels for the corresponding input are generated on the decoder side.

Darwish et al. [13] used the attention-based encoder-decoder, which focuses on the different words of the input text for each target in the multi-target stance classification task. The authors [21] have taken the fact that the stance targets, for example, different brands or candidates in the elections, are closely related. They proposed a novel method of having a dynamic memory-augmented network to capture and store the information related to the stance of the related targets. For detecting stances towards 'n' targets which are treated as 'n' tasks, a Multi-tasking Learning framework is used, which improves the generalization of all functions by jointly learning them. Since this task deals with multi-target Stance Detection, every stance word is generated conditionally on its previous stance words. Still, for the decoder, the order of the targets should be specified in advance.

Logistic Regression classifier [14] is used for the Stance Detection task [15] on the novel dataset on rumored claims and its associated news articles introduced. The features are extracted from the articles using the Bag-of-Words (BOW) approach. Along with the BOW representation, an additional feature is extracted from the text, which indicates if the text is negated according to the parser. Finally, the cosine similarity between the claim and the headline's vector representation is calculated using word2vec.

Message-Level Transformer with Masked Document Representations (MeLT) [16] understands the context of a user's message in the social media using the sequence of their previous messages. Word level language model is used to process the user's messages, and from the individual messages, words are aggregated, ordered, and then masked. Finally, the reconstruction loss is adjusted with the predicted masked vector in the finetuning process. The experimental result shows an increase in performance when the previous messages are concatenated to the current message during the stance prediction task. Also, the proposed model MeLT can be used separately to get the context of the user's message.

Hardalov et al. [17] explored the possibility of cross-domain learning with 16 different datasets from four different sources, namely debates, news, social media, and others. This paper used an end-to-end unsupervised framework for out-of-domain prediction of unseen, user-defined labels. For out-of-domain predictions, the unseen targets are computed based on the label name similarity (soft mapping), or the labels are grouped. The nearest neighbors are identified to choose the most similar label (weak mapping). The domain-specific and global representation of the input sentence is passed through the Label embedding layer to obtain probability distributions. The experimental results show that the proposed model outperforms the baseline models in 9 out of 16 datasets, and weak mapping achieves the highest F1 average for the out-of-domain experiment.

Joseph et al. [18] compares an individual's self-reported stance with the inferred stance from social media to check whether the human annotations capture stance and what Stance Detection truly measures. The experiment results show how past behaviors influence the nature of an individual's stance.

Abstract Rational Stance JOINT (ARSJOINT) [19] is introduced to tackle the error propagation problem among the three modules; Abstract summarization, Rational, and Stance Detection. It addresses the issue of missing sharing valuable information between the modules. Now, focusing on the Stance Detection part, representation for a sentence is computed by the module using a Hierarchical Attention Network. Then the stance labels are computed on the sentence level attention layer and an MLP with softmax, which gives the logits of the stance labels. The results are compared with the Paragraph-Joint model, where the proposed model pretrained with ROBERTa performs better than the Paragraph-Joint model.

Jayaram et al. [20] tested the faithfulness of a model's prediction by introducing attention weights for the words in the text. To impart human-like rationalization, a small part of the training data is annotated using crowd-sourced annotations. An additional loss term is included with the standard loss term for the task, which allows the model to produce attributions for each example that are very similar to oracle attributions. The results suggest that attribution prior as well as the attention weights improve the model's rationales.

The model's reliability is tested by having a negative version of the original Perspective in the training dataset, i.e., for some of the original perspectives, a negative version of the same Perspective is generated by some methods to include as part of the training. The model (TRIplet Bert-based Inconsistency Detection) [21] works with three inputs, Claim, Perspective, and Negated version of the Perspective, and the latent representations for all these inputs are obtained from BERT, which is concatenated to a single representation. The final dense layer with two outputs provides the probabilities for supporting and opposing stances for a perspective. The result also supports that including negative perspectives with the input improves the performance and allows the model to filter out doubtful predictions.

2.2.2 Supervised Approach - User Level

Instead of text level approach for Stance Detection, Darwish et al. [13] use the interaction elements for any user, such as mentioned hashtags and retweeted accounts, to identify the similarities between the users and to cluster the similar users in a user similarity feature space. Also, the authors have used an attention mechanism to focus on the input text when generating stance labels. The unigrams and bigrams are feature representations from the texts for the SVM classifier. The results show that considering the interaction elements of any user in transforming the text feature space into user similarity feature space has improved performance, but the classification features considered here are not generalizable beyond the test sets used for this experiment.

Aldayel and Magdy [22] again use the user's interactions, network, and preferences on social media, which are termed social signals. It is believed that even when users do not post anything online, their interactions and preferences could influence their stance on any topic. Word n-grams are used as the features for the SVM classifier to identify a user's stance. The experimental results support the claim of the authors that the interactions or network activity of a user contain enough social signals to identify the stance of the tweet posted by that user.

Rashed et al. [23] project and cluster users to identify if they are polarized on a specific topic. The embeddings of users tweets on a topic are created with Google's Multilingual Universal Sentence Encoder (MUSE) with pre-trained CNN embeddings. This method of projecting the embedding representations of user tweets into lower dimensional space and clustering users allows for identifying the polarization between groups on any given topic.

2.2.3 Unsupervised Approach - User Level

Darwish et al. [20] proposed an unsupervised framework for the Stance Detection task where the Twitter users are first projected to low dimensional space using dimensionality reduction methods and then cluster the users to find the representative users of different stances. The members of homophilous groups are similar-minded users who are inclined to share similar views on specific topics. Once the users are clustered, human analysts are allowed to label each cluster based on the typical characteristics of the users. Here the cost of labeling the clusters is lower than labeling each user. The similarity between user pairs on retweets and hashtags is identified using the cosine similarity measure to form clusters. The experiment is carried out with the unlabeled datasets called 'immigration and gun control', 'the benefits/dangers of vaccines', and 'controversial remarks by Representative Ilhan Omar on the Israeli lobby'. The experiment results show that the average cluster impurity for the user in the unlabeled dataset is 98%, with an average recall of 86.5%.

2.2.4 Multi-Target Stance Detection

Sobhani et al. [12] introduced a dataset for multi-target Stance Detection where all 4455 tweets are manually annotated for stance towards more than one target simultaneously. The model with independent classifiers for every target predicts the stance of the first target and uses this prediction as an extra feature for predicting the stances of the subsequent targets with different independent classifiers. A bidirectional Recurrent Neural Network is used on the encoder side to map the input to a vector, and stance labels for the corresponding input are generated on the decoder side. It is assumed that the words in the tweet or text outside the context window have no influence on the target stance for that tweet.

Wei et al. [24] show that the generalization of data is improved when the model is jointly trained with multi-targets as the multi-target learning implicitly augments training data. Since this is multi-target learning, each stance word is generated conditionally on its previous stance words concerning the other targets. However, the decoder needs the specification of the order of targets in advance.

Li et al. [25] capture significant similarities with the help of multi-dataset and multi-target learning as the model is trained with different datasets and learns more universal representations for targets. This model uses the method of adaptive knowledge distillation (student-teacher model) where the sum of cross entropy loss between the predictions of student and hard labels and the distance loss between the predictions of student and teacher is minimized. Also, to classify the samples which are more representative of the label class than others with more confidence than the ambiguous ones, the samples with more considerable confidence from the teacher predictions receive less amount of temperature scaling in the loss function for knowledge distillation. The experimental results show that the multi-target and multi-dataset models outperform the Ad-hoc models. Also, the multi-dataset model outperforms the current state-of-the-art models on the Multi-Target (MT) [12] and SemEval [26] stance datasets.

Schiller et al. [27] introduced the Stance Detection benchmark, where the ML model is allowed to learn from ten different datasets in multi-dataset learning (MDL) environment. The experiment is carried out in 2 different fashions, Single dataset learning (SDL) and Multi-dataset learning (MDL) with pre-trained BERT weights or pre-trained weights of BERT finetuned on all GLUE datasets (Transfer Learning) known as MTDNN weights. Hence there will be four different models (BERT-Single Dataset Learning, BERT-Multi dataset learning, MTDNN-Single Dataset Learning, and MTDNN-MDL) for the experiment. The experiment result shows that the multi-dataset setup achieves better performance for 7 out of 10 datasets compared to the state-of-the-art results. Still, both the SDL and MDL failed to cross the state-of-the-art performance for SemEval-2019 Task 7 dataset [28]. The test dataset introduces three adversarial effects, paraphrasing, spelling errors, and negation, to identify the robustness of the MDL models. Surprisingly, the relative performance drop is higher for MDL models compared to SDL models. The resilience (the measure of the robustness of a model against all the adversarial attacks) of MDL and SDL are almost similar.

2.3 Stance Detection Datasets

This section provides a list of publicly available datasets utilized for the Stance Detection task.

Conforti et al. [29] introduced the largest available expert annotated dataset for stance classification tasks with over 51284 tweets in English. It has two different domains, Healthcare and Entertainment, with stance labels 'support', 'refute', 'comment', and 'unrelated'.

Ferreira and Vlachos [15] introduced a novel dataset called Emergent, which contains over 300 rumored claims and 2,595 associated news articles. The associated news articles are summarized and labeled with the stance labels of 'for', 'against', and 'observing'. The class distribution of these stance labels is 47.7%, 15.2%, and 37.1% for 'for', 'against', and 'observing', respectively.

Mohammad et al. [11], Matero et al. [16] and Aldayel and Magdy [22] used the tweets dataset from the SemEval Stance Detection task 2016. Mohammad et al. [11] and Matero et al. [16] used the dataset at the text level to analyze the tweet text for Stance Detection. In contrast, Aldayel and Magdy [22] used the dataset at the user level to identify the features, including the user's interaction, network, and preference for stance classification.

Sobhani et al. [12] introduced a dataset for multi-target Stance Detection, which contains over 4400 tweets for three different target pairs of Clinton-Sanders, Clinton-Trump, and Cruz-Trump related to the 2016 US election. The stance labels are 'for', 'against', and 'neutral'for every pair of presidential candidates.

Baheti et al. [30] used the Amazon Mechanical Turk crowd annotated TOXICHAT dataset, which has 2000 Reddit threads labeled with offensive language and stance. [13] used the same Amazon Mechanical Turk crowd-annotated workers to annotate the twitter data on the targets Donald Trump, COVID-related lockdowns, face masks, and COVID-19 vaccines

Hardalov et al. [17] experimented on the debate posts with over 4500 posts in four domains as Abortion (ABO), Gay Rights (GAY), Obama (OBA), and Marijuana (MAR). These targets have two stance labels, namely 'for' and 'against'.

Jayaram and Allaway [20] used the benchmark dataset VAST, posts from New York Times for the experiments. These posts are assigned stance labels 'for', 'against', or 'neutral' using crowd-sourcing annotations to impart human-like rationalization to a Stance Detection model.

Zhang et al. [19] worked on the benchmark dataset SCIFACT1, which has 5,183 scientific papers with titles and abstracts and 1,109 claims to experiment with joint learning for three different tasks Abstract, Rational, and Stance Verification in the same pipeline.

The approach proposed by Darwish et al. [13], Dong et al. [31] and Darwish et al. [32] are related to stance classification at the user level. Darwish et al. [13] used two different datasets, Islands and Islam, for the Stance Detection task. Dong et al. [31] used the crawled news articles from CNN with user comments for 4 of these articles. Rashed et al. [23] employs the largest available Turkish dataset on election-related tweets of a count of over 108k tweets collected between April 29 and June 23, 2018.

2.4 Previous Work on Contrastive Learning

As discussed in Section 1.1, the Stance Detection task involves identifying the stance of the author of the text on a controversial topic, however it is not always necessary that author conveys their stance explicitly. The author's text can implicitly infer the stance with the reasons provided. Hence, it is important for the Stance Detection system to understand the sentence-level meaning of the text, not just the word-level semantics. Inspired by the work of Oord et al. [3], we use Contrastive Learning in combination with the MLM objective to learn sentence-level and word-level semantics of the text. Contrastive Learning makes the similar examples (examples belonging to the same class label) to have similar representations in the representation space which makes the language model to be less sensitive (more robust) to the adversarial errors including changes in the vocabulary of the text. The following are the works carried out previously on Contrastive Learning to learn representations of text.

Sun et al. [33] used the Contrastive Learning framework to alleviate the exposure bias problem (discrepancy between training and inference) in text summarization by decreasing the likelihood of the low-quality of the summaries generated during inference time and at the same time increasing the likelihood of the golden summary by providing golden summary tokens as input to the decoder during training time. However, the golden summary tokens will not be available during inference time, and the generated tokens will be used in place of golden summary tokens (silver summary). During inference time, the beam search algorithm [34] is used for other candidate summaries, and the summary with the highest beam search score is selected as the output summary. It should be noted that, during inference, the token $\mathrm{y_i}$ is predicted using the tokens previously generated $\mathrm{y_{<i}}$. The Contrastive Learning Loss decreases the negative score and increases the positive score, and the model is optimized to have a higher positive score than a negative score. The overall loss function for the training data includes the Contrastive Learning Loss and the Negative Log Likelihood loss calculated with the golden summary during training.

A Contrastive Learning Framework is used to improve faithfulness and factuality in Abstractive Summarization [35]. The summarization model is trained to distinguish between the true reference summaries and the automatically generated faulty summaries. Positive and Negative Samples for a given article are used during training so that the model learns to discriminate between the reference and faulty summaries and hence acquire better representations.

Instead of using human-generated summaries (gold standard) for each test summary for evaluation, the quality of the summaries is evaluated without reference summaries by unsupervised Contrastive Learning [36]. Here the evaluation metric is the combination of linguistic quality and semantic informativeness based on BERT. The semantic similarity between the target summary and the source document is calculated using the cosine similarity function. The linguistic quality of the target summary and the source document is calculated with the probability of the sequence based on its representation. The linguistic and semantic scores are combined to form the evaluation metric. The negative samples (summary with worst quality) for Contrastive Learning are created by deleting random words, adding new sentences, or rearranging words.

Contrastive Learning is used to answer out-of-domain questions [37], which is useful when the size of the text corpora is small and when cross-domains are involved, such as operating with different domain data during the training and testing phase. The questions for a given context are generated using the QAGen-T5 model. The answer tokens are considered one class, and the question and context tokens are considered a separate class. The contrastive adaptation loss is applied in two folds – intra-class and inter-class. The domain invariant feature is learned by decreasing the discrepancy between answer tokens and among other answer tokens (intra-class). At the same time, the answer-context and answer-question discrepancy is increased (inter-class). The contrastive adaptation loss minimizes the intra-class discrepancy, and the last term in the equation increases the inter-class discrepancy. The overall training objective combines the cross-entropy loss and the contrastive adaptation loss. This method thus learns domain invariant features, capturing information from both the source and target domains and transferring knowledge to the target distribution.

Contrastive Learning is used to pre-train the model for zero-shot video and text understanding without using labels on downstream tasks [38]. This method works with multi-modal inputs such as video and text. The video and text tokens are obtained from the video and text. The proposed approach aims at pretraining the unified video-text representation captured by the Transformer model parameters for video and text and consequently using it for zero-shot downstream tasks. Contrastive loss is used to learn the correspondence between the video and text, i.e., the sum of the two contrastive losses is minimized. The two contrastive losses include video-to-text similarity and text-to-video similarity. This pretraining method is applied to a variety of end tasks such as Text to Video retrieval (HowTo100M dataset), multiple choice Video Questions (Youcook2, MSR-VTT, and DiDeMo datasets), Action Step Localization (CrossTask dataset), and Answer and Action segmentations (COIN dataset).

To eliminate the gap in the semantic matching of context and response from a dialogue between the training and evaluating phase, this paper proposed a novel approach, Dialogue-based Contrastive Learning of Sentence Embeddings [39]. For each candidate response embedding, this method generates a context-aware embedding, and these two embeddings are used in a Contrastive Learning framework to minimize the contrastive loss across all the combined pairs. Context-aware embedding is generated with the help of multi-turn matching matrices generated by taking a dot product of all the utterances from the context with the response. The label for the context-free embedding and context-aware embedding pair is determined by whether the context and response are from the same dialogue. Positive samples for Contrastive Learning are generated by combining context-free embedding and context-aware embedding derived from the same dialogue. The negative samples are generated by randomly sampling an utterance from dialogue and combining context-free embedding and context-aware embedding of those utterances with the response. The experiment is carried out with three multi-turn dialogue datasets. 1) The Microsoft Dialogue Corpus (MDC), 2) The Jing Dong Dialogue Corpus (JDDC), and

3) The E-commerce Dialogue Corpus (ECD). Semantic Retrieval (SR) and dialogue-based semantic textual similarity (D-STS) are evaluation metrics for the experiment. Self-supervised learning methods and dialogue-based self-supervised learning methods are taken as baselines for the evaluation. The experimental results show that the proposed model achieves the best performance in terms of all metrics across the three different datasets. This performance improvement is attributed to the reason that the proposed model identified semantic relationships in each utterance-response pair which distills the vital information at the turn level from the multi-turn dialogue context rather than using the element-wise distance metrics such as the cosine similarity or L2 distance.

Zhang et al. [40] use few-shot learning framework for intent detection. Few shot intent detection is achieved in two folds with pretraining and then finetuning. The semantically similar utterances from the intent dataset are discriminated from the non-similar utterances (discrimination) without any labels (self-supervised contrastive pre-training). Then, few-shot intent detection is performed with supervised Contrastive Learning, which makes the similar utterances even closer, and the non-similar utterances are pushed apart furthermore.

Contrastive Learning is used to learn noise invariant sentence representation with the help of different sentence-level augmentation strategies like span deletion, substitution, and reordering [41]. For each sentence, two random augmentations are generated, which form the positive pair for that sentence. All the other instances from the same batch during training are considered to create a negative pair for Contrastive Learning. The loss function (for positive pairs) is defined as the cosine similarity of two vectors u and v. The overall Contrastive Learning, the loss is given by the sum of all positive pairs in the same batch. The overall loss function for the training phase is the combination of Contrastive Learning loss and masked language modeling.

To get better text representations for text classification tasks with limited annotations, contrastive samples are constructed for language tasks using text summarization [42]. Here text summarization and mix sum (a combination of mix-up and summarization) is used as a data augmentation method to create negative and positive samples for supervised Contrastive Learning. The Mixsum method combines texts from different categories to create new samples for Contrastive Learning. For text summarization, the PreSumm method is used, which utilizes BERT as a general framework for both extractive and abstractive summarization. The objective is to make the classifier learn a good decision boundary by minimizing the intra-class representation distance and maximizing the inter-class representation distance with the help of Contrastive Learning. Since training data is limited in few-shot learning, fine-tuning the already pre-trained model with cross-entropy loss cannot achieve optimal performance.

2.5 Robustness in NLP Models

The NLP models are trained and then deployed in the real world to assist humans with various downstream tasks such as Sentiment Analysis, Speech Recognition, Language Translation, Question Answering, etc. These models are trained with the scenarios already encountered in the real world, i.e., the models are trained with the dataset collected for the specific purpose. However, it is not always possible for the NLP models to encounter versions similar to the examples seen during the training. The models are expected to be reliable and robust against challenging scenarios. The studies show that humans can recognize and understand small perturbations in the input text; however, the NLP models still struggle to preserve the original meaning of the input, and they are easily deceived when they see adversarial examples. These adversarial examples include examples having spelling errors, missing words, replaced words, paraphrased sentences, etc. Hence, the robustness or reliability of a model is defined as follows: Let 'x' be the input text, 'y' be the ground truth label associated with 'x', 'p' be the model trained on (x,y) and p(x) be the predictions from the model for the input 'x'. The trained model 'p' is now allowed to make predictions over the test data (x_{T_o}, y_{T_o}) and p(x_{T_o}) be the predictions from the model for the input 'x_{T_o}'. Now the robustness of the model can be calculated by having the model 'p' to make predictions for the perturbed test set (x_{T_p}, y_{T_p}). The perturbed test set (x_{T_p}, y_{T_p}) is generated by using various adversarial attack techniques.

Dong et al. [43], Zhang et al. [44] and Wang et al. [45] measured the adversarial robustness of the model by having the model make predictions against the test set with char-level and sequence-level modifications to the input as well as with word substitutions. Furthermore, To check if there is any deviation in the outcome when the input is perturbed, Moradi and Samwald [46] used various perturbations for Char-level such as Insertion, Deletion, Replacement, Swapping, Repetition, Common Misspelled word (e.g., Flourescent), Letter Case Changing and for Word-level such as Deletion, Repetition, Replacement with Synonyms, Negation, Singular Plural Verb, Verb Tense, Word Order.

The defense strategy of training the model with adversarial examples [47] improves the model's potential to battle against the adversarial examples in the test set. Alshemali and Kalita [47] categorized the robustness check with stress tests and adversarial strategies. The stress tests include distracting the model by having negation, word overlap, and length mismatch in the test inputs. The model is introduced to noisy test samples by swapping or substituting the characters of the text. The adversarial strategies, however, affect the output for the corresponding inputs. The strategies such as swapping words and paraphrasing do not change the outcome of the input, and the strategies such as negating the input and introducing antonyms for words in the input alter the result of the input sample.

The robustness of the model is identified in the cross-domain environment [29] by having the model trained with a dataset of a particular domain and testing the same model with a dataset of a different domain. Table 2.2 summarizes the various adversarial attacks that can be introduced into a test set.

Level of attack	**Adversarial Attack**	**Original Sample**	**Perturbed Sample**
Char Level	Insertion	Plants and trees make oxygen which we breathe	Plants and trees malke oxygen which we breathe
	Deletion	Green is the way forward	Green is the way forard
	Repetition	Weather patterns evolving very differently over the last few years	Weather patterns evolvving very differently over the last few years
	Swapping & Spelling Error	Observations on the atmosphere and oceans reflect the human influence in climate change	Observations on the atmosphare and oceans reflect the humna influence in climate change
Word Level	Deletion	True education and a free mind is the best weapon against any obstacle	True education and a free mind is the weapon against any obstacle
	Repetition	The more education you have the more opportunities	The more education you have the more more opportunities
	Synonym Replacement	Golf is one of dozens of independent sports like running or swimming	Golf is one of dozens of stand-alone sports like running or swimming
	Negation	The Olympic spirit is a universal message for peace and togetherness	The Olympic spirit is not a universal message for peace and togetherness
	Adding Tautology	An ICC enforcement arm would make the ICC more credible as an organization	False is not True and an ICC enforcement arm would make the ICC more credible as an organization
Sentence Level	Paraphrasing	The Olympics create a sense of national pride	The Olympics instil a sense of pride in the country

Table 2.2: Illustration of the different types of adversarial attacks for perturbing the test set to measure the robustness and the reliability of the model.

Now assume the model's performance with the test set as P(x_{T_o}) and the model's performance with the perturbed test set as p(x_{T_p}). It is essential to measure the performance difference of the model between the original and the perturbed test to indicate the robustness of the model 'p'.

Schiller et al. [27] used the resilience score introduced by [48] to measure the robustness of the model, and also the impact of the adversarial attacks is taken into account with the help of the potency score [48] and the correctness ratio while calculating the resilience score. The Correctness ratio, Potency, and Resilience score are explained in detail using their corresponding mathematical equations in the following sections.

2.5.1 Potency Score

The Potency of an adversarial attack 'a' given in Equation 2.2 provides the measure of the effectiveness of an adversarial attack and is defined as the reduction in score from a perfect score across all the systems, s ∈ S weighted by the correctness ratio c_a. p(s,a) is the performance of the model for the system 's' against the adversarial attack 'a'

The correctness ratio c_a for an adversarial attack 'a' is the ratio of correctly transformed (perturbated) samples to the total number of samples considered in the adversarial attack.

$$\text{Correctness Ratio, } c_a = \frac{\text{Total No. of Correctly Transformed samples}}{\text{Total No. of samples considered}} \tag{2.1}$$

$$Potency(a) = c_a \frac{1}{|S|} \sum_{s \in S} (1 - p(s,a)) \tag{2.2}$$

where:

s is the Natural Language Models system

c_a is the correctness ratio of the adversarial attack 'a'

p(s,a) is the performance of the model for the system 's' against the adversarial attack 'a'

2.5.2 Resilience Score

The resilience score provides the robustness of the model against all the adversarial attacks scaled by the correctness ratio. The performance of the model is measured with the original test set and each of the adversarial attacks separately. The correctness ratio of each adversarial attack is calculated with the Equation 2.1. The resilience score (Equation 2.3) measures the deviation in performance of the model from the original test set to the adversarial perturbed test set scaled by the correctness ratio of the corresponding adversarial attack.

$$Resilience = \left| \frac{\sum_{a \in A} c_a * (p(s,t) - p(s,a))}{\sum_{a \in A} c_a} \right| \tag{2.3}$$

where:

s is the Natural Language Models system

c_a is the correctness ratio of the adversarial attack 'a'

p(s,t) is the performance of the model for the system 's' on the original test set

p(s,a) is the performance of the model for the system 's' against the adversarial attack 'a'

Chapter 3

Methodology

This Chapter elaborates on the methodology and other frameworks used for the stance detection task. The methodology mainly involves the Transformer model DistilRoBERTa, Contrastive Learning, and Masked Language Modeling for pretraining the model with the stance dataset, followed by finetuning the model for the Stance Detection downstream task. To evaluate the approach, the Robustness of the model is calculated through the resilience score, which provides the deviation in the model's performance between the original test set and the perturbed test set, see section 2.5.2.

Our methodology uses the pretrained DistilRoBERTa [49], which is trained on OpenWebText corpus [50], for training the Stance Detection datasets with the Contrastive Learning and Masked Language Modeling [51] objectives.

The MLM objective in the DistilRoBERTa pretraining captures the word-level representations. However, in a Stance Detection problem, it is crucial to capture the sentence-level meaning of a text since the text may or may not contain the topic of discussion. In other words, there may not be any correlation between the lexicons of the topic and the lexicons contained in the text. The example (taken from the dataset DebateForum, see section 4.1), *'They are living homo sapien. What if you were a little unicellular embryo, and you had the vote, would you kill yourself?'* is having a stance *'against'* on the topic *'Legalization of Abortion'*. The lexicons in the example did not contain *'Abortion'* or its synonyms; however, the sentence takes a stand regarding the topic. The words in the text can implicitly refer to the topic; hence, it is imperative to understand the sentence-level meaning of the text concerning the topic.

3.1 Contrastive Learning for Stance Dataset

This section explains the proposed methodology for labeled stance dataset and identifies the need for learning representations from the unlabeled stance dataset.

3.1.1 Proposed Methodology on Labeled Dataset

Contrastive Learning groups the texts of similar views together and makes the representations of the text of the same class more similar and closer in the representation space; the model tends to learn more about the sentence. The contrastive loss function described in Equation (3.1) needs three inputs, an anchor text $\mathrm{d_Q}$, a sample that has the same ground truth label as anchor text $\mathrm{d_+}$ (positive example for anchor), and a sample that has a different ground truth label as anchor text $\mathrm{d_-}$ (negative example for anchor). As described in Figure 3.1, in a batch of examples, we sample three texts to form a triplet, i.e., for a given anchor text $\mathrm{d_Q}$, we sample a positive pair to it (which is like the anchor in terms of ground truth label) and a negative pair (which is not like the anchor in terms of ground truth label). The encoder provides the representations for the sampled triplet. The distance between the embeddings is minimized for the anchor-positive pairs and maximized for the anchor-negative pairs with the help of the contrastive loss. The 'm' in the contrastive loss equation 3.1 defines the desired difference between the anchor-positive and anchor-negative distances. The goal of the contrastive loss function is to minimize the distance between the anchor-positive example pair and to maximize the distance between the anchor-negative example pair.

$$ContrastiveLoss = max\{|d_Q - d_+| - |d_Q - d_-| + m, 0\} \tag{3.1}$$

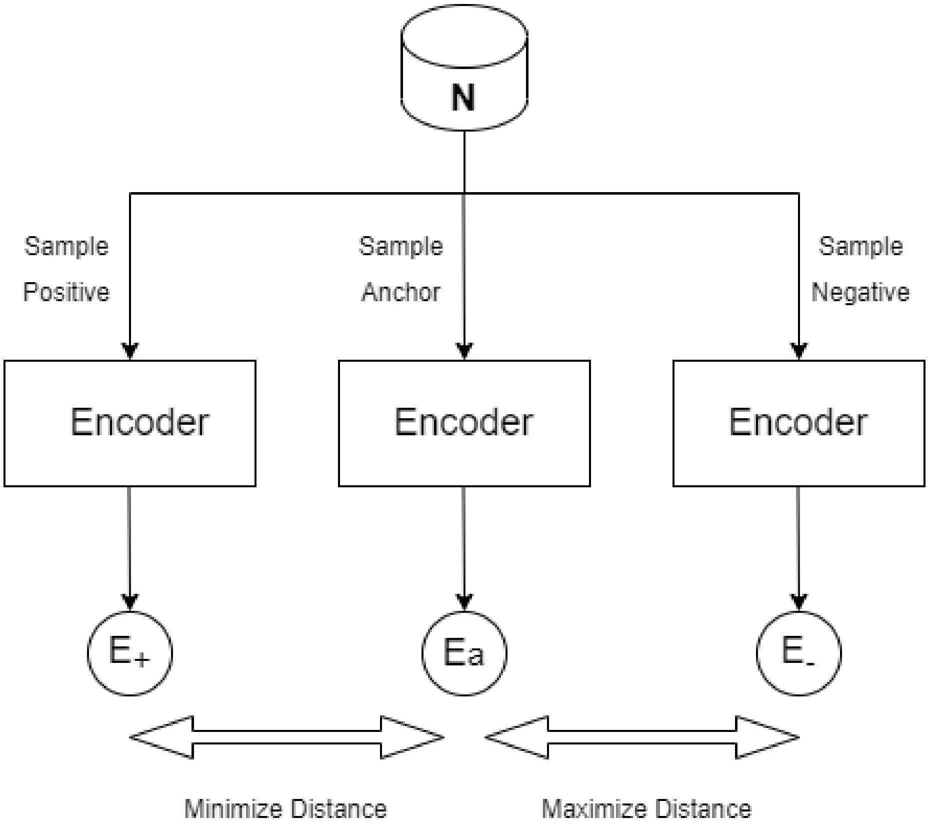


Figure 3.1: Overview of Contrastive Learning Framework that samples an anchor, positive and negative sample for the anchor from a batch of N samples. The distance between the anchor and positive samples should be minimized, and the distance between the anchor and negative samples should be maximized.

To learn the word-level representations, the Masked Language Modeling objective is leveraged by randomly masking tokens of an example and allows the model to predict the masked token during training. The MLM objective is implemented in each example; the final loss is the sum of the MLM loss and the contrastive loss.

$$Loss = ContrastiveLoss + MLMLoss \tag{3.2}$$

The representations learned with the Contrastive Learning and MLM objectives are then finetuned with the same stance dataset. The model is attached with a classification head as the final layer to output the probabilities for the stance labels; see Figure 3.3.

The test dataset is perturbed with three adversarial attacks spelling errors, tautology addition, and synonym replacements (see Table 2.2 for examples). The model's performance is captured for the original test set and the perturbed test set to identify the Robustness of the model. The Robustness of the model is calculated through the resilience score (explained in Section 2.5.2), which tells if there is any deviation in the model's performance with the perturbed test set from the model's performance with the original test set.

3.1.2 Proposed Methodology on Unlabeled Dataset

The contrastive loss function given by Equation 3.1 requires a ground truth label to be more effective, with which the positive and negative examples are separated from the anchor example. In the case of the Stance Detection problem, it is essential to learn sentence representation. It is standard that subject matter experts manually annotate the stance labels for all the examples in the dataset. The effort needed to manually annotate the stance labels is huge, indicating that data labeling is costly. Also, the model based on labeled data tends to be domain specific. To avoid this manual effort, we have experimented by discarding the ground truth label of the dataset during pretraining with the Contrastive Learning and Masked Language Modeling. Since contrastive loss requires stance labels to form anchor-positive-negative triplets, we have used a clustering method 'Unsupervised Belief Representation Learning with Information-Theoretic Variational Graph Auto-Encoders' [52] to cluster the 'assumed' unlabeled dataset. Here initially, the metadata of users, such as the username and whether the text is a reply post to another post, are considered for clustering the users into two groups. The texts posted by the users are then picked to form two groups for Contrastive Learning. Inspired by [52], Figure 3.2 illustrates the two user groups and the post made by each user within the group. The posts are about the topic Legalization of Abortion. The users U1, U2, and U3 have the same ideology or stance toward the topic, whereas the users U4, U5, and U6 form a group with a sim-

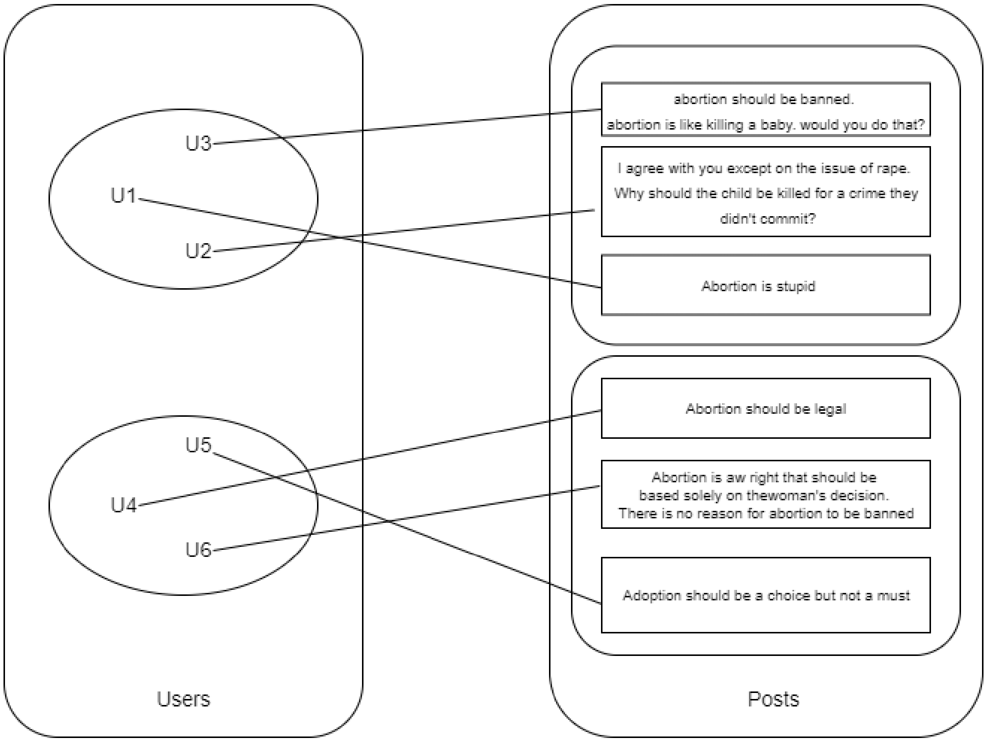

U3
U1
U2
U5
U4
U6
Users
abortion should be banned.
abortion is like killing a baby. would you do that?
I agree with you except on the issue of rape.
Why should the child be killed for a crime they didn't commit?
Abortion is stupid
Abortion should be legal
Abortion is aw right that should be based solely on thewoman's decision. There is no reason for abortion to be banned
Adoption should be a choice but not a must
Posts

Figure 3.2: Illustration of two different groups of users post for the topic Legalization of Abortion.

ilar stance toward the topic. The edge between the Users and Posts block indicates that the User Ui has posted the corresponding text for the topic of discussion, the Legalization of Abortion.

3.2 Strategies Used in Contrastive Learning

In our experiments, different strategies, as described below (Random, Hard, and Hard & Easy strategies), are used to select positives and negatives for a particular anchor for Contrastive Learning. The combination of anchor, positive and negative, is called a triplet.

Let n be the size of the batch of training examples, for each anchor sample S^A, n^+ is the number of positive samples selected for an anchor, and n^- is the number of negative samples chosen for the anchor.

3.2.1 Random Strategy

The triplets are formed randomly, satisfying the anchor-positive and anchor-negative selections. For example, if the batch size is 8, with the random triplet strategy, all possible triplets for an anchor (8 anchors in this case).

3.2.2 Hard Strategy

The hard positives and hard negatives are chosen for an anchor to form a triplet. A hard positive is an example with the same ground truth label as the anchor example, but it is not close to the anchor example in the embedded feature space. A hard negative is an example with a different ground truth label from the anchor example, but it is close to the anchor example in the embedded feature space.

3.2.3 Hard & Easy Strategy

For an anchor sample, one hard and one easy sample are chosen for both the positive and negative. This strategy is especially useful when the stance labels are unknown, or pseudo-stance labels from unsupervised methods are assigned to the examples.

3.3 Proposed Methodology for a More Robust Stance Detection

Our methodology forms the below pipeline for the Stance Detection problem as illustrated in Figure 3.3.

- Using the Contrastive Learning and MLM objectives, an embedding space is learned such that similar samples or samples belonging to the same group (same class labels) will have close representations. In contrast, the dissimilar samples or samples belonging to a different group will have representations far from each other.

- The trained model, with the text representations for the Stance Detection dataset, is finetuned with a classification head as the final layer to classify the samples either 'for' or 'against' the topic.

- The finetuned model is then tested with two sets of test data. The first test data is the original test set without any perturbations. Different adversarial attacks, such as spelling errors, adding tautology, and synonym replacements, are introduced into the second test set to check how perturbations affect the model's performance.

- The Robustness of the model (how the machine learning model behaves when there are any perturbations identified in the text) is calculated through the resilience formula as described in Section 2.5.2

The pipeline described above is suitable for the Stance Detection task with labeled stance datasets having the ground truth label for all the stance examples. Since the Stance Detection task is not majorly explored with the unlabeled dataset, we wanted to extend this framework to the unlabeled stance datasets. We have considered the labeled dataset again for this experiment but assumed it to have no ground truth labels. In this case, the ground truth labels for the stance examples are unknown for the initial step. However, the Contrastive Learning framework requires labels of stance examples to select the triplets. Hence, the unlabeled stance dataset is first clustered to form two groups. The basis for clustering is the user's metadata, the user name, the number of posts that the users have made, and the posts they replied to. A threshold for the number of posts that the user should have made and a threshold for the number of words that the posts should have made are used in the clustering algorithm to work better in finding the clusters. We have followed the clustering algorithm proposed by Li et al. [52] called InfoVGAE to form the required clusters with the help of users metadata. The samples in the same clusters are assigned the same label. The combined loss (Contrastive Learning + MLM) pretraining is then

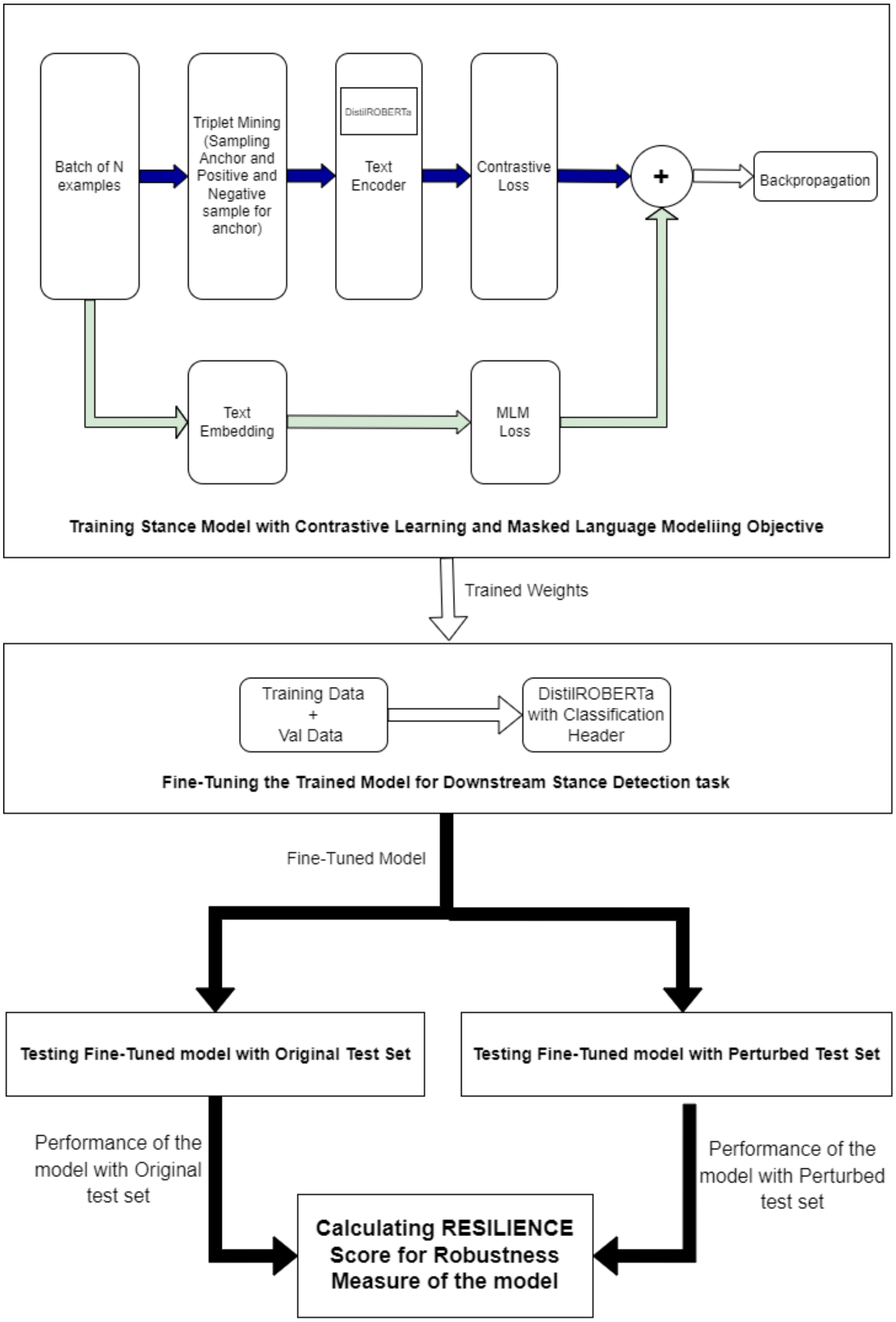

Batch of N examples
Triplet Mining (Sampling Anchor and Positive and Negative sample for anchor)
DistilROBERTa
Text Encoder
Contrastive Loss
+
Backpropagation
Text Embedding
MLM Loss
Training Stance Model with Contrastive Learning and Masked Language Modeliing Objective
Trained Weights
Training Data + Val Data
DistilROBERTa with Classification Header
Fine-Tuning the Trained Model for Downstream Stance Detection task
Fine-Tuned Model
Testing Fine-Tuned model with Original Test Set
Testing Fine-Tuned model with Perturbed Test Set
Performance of the model with Original test set
Performance of the model with Perturbed test set
Calculating RESILIENCE Score for Robustness Measure of the model

Figure 3.3: Architecture diagram of the proposed model for Stance Detection with Contrastive Learning and MLM objective for pretraining and Robustness check of the model with the Resilience score calculated between original and perturbed test sets

carried out with the labels assigned to the samples from the clustering to have robust representations for the given stance datasets. Unlike the labeled dataset experiment, the pretrained model (labels from clusters) in our pipeline is then finetuned with the same dataset but with the original ground truth label of the stance dataset.

Labeling the stance examples requires domain-level or topic-level knowledge and a lot of manual effort to analyze the text to determine the pertinent stance labels. Also, labeling the stance examples manually is a time-consuming, and financially costly process. We extended the methodology illustrated in Figure 3.3 to accommodate the unlabeled stance datasets. Our proposed methodology allows the model to learn from the unlabeled datasets to make the model independent of the domain-specific labels and to speed up the construction of the representations for different domains concerning stance content.

The datasets used for the experiments, the parameters for the task, and the hyperparameters for the transformer model are explained in detail in Chapter 4.

Chapter 4

Experimentation Settings

This chapter explains the experiments conducted, their setup, and the datasets used for the experiments.

4.1 Datasets

We have chosen seven Stance Detection datasets from different domains. Some of the topics in the datasets considered for the experiments include 'Legalization of Abortion', 'Climate change is a real concern', 'Feminism', 'Atheism' etc. The main motive of the proposed methodology is to identify whether the text posted by any user in the social medium supports or refutes the topic of discussion, which requires two labels, either support or refute. Some of the datasets considered for the experiments have more than two stance labels. These labels include 'unrelated', 'comment', and 'query'. Table 4.1 describes the total number of instances in the datasets, the class labels and their ratio, and the train/dev/test split of the corresponding dataset.

The datasets SemEval2016 and FNC-1 are imbalanced in class ratio, with one of the classes in both datasets representing 65% or more of the stance labels.

Table 4.2 describes the datasets, the domain of the corresponding datasets, and an example from the dataset to show the input and the stance output.

The following introduces all the datasets used in the experiments for the Stance Detection task. Since this is a two-class Stance Detection task, the dataset is prepared

Dataset	# Examples	Classes	Splits		
			Train	Dev	Test
DebateForum	4904	for(60%), against(40%)	3431	589	884
SemEval2016	3170	favor(35%), against(65%)	2149	205	816
ARC	3368	agree(47%), disagree(53%)	2660	283	425
Perspectrum	11825	support(52%), undermine(48%)	6979	2072	2774
FNC-1	7121	agree(78%), disagree(22%)	4519	1301	1301
KSD-Biden	766	favor(50%), against(50%)	546	110	110
KSD-Trump	843	favor(41%), against(59%)	591	126	126

Table 4.1: Statistics about the different datasets used for the experiments

Dataset	**Domain**	**Example**	**Topic**	**Stance Label**
DebateForum	Debating Forum	Passive smoking is harmful and secondhand smoke from the use of marijuana increases the chances of others suffering the damage by inhaling the smoke.	Marijuana	against
ARC		This is a great move by Wal-Mart. I hope they take out all the high fructose corn syrup out of their products as well. I avoid anything with high fructose corn syrup and as a result I have lost 37 pounds.	Wal-Mart can make us healthier	agree
Perspectrum		A game is less enjoyable if there is video replay.	There should be video replays for refs in football	undermine
SemEval2016	Social Media	Today Europe is breaking heat records, while Asia is breaking the lowest temperature records!! Should we not be concerned	Climate Change is a Real Concern	favor
KSD-Biden		i miss having a president that speaks eloquently. that has empathy and hope for a better tomorrow. fortunately, we will soon have that again with #bidenharris2020.	Biden	favor
KSD-Trump		not everyone in oklahoma is welcoming the president's visit	Trump	against
FNC-1	News	Tesla is reportedly choosing Nevada for its new battery factory.	Tesla to Choose Nevada for Battery Factory	agree

Table 4.2: Illustrates the domain of the different datasets used for the experiments and an example from each of the datasets

to have only two labels. The original dataset will already contain two labels, or the examples that contain only the for/against labels are preserved, and examples with other labels are removed.

DebateForum The DebateForum [53] Dataset contains posts from an online debate forum. The posts are on four different topics 'Abortion', 'Gay Rights', 'Marijuana'and 'Obama'. The examples have the class labels 'for' and 'against'.

SemEval2016 The SemEval2016 [2] dataset is a widely used dataset for the Stance Detection problem until recent [54]. It contains tweets on the topics of 5 major controversial topics Abortion, Atheism, Feminism, Climate, and Hillary Clinton that has text examples which are challenging and difficult to infer a stance towards the target topic. The stance labels are 'favor' and 'against'.

ARC The Argument Reasoning Corpus [55] contains examples consisting of a claim, a user post from a debating forum, and the stance label for the corresponding user post. The dataset contains various topics (e.g. WalMart can make us healthier) and an example user post for the topic is given in Table 4.2 The class labels of the examples are 'agree' and 'disagree'.

Perspectrum The Perspectrum [56] dataset contains claims and perspectives and their corresponding stance labels. The examples have the class labels 'support' and 'undermine'.

FNC-1 The Fake News Challenge [57] dataset contains examples from news websites with headline-article pairs. The class labels in this dataset are 'agree' and 'disagree'.

KSD-Biden, KSD-Trump These datasets [58] contain tweets related to the 2020 US presidential elections to determine the stance of the two presidential candidates, Joe Biden and Donald Trump. The class labels are 'support' and 'oppose'.

For the DebateForum and SemEval2016 datasets, the different topics, their specific class ratio, and the train/dev/test split of the datasets DebateForum and SemEval2016 are given in Table 4.3. Examples related to each topic from the datasets

Dataset	**Topic**	**Class Ratio**	**# Examples**	**Splits**		
				Train	**Dev**	**Test**
Debate Forum	***Abortion***	for(56%), against(44%)	1918	1341	288	289
	Gay Rights	for(64%), against(36%)	1378	963	207	208
	Marijuana	for(71%), against(29%)	629	439	95	95
	Obama	for(53%), against(47%)	988	690	149	149
Sem Eval2016	***Abortion***	favor(24%), against(76%)	714	498	108	108
	Atheism	favor(21%), against(78%)	591	412	89	90
	Climate Change	favor(90%), against(10%)	364	253	55	56
	Feminism	favor(35%), against(65%)	782	546	118	118
	Hillary Clinton	favor(23%), against(77%)	730	510	110	110

Table 4.3: The topicwise distribution of the datasets DebateForum and SemEval2016

DebateForum and SemEval2016 are also considered to form an intrinsic dataset for the topicwise (individual topics) experiments (section 4.2), which are explained in detail in the further sections. The topicwise datasets are comparatively small (< 1000 examples) and are highly imbalanced. These datasets are introduced in the experiments to identify the robustness of our proposed model with small and imbalanced datasets.

4.2 Experimental Setups

We conducted different experiments under different setups for the Stance Detection task. These are described below. Further, this section explains the various parameters for the experiments and hyperparameters for the neural network model used in the experiments.

4.2.1 Different Setups

The setups described here vary according to how the dataset was set or the level of information leveraged to train and evaluate the conceived models.

1. ***Dataset with Ground Truth Label***

   In this setup, the dataset has ground truth stance labels for every example.

2. ***Dataset without Ground Truth Label***

   In this setup, the dataset has the ground truth stance labels, but it is assumed to be not having the ground truth label. The ground truth label is removed from the dataset during training with Contrastive Learning and Masked Language Modeling.

3. ***Mixed Topics***

   Since all the datasets described in Table 4.3 consist of more than one topic, in this setting, we do not construct or evaluate the models on separate topics, i.e., by topicality. Instead, we consider all the examples of all topics as a whole during our experiments. We call this setting mixed-topic experiments. Each topic of discussion has two stance labels.

4. ***Individual Topics***

   In this setup, in contrast to the previous setting of Mixed Topics, the models are constructed and evaluated based on individual topic-related sub-dataset, e.g., Abortion for DebateForum, see Table 4.3. Each topic is an individual dataset with its own train/dev/test splits. We mainly consider DebateForum and SemEval2016 topics.

We have used the DistilRoBERTa as the transformer model (6 layers, 768 dimensions, 12 heads, and 82M parameters) for all our experiments. We have used the code architecture of Deep Contrastive Learning for Unsupervised Textual Representations

[59] and modified the loss objectives and the pipeline according to our experiment setup. The experiment setup uses the AllenNLP [60] library to set up the workflow for training the model with our proposed methodology; see Chapter 3. AllenNLP is a framework used for developing deep learning models. It provides a workflow to read the dataset and tokenize it for model training. The transformer model in our proposed methodology is not pre-trained from scratch. We use DistilRoBERTa pre-trained weights as the initial weights for the DistilRoBERTa model.

4.2.2 Processing and Tokenizing the Input Text

The number of characters and words used in social media posts is usually restricted to cut out the fluff. For example, currently, Twitter [61] has a character limit of 280 characters per post to express the user's thoughts. In all our experiments, we used a word limit of 100 to capture the valuable meaning of the user's post. The pre-trained DistilRoBERTa tokenizer is used for preparing inputs for training the model. The pre-trained tokenizer is applied over the input sequence to convert it into a numerical vector. The lazy loading option from the AllenNLP framework allows to load the dataset in a batch-size fashion from the disk rather than loading the entire dataset from the disk at once.

4.2.3 Model Hyperparameters

Our proposed methodology involves three components. The sentence-level features are learned using contrastive samples with the help of a Contrastive Learning framework. The token/word level features are learned with the help of Masked Language Modeling. The representations learned with the combined objective of Contrastive Learning and MLM are used in the stance classification task by adding a classification layer over the DistilRoBERTa transformer architecture. The hyperparameters for the model, Contrastive Learning framework, and Masked Language Modeling are provided in Table 4.4 and 4.5, respectively. In Table 4.4, the maximum sequence length hyperparameter is chosen as 100 as explained in Section 4.2.2. The triplet mining strategies such as Hard Strategy and Hard & Easy Strategy mine one and two triplets respectively from a batch of examples during training for Contrastive Learning. Hence, the batch size for training (Table 4.2.2) is reduced from 16 to 8 to allow maximum participation of different examples in Contrastive Learning. All the other hyperparameters in Table 4.2.2 are as per the transformer model's predefined values. The experiments with Setup 2, Datasets without Ground Truth Labels (section 4.2.1), have an additional step of clustering. This will group the text data points into two different clusters that are used in the representation learning step, as described in Chapter 3.1.2. The hyperparameters required for clustering using InfoV-GAE (the clustering algorithm) are provided in Table 4.6. The models are pre-trained on NVDIA 8GB GPUs for all experiments.

Hyperparameter	**Value**
Batch Size	8
Epochs	20
Maximum Sequence Length	100
Optimizer	Adam
Learning Rate	5e-5
Gradient Clipping	max norm: 1.0
Epsilon	1e-6
Weight Decay	0.1

Table 4.4: Hyperparameters for the experiments

Objective	**Hyperparameter**	**Value**
Masked Language Modeling	% of tokens masked	15%
Contrastive Learning	Margin	0.5

Table 4.5: Hyperparameters for the Objectives Contrastive Learning and Masked Language Modeling

Hyperparameter	**Value**
Post/Tweet count threshold	4
User count threshold	2
Epochs	100
Learning Rate	1e-2

Table 4.6: Hyperparameters for the clustering of the unlabeled stance dataset examples using InfoVGAE

4.2.4 Baseline Model

The DistilRoBERTa model, finetuned with the stance dataset, serves as the baseline for comparing our proposed methodology. Let B be the baseline model and ($\mathrm{x}^{(\mathrm{i})}$,$\mathrm{y}^{(\mathrm{i})}$) be the i^{th} instance with $\mathrm{x}^{(\mathrm{i})}$ be the input sequence and $\mathrm{y}^{(\mathrm{i})}$ be the corresponding label. $\mathrm{B}(\mathrm{x}^{(\mathrm{i})}) \longrightarrow \mathrm{y}^{(\mathrm{i})}$ is the prediction for the input sequence $\mathrm{x}^{(\mathrm{i})}$ by the baseline model B, and the performance of the baseline model is to be compared with that of our proposed methodology. The hyperparameters for the baseline model are provided in Table 4.7.

4.2.5 Contrastive Learning and Masked Language Modeling Methodology

This section explains the learning of sentence representations using Contrastive Learning and Masked Language Modeling objectives.

Learning Robust Representations

In our proposed methodology, let F be the transformer model (DistilRoBERTa), ($\mathrm{x}^{(\mathrm{i})}$,$\mathrm{y}^{(\mathrm{i})}$) $\in \mathrm{D}^{(\mathrm{j})}$ be the i^{th} instance of the batch j of dataset D, and $\mathrm{R_I}^{(\mathrm{i})} \longleftarrow \mathrm{F}(\mathrm{x}^{(\mathrm{i})})$ be the initial representation of the input sequence $\mathrm{x}^{(\mathrm{i})}$. For each of the input sequences $\mathrm{x}^{(\mathrm{i})}$ from batch j, the MLM objective masks 15% of tokens, and the model predicts the masked token with the help of the surrounding tokens. Again, for the same input sequences from batch j, the Contrastive Learning framework identifies the triplets for each $\mathrm{x}^{(\mathrm{i})}$ (anchor) based on the strategies explained in section 2.1.2. The Contrastive Learning generates a loss to minimize the distance between (anchor, positive) sequences pair and to maximize the distance between (anchor, negative) sequences pair. The combined loss (Contrastive Learning + MLM) is backpropagated to adjust the weights of the DistilRoBERTa model. $\mathrm{R_T}^{(\mathrm{i})} \longleftarrow \mathrm{F}(\mathrm{x}^{(\mathrm{i})})$ is the final representation of the text sequence $\mathrm{x}^{(\mathrm{i})}$ after training with the Contrastive Learning and MLM objectives.

Now the transformer model F trained with the Contrastive Learning and MLM objectives learned the representations of stance data in a contrastive fashion which can be leveraged in the Stance Detection task.

As mentioned in Section 3.2, we use three strategies for selecting positive and negative examples for an anchor example to learn the sentence representations using Contrastive Learning approach. In our experiments, the models trained with Contrastive Learning and MLM objectives using Random, Hard and Hard & Easy strategies are called $Model_{Random}$, $Model_{Hard}$ and $Model_{Hard\ \&\ Easy}$ respectively.

Hyperparameter	**Value**
Batch Size	16
Epochs	4
Optimizer	Adam
Learning Rate	5e-5

Table 4.7: Hyperparameters for finetuning the DistilRoBERTa model with stance dataset

Leveraging Robust Representation for Stance Detection

Now a classification layer is added on top of the model F (DistilRoBERTa). The weights of the model F are finetuned (see Table 4.7 for hyperparameters) with the stance dataset for Stance Detection/classification. Let $\mathrm{P}_{(\mathrm{o})}$ be the model's performance after finetuning with the stance dataset D. The robustness of model F is identified by testing the finetuned model F against the perturbed test set $\mathrm{D}_{(\mathrm{p})}$. The original test set D is perturbed with three adversarial strategies spelling errors, adding tautology, and synonyms (see Section 2.5 for examples). Let $\mathrm{P_p}^{(\mathrm{se})}$, $\mathrm{P_p}^{(\mathrm{n})}$, and $\mathrm{P_p}^{(\mathrm{sm})}$ be the performances of the model against the perturbed test sets generated with the adversarial attacks spelling errors, tautology, and synonyms respectively. The correctness ratio for the adversarial attacks is used in the experiments is calculated according to the Equation 2.1. The Correctness Ratio for the adversarial attack 'adding tautology' is 1 as the test data is perturbed by prefixing the example sentence with the words *False is not True and* which does not change the truth value of the sentence, hence the stance labels for the sentence remains the same. The Correctness Ratio for the adversarial attack 'synonyms replacement' is also 1 as the words in a sentence are replaced with its synonyms and hence the stance labels for the sentences remains the same. We use Flesch–Kincaid grade level [62] to check if the transformed sentence with the adversarial attack 'spelling error' is readable. We consider the example after perturbation which has the same readability grade level as the original example as correctly perturbed example. The Correctness Ratio of adversarial attack 'spelling error' is 1 as all the examples used in the experiments are correctly perturbed for all the datasets. Since the Correctness Ratio of all the adversarial attacks is 1, the potency of the adversarial attacks given by Equation 2.2 is the reduction of performance of models with the perturbed test set from the perfect score of 100%. The Resilience of the model (see section 2.5.2) against all the adversarial attacks given by Equation 2.3 measures how robust the model F is when there is a perturbation in the original input data.

Learning Robust Representations without Ground Truth Labels

Now we explain the experiment setting for our proposed model with the unlabeled dataset (as Setting 2 in section 4.2.1). Since the dataset is not containing the ground truth label, the pseudo-ground truth labels are assigned for the dataset with the help of the clustering method proposed in InfoVGAE. The user's metadata username and user replies are used to cluster the text data from the DebateForum dataset. The experiment with the unlabeled dataset is carried out with the label removed from the examples of DebateForum dataset. Table 4.6 shows the hyperparameters for the clustering method explained previously. The Post/Tweet count threshold is the minimum number of words/tokens a post/tweet should contain, and the User count threshold is the minimum number of posts/tweets a user should have posted in the debate forum to be considered in the clustering method. The InfoVGAE method provides two clusters from the input dataset; all the examples in a cluster are assigned the same stance labels. The DistilRoBERTa model is trained with the input text sequence and the stance labels assigned from clustering and use the Contrastive Learning and Masked Language Modeling objectives to learn the representations of the input text sequence.

The size of the dataset for the experiment with the unlabeled dataset setup is smaller than the labeled DebateForum dataset as the clustering method, which generates the pseudo-labels for the representation learning filters some of the input text sequences based on the threshold criteria mentioned in Table 4.6. The total number of examples for representation learning is 4164, which is approximately 700 examples lesser than the labeled DebateForum dataset. The train/dev/test split for the unlabeled dataset experiment is given by 2915/624/625, correspondingly. The class ratio for the same is for(47%) and against(53%).

Let F be the transformer model (DistilRoBERTa) and $(\mathrm{x}^{(\mathrm{i})},\mathrm{y}^{(\mathrm{i})}{}_{\mathrm{c}}) \in \mathrm{D}^{(\mathrm{j})}$ be the i^{th} instance of batch j in the dataset D where $\mathrm{x}^{(\mathrm{i})}$ is the input text sequence and $\mathrm{y}^{(\mathrm{i})}{}_{\mathrm{c}}$ be the label of the input text sequence identified in the clusters generated by the InfoVGAE clustering method. Let $\mathrm{R_U}^{(\mathrm{i})} \longleftarrow \mathrm{F}(\mathrm{x}^{(\mathrm{i})})$ be the representation of the input text sequence $\mathrm{x}^{(\mathrm{i})}$ after training with the contrastive Learning and MLM objective. The representation learning step is similar to the setup explained with the labeled dataset (see Setup 1 in Section 4.2.1). The representations learned are further used in the finetuning of the model with the same dataset but with the ground truth label $\mathrm{y}^{(\mathrm{i})}$. The hyperparameters for the training and finetuning of the model with the unlabeled dataset experiment are the same as the hyperparameters of the experiment with the labeled datasets as shown in Tables 4.4, 4.5 and 4.7.

The interpretation and analysis of the results of the experiments are discussed in Chapter 5.

Chapter 5

Results & Analysis of Performance and Robustness of the Models

In this chapter, we report the results of all our experiments. As explained in Chapter 4, the DistilRoBERTa model fine-tuned with the labeled stance dataset is the baseline to compare with our proposed methodology. The experiments are carried out with different setups, as explained in Section 4.2. The results of the experiments are interpreted and analyzed in different dimensions as follows. The results reported for the experiments with F1-score and resilience score are for 100%.

Dataset	**Models**			
	Baseline	***Model*** $_{(Random)}$	***Model*** $_{(Hard)}$	***Model*** $_{(Hard\ \&\ Easy)}$
DebateForum	64.06	**68.68**	62.22	62.97
SemEval2016	**74.04**	72.21	71.18	71.27
ARC	60.94	61.77	62.21	**62.25**
Perspectrum	65.5	**66.05**	64.75	63.15
FNC-1	48.86	**52.87**	52.63	52.2
KSD-Biden	82.08	**88.77**	85.22	84.21
KSD-Trump	86.95	**88.81**	85.97	83.58
Average	68.91	**71.30**	69.16	68.51

Table 5.1: Results (F1-score) of experiments on all the datasets without perturbation

5.1 Performance Analysis and Comparison of Models according to the Mixed Topic Setup

In this section, we present the results and findings of all the experiments carried out under the Mixed Topic setup (see section 4.2.1). Instead of assessing the models on distinct topics, in the Mixed Topic setup, we considered all the examples of all topics as a whole during our experiments. The experiments include testing different models with non-perturbed data as well as perturbed data with adversarial attacks such as spelling errors, adding tautology, and synonym replacements.

5.1.1 Evaluation of Models with Non-Perturbed Data

Table 5.1 shows the performance of different models on all the non-perturbed datasets. The evaluation metric for the experiment is the F1-score. The values in bold indicate the model with the highest performance (F1-score) for the respective dataset. The models based on our proposed methodology ($Model_{Random}$, $Model_{Hard\ \&\ Easy}$ and $Model_{Hard}$, see Section 4.2.5) are compared against the baseline model. We show that the models based on our proposed methodology outperform the baseline model in 6 out of 7 datasets. There is a 2.4% increase on average from the baseline performance for the $Model_{Random}$. Also, the $Model_{Hard}$ has a better performance than the baseline. However, the average performance is slightly lower than the $Model_{Random}$. The $Model_{Random}$ is trained to learn the representations with all the possible triplets for an anchor sentence from the training batch. The $Model_{Hard}$ takes one triplet for an anchor that contains a hard positive and a hard negative. The $Model_{Hard\ \&\ Easy}$ takes two triplets for an anchor, i.e. (anchor, hard positive, hard negative) and (anchor, easy positive, easy negative). Each dataset has several topics, some of which align with the same ideology, and this gives the $Model_{Random}$ an advantage over the other two methods. We see that the performance of the $Model_{Random}$ is better than the other two proposed models on five out of the six datasets which outperformed the baseline model. The $Model_{Random}$ has a $> 4\%$ and $> 6\%$ increase over the baseline model for the DebateForum and KSD-Biden datasets respectively.

The dataset FNC-1 is highly imbalanced with a class ratio of 78:22. The models based on our proposed methodology significantly increase performance over the baseline method, and the $Model_{Random}$ has the best performance (greater than 4% increase over the baseline).

Dataset	**Models**			
	Baseline	$\boldsymbol{Model_{(Random)}}$	$\boldsymbol{Model_{(Hard)}}$	$\boldsymbol{Model_{(Hard\ \&\ Easy)}}$
DebateForum	67.34 (64.06)	**68.71 (68.68)**	60.77 (62.22)	64.33 (62.97)
SemEval2016	71.31 (74.04)	71.76 (72.21)	70.65 (71.18)	**71.05 (71.27)**
ARC	60.63 (60.94)	63.43 (61.77)	65.06 (62.21)	**62.02 (62.25)**
Perspectrum	62.58 (65.5)	62.41 (66.05)	60.64 (64.75)	**60.71 (63.15)**
FNC-1	**49.61 (48.86)**	48.37 (52.87)	51.33 (52.63)	50.19 (52.2)
KSD-Biden	87.87 (82.08)	86.9 (88.77)	**85.26 (85.22)**	85.17 (84.21)
KSD-Trump	86.47 (86.95)	**88.79 (88.81)**	84.52 (85.97)	80.9 (83.58)

Table 5.2: Results (F1-score) of experiments on all the datasets perturbed with spelling error adversarial attack. The F1-score for the datasets without perturbation is provided in the parentheses

5.1.2 Evaluation of Models with Perturbed data

In this section, we analyze and compare the performance of the models with the data perturbed by the three adversarial attacks as follows.

Table 5.2 shows the performance of different models on all the datasets which are perturbed by the spelling errors adversarial attack. The F1-score of the corresponding non-perturbed test sets for all the models is given in parentheses. The bold values indicate the lowest difference in performance among the models between the non-perturbed test set and the test set with spelling errors. We show that our proposed models have the lowest difference in performance for 6 out of 7 datasets over the baseline method. Though the performance of the $Model_{Hard\ \&\ Easy}$ is comparatively lesser than the performance of the $Model_{Random}$ on the non-perturbed test set, the performance difference is lesser than that of the $Model_{Random}$ for three datasets. The $Model_{Hard}$ and $Model_{Hard\ \&\ Easy}$ learn with hard triplets when compared to the $Model_{Random}$ which learns with all the possible triplets. The representations learned with the $Model_{Hard}$ and $Model_{Hard\ \&\ Easy}$ are robust since the Contrastive Learning framework is provided with the selected Hard triplets that allow the model to learn clearer distinctions between similar and dissimilar examples and thus improve the structure of the embedding space.

Dataset	**Models**			
	Baseline	***Model$_{(Random)}$***	***Model$_{(Hard)}$***	***Model$_{(Hard\ \&\ Easy)}$***
DebateForum	47.26 (64.06)	64.31 (68.68)	63.3 (62.22)	**62.3 (62.97)**
SemEval2016	72.01 (74.04)	70.58 (72.21)	**70.97 (71.18)**	70.16 (71.27)
ARC	**61.25 (60.94)**	63.89 (61.77)	62.9 (62.21)	63.39 (62.25)
Perspectrum	47.58 (65.5)	56.19 (66.05)	56.17 (64.75)	**62.4 (63.15)**
FNC-1	62.45 (48.86)	53.66 (52.87)	**52.41 (52.63)**	56.33 (52.2)
KSD-Biden	89.62 (82.08)	**87.65 (88.77)**	88.87 (85.22)	88.71 (84.21)
KSD-Trump	85.65 (86.95)	87.97 (88.81)	**85.16 (85.97)**	84.37 (83.58)

Table 5.3: Results (F1-score) of experiments on all the datasets perturbed with tautology addition adversarial attack. The F1-score for the datasets without perturbation is provided in the parentheses

The performance of the different models on all the datasets which are perturbed by adding tautology and synonym replacements adversarial attacks (see Table 2.2 for examples) is shown in Table 5.3 and Table 5.4 respectively. Again, though the performance of the $Model_{Random}$ is better with the non-perturbed datasets, the difference in the performance of the $Model_{Hard}$ and $Model_{Hard\ \&\ Easy}$ is better than the $Model_{Random}$ for five out of six tautology addition perturbed datasets on which the baseline model was outperformed.

Dataset	**Models**			
	Baseline	$\boldsymbol{Model_{(Random)}}$	$\boldsymbol{Model_{(Hard)}}$	$\boldsymbol{Model_{(Hard\ \&\ Easy)}}$
DebateForum	**64.27 (64.06)**	68.08 (68.68)	64.44 (62.22)	60.57 (62.97)
SemEval2016	73.71 (74.04)	**72.01 (72.21)**	70.42 (71.18)	71.47 (71.27)
ARC	**61.19 (60.94)**	63.43 (61.77)	60.57 (62.21)	61.66 (62.25)
Perspectrum	**65.07 (65.5)**	65.03 (66.05)	64.07 (64.75)	62.8 (63.15)
FNC-1	53.2 (48.86)	51.01 (52.87)	51.33 (52.63)	**52.2 (52.2)**
KSD-Biden	88.77 (82.08)	86.9 (88.77)	86.12 (85.22)	**84.27 (84.21)**
KSD-Trump	85.65 (86.95)	**88.81 (88.81)**	85.16 (85.97)	83.58 (83.58)

Table 5.4: Results (F1-score) of experiments on all the datasets perturbed with synonym adversarial attack. The F1-score for the datasets without perturbation is provided in the parentheses

Dataset	**Models**			
	Baseline	$\boldsymbol{Model_{(Random)}}$	$\boldsymbol{Model_{(Hard)}}$	$\boldsymbol{Model_{(Hard\ \&\ Easy)}}$
DebateForum	93.24	98.33	98.42	**98.53**
SemEval2016	98.31	99.24	**99.5**	99.49
ARC	**99.71**	98.19	95.92	99.35
Perspectrum	92.91	95.16	95.55	**98.82**
FNC-1	93.77	97.61	**99.06**	97.95
KSD-Biden	93.32	98.38	**98.47**	98.16
KSD-Trump	98.97	**99.72**	98.97	98.84
Average	95.74	98.09	97.98	**98.73**

Table 5.5: Resilience of all the models with respect to the original test set and the perturbed test sets

5.1.3 Resilience of Models

Table 5.5 shows the resilience score which measures the sensitivity of the model to the changes and variations introduced by the adversarial attacks on the test data. The resilience of the model is calculated according to the Equation 2.3 for all the datasets against all the adversarial attacks. The proposed models outperform the baseline model in six out of seven datasets. As discussed previously, the $Model_{Hard}$ and $Model_{Hard\ \&\ Easy}$ are more resilient to adversarial attacks than the $Model_{Random}$ as the former are trained with hard triplets. The $Model_{Hard\ \&\ Easy}$ has the better resilience score compared to all the other proposed models and the baseline model.

5.2 Performance Analysis and Comparison of Models in Individual Topic Setup

In this section, we present the results and findings of all the experiments carried out under the Individual Topic setup (see Section 4.2.1). In the Individual Topic setup, the models are built and assessed based on individual topic-related sub-dataset. The experiments include testing different models with non-perturbed data as well as perturbed data with adversarial attacks such as spelling errors, adding tautology, and synonym replacements.

5.2.1 Evaluation of Models with Non-Perturbed data

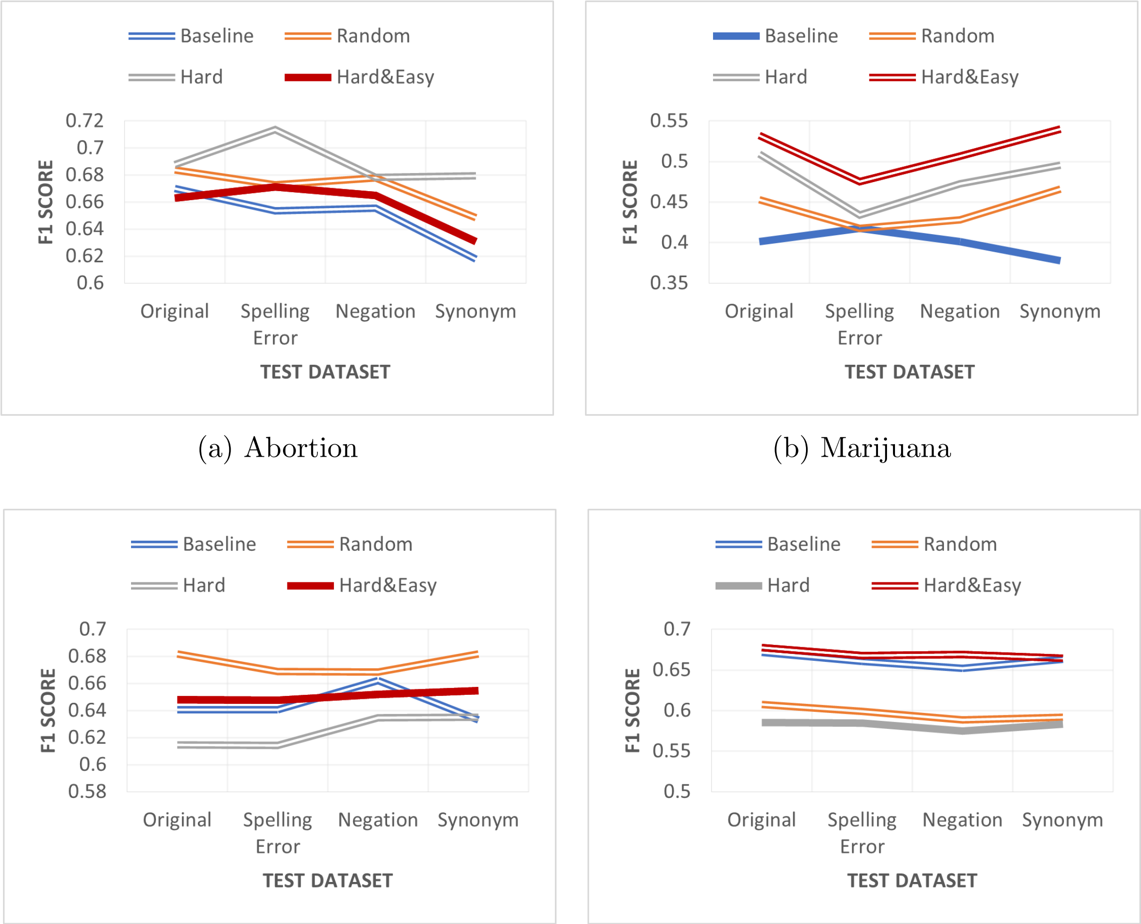
The baseline and our proposed models have been experimented with the individual topics from DebateForum and SemEval2016 datasets. The results are shown in Table 5.6. Out of 9 datasets (individual topics), our proposed method outperforms the baseline method in 8 datasets. It is noted that there is a significant increase in performance for the proposed models with the topics dataset $\text{Marijuana}_{\text{DebateForum}}$, $\text{Abortion}_{\text{SemEval2016}}$, $\text{Atheism}_{\text{SemEval2016}}$, and $\text{Climate}_{\text{SemEval2016}}$ which are relatively small datasets containing approximately less than 750 examples. The proposed models, especially $Model_{Hard}$ and $Model_{Hard \;\&\; Easy}$ trained with hard triplets, show better performance with the smaller datasets compared to the $Model_{Random}$ and baseline model. 4 out of 5 individual topics (Abortion, Atheism, Climate, Hillary Clinton) for the SemEval2016 dataset are significantly imbalanced with at least one of the class labels having more than 75% examples as shown in Table 4.3. There is a significant increase in performance demonstrated by the $Model_{Hard}$ for Abortion (9.8%) and Climate (>20%) datasets. Also, the $Model_{Hard}$ has a better performance for the Atheism dataset (3.3%) than the baseline model. This shows that the $Model_{Hard}$ trained with hard triplets show better performance with small as well as imbalanced datasets.

Dataset		**Models**			
		Baseline	***Model$_{(Random)}$***	***Model$_{(Hard)}$***	***Model$_{(Hard \;\&\; Easy)}$***
Debate Forum	***Abortion***	67.01	68.39	**68.78**	66.29
	Marijuana	40.14	45.31	50.94	**53.19**
	Gay Rights	67.14	60.75	58.51	**67.75**
	Obama	64.07	**68.2**	61.48	64.8
Sem Eval16	***Abortion***	71.39	74.3	**81.19**	78.68
	Atheism	77.14	78.18	**80.43**	77.14
	Climate	61.81	68.57	**82.37**	72.97
	Feminism	64.32	**65.06**	62.97	63.97
	Hillary Clinton	**84.63**	82.37	71.52	73.46
Average		66.31	68.79	**69.96**	68.81

Table 5.6: Results (F1-score) of experiments on all the datasets topic-wise without perturbation

5.2.2 Analyzing Resilience of Models

Figures 5.1 and 5.2 illustrate the performance of the different models for the non-perturbed dataset and the perturbed datasets with the spelling error, adding tautology and synonym adversarial strategies. The solid line indicates the model which has



(a) Abortion

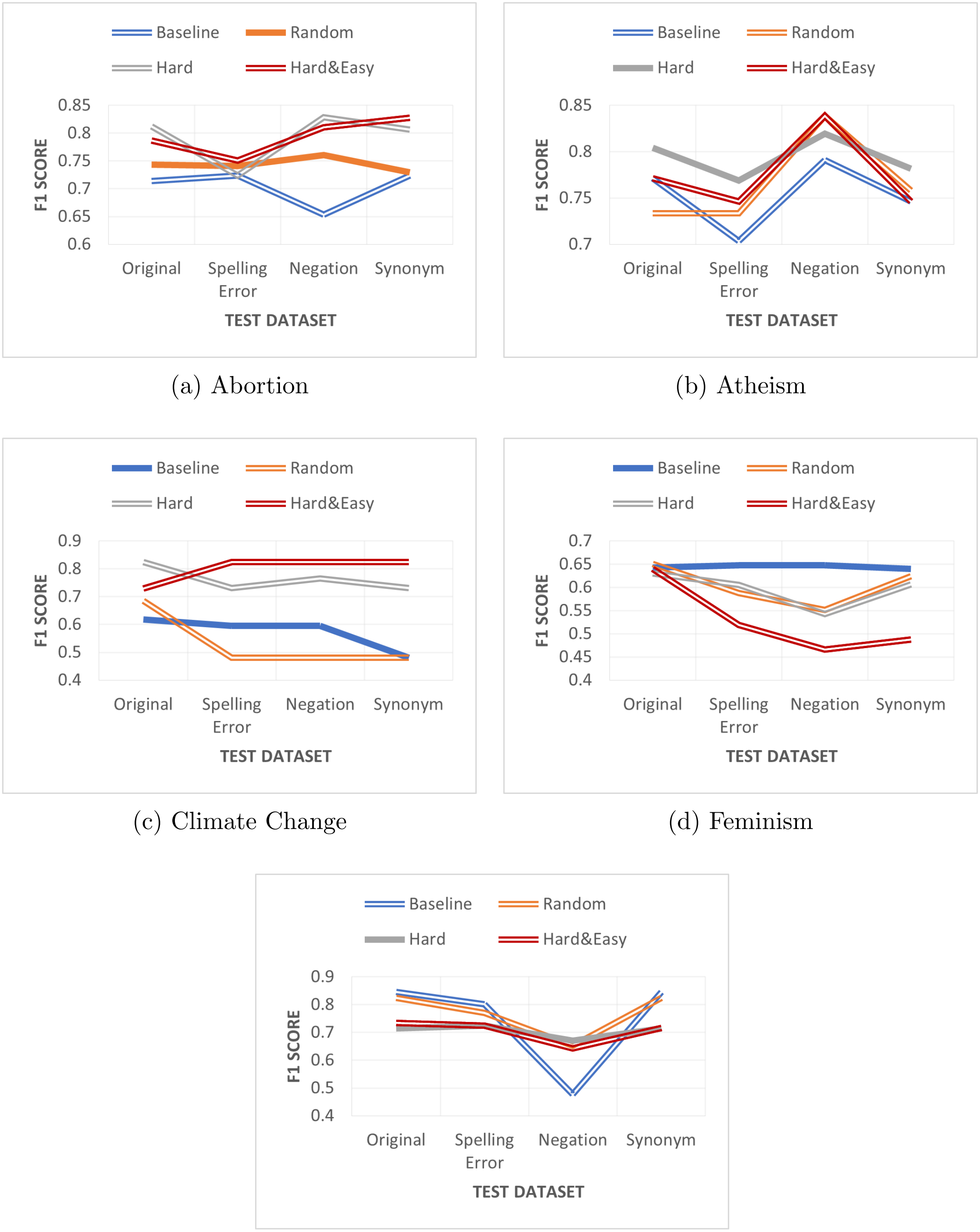
(b) Marijuana

(c) Obama

(d) Gay Rights

Figure 5.1: Illustration of the performance of the different models with the original test set and the perturbed test sets for each topic in the DebateForum dataset individually. The solid line in each of the graph indicates the model with better resilience than the other models

better resilience than the other models. Table 5.7 shows the resilience of the model for the individual topics of DebateForum and SemEval2016 datasets. Our proposed models outperform the baseline model in resilience score in 6 out of 9 datasets. The proposed models have a better resilience score than the baseline model for the smaller size and class-imbalanced datasets (Abortion, Atheism, and Hillary Clinton). The resilience of the proposed models is lesser than the baseline model for Climate topic dataset. Since the synonym adversarial attack replaces a word in a sentence with its synonym/meaning (see Table 2.2 for example), the meaning and context of the



(a) Abortion

(b) Atheism

(c) Climate Change

(d) Feminism

(e) Hillary Clinton

Figure 5.2: Illustration of the performance of the different models with the original test set and the perturbed test sets for each topic in the SemEval2016 dataset individually. The solid line in each of the graph indicates the model with better resilience than the other models

Dataset		**Models**			
		Baseline	***Model***$_{(Random)}$	***Model***$_{(Hard)}$	***Model***$_{(Hard\ \&\ Easy)}$
Debate Forum	***Abortion***	97.22	98.24	98.54	**98.58**
	Marijuana	**98.65**	97.56	95.79	96.99
	Gay Rights	96.61	95.66	**98.74**	96.85
	Obama	98.91	99.10	98.65	**99.64**
Sem Eval16	***Abortion***	97.33	**98.89**	96.31	96.63
	Atheism	96.23	95.70	**97.56**	96.07
	Climate	**93.89**	79.54	91.89	90.60
	Feminism	**99.60**	93.72	95.51	85.12
	Hillary Clinton	86.17	92.49	**98.19**	96.01
Average		96.06	94.54	**96.79**	95.16

Table 5.7: Resilience of all the models topic-wise for the DebateForum and SemEval2016 datasets with respect to the original test set and the perturbed test sets

sentence will not change, and hence the model should be able to preserve the classification output regardless of the original sentence or synonym perturbed sentence. Though the resilience of the baseline model is better for the Climate topic dataset when compared with the proposed models, Figure 5.2c shows that the performance of the baseline model decreased significantly ($\sim$14%) for the dataset perturbed with synonym adversarial attack, and the performance of the proposed model ($Model_{Hard}$) has a drop in performance of lesser than 10% for the same synonym perturbed dataset.

5.3 Performance Analysis of Proposed Models in Unlabeled data setup

In this section, we analyze the performance of our proposed models according to the setup with data having no ground truth labels, i.e., unannotated/unlabeled data (see Section 4.2.1. Table 5.8 shows the performance of the transformer model fine-tuned for the labeled DebateForum dataset with the representations learned from the

Dataset	Model$_{\textbf{(Random)}}$	Model$_{\textbf{(Hard)}}$	Model$_{\textbf{(Hard \& Easy)}}$
Original	64.68	62.34	61.08
after Spelling Error Perturbation	65.3	61.35	62.06
after Adding Tautology Perturbation	63.51	62.63	62.41
after Synonym Perturbation	64.53	63.18	61.02
Resilience	99.36	99.3	99.21

Table 5.8: Illustration of the performance (F1-score) of the different models with the original test set and the perturbed test sets from the unlabeled DebateForum dataset in the Mixed Topic setup.

unlabeled dataset using Contrastive Learning and MLM objectives. The performance of the $Model_{Random}$ is better than the models trained with other strategies (Hard and Hard & Easy). The $Model_{Random}$ has better performance (greater than 2%) since the topics are mixed within the dataset, and the $Model_{Random}$ considers all possible triplets for an anchor within the batch. The resilience of all the models is almost similar, and the $Model_{Random}$ has better resilience than the other two models. Though the size of the dataset is different for the unlabeled dataset experiment setup, the $Model_{Random}$ (unlabeled setup) has better performance than the Baseline method for the same dataset with labels (with F1-score of 64.06%, see Table 5.1). Also it is important to note that, the performance difference between the $Model_{Random}$ (unlabeled data setup) and the Baseline model (labeled data setup) with the tautology perturbed data is greater than 16%. The other two proposed models ($Model_{Hard}$ and $Model_{Hard\ \&\ Easy}$) in the unlabeled data setup achieves F1-score of greater than 62% better than the Baseline model with the labeled data setup. This shows that the proposed models (unlabeled data setup) are resilient to the strong negative words (false, not) which are perturbed into the test data in the form of a tautology.

5.4 Summary of Results

This section summarizes the results and findings of the experiments with different setups. In the Mixed Topics labeled data setup, our proposed methodology outperforms the Baseline model in terms of both the performance on the non-perturbed dataset as well as the resilience score averaged on all the datasets in the same setup. In the Individual Topics labeled data setup, the performance of the models with our proposed methodology again outperforms the Baseline model. One of our proposed model ($Model_{Hard}$) has greater than 3% increase on average over the Baseline model with the non-perturbed dataset and the other two proposed models have better performance over the Baseline model on average. In the Unlabeled data setup (Mixed Topics), the results shows that the resilience of the models with our proposed methodology is better than the resilience of the Baseline model with the annotated version of the dataset. Also, the proposed models trained under Unlabeled data setup is more resilient to the tautology perturbed dataset compared to the Baseline model trained with labeled stance data. The results of the Individual Topic labeled data setup show that our proposed methodology is effective for the small-sized dataset and also, the performance of our proposed models are better for the imbalanced stance classes data compared to the baseline model.

Chapter 6

Conclusion and Future Work

6.1 Conclusions

The motive of this research thesis is to create a pipeline framework to learn robust representations of sentences expressed in social media discussions regarding contentious issues. This thesis though partly focused on annotated data from social media such as Twitter, Reddit, etc., it also identified the need for learning the sentence representations from the unannotated data. The pipeline framework is extended to learn sentence representations from the unannotated data to eliminate the problem of requiring domain-specific features or labels. In this thesis, we first introduced the problems from the Stance-oriented tasks and the key concepts that are used to address the problems. Further, in Chapter 1 we explained the key contributions made toward the Stance Detection task for this thesis work. In Chapter 2, the closely related works and the background of several key concepts such as Stance Detection, Robustness in Natural Language Processing, and Contrastive Learning are explained. The methodology to accommodate and tackle the problems described in Chapter 1 is explained in detail with the framework in Chapter 3. The Contrastive Learning framework for the Stance-oriented tasks, clustering of unannotated stance examples, and the reliability measure to analyze the robustness of the stance Detection models are explained in Chapter 3. The experiments with different setups to analyze the stance Detection models in different dimensions such as the size of the dataset, and the nature of the dataset (annotated/unannotated, imbalanced stance labels ratio) are explained in Chapter 4 and its corresponding results are reported in Chapter 5. Further, in Chapter 5, the resilience or the reliability measure is reported for all the models which are exposed to different adversarial attacks during testing.

To summarize, in order to acquire more robust sentence representations to employ in the Stance Detection task, we created a method that makes use of a Contrastive Learning framework with different positive and negative pairs construction strategies (triplet-mining). We used three different strategies to build triplets (anchor, positive, negative) to learn the sentence representations in a contrastive fashion. These strategies include Random triplets (all possible triplets for an anchor in a batch of examples), Hard triplets (the far away positive and the closer negative for an anchor in a batch of examples), and Hard & Easy triplets (similar to Hard triplets but include one more triplet of closer positive and far away negative for an anchor). Along with the Contrastive Learning objective, we employed the MLM objective to learn the word-level representations during training with the stance datasets. To make the representation learning independent of domain-specific features or labels, we experimented with unannotated examples where we clustered them first to create two stance groups to use in Contrastive Learning. The results in Chapter 5 show that our proposed methodology with the annotated examples setting outperforms the baseline model in terms of the performance with the non-perturbed dataset and the resilience score. Also, as described in the results of Chapter 5, the proposed methodology is effective for the small-sized datasets and the stance class imbalanced datasets when compared to the baseline which is the traditional DistilRoBERTa model fine-tuned with the stance datasets. Though the size of the dataset for the unannotated examples experiment setup is slightly smaller than the size of the dataset for the annotated examples experiment setup (because of the clustering method in the initial step), the resilience of the model is relatively better than the resilience of the baseline model with the annotated version of the dataset. This shows that the stance of examples can be effectively identified even when the stance annotations are not associated with them. Next, we present the challenges, limitations, and possible improvements of the proposed approach identified, and discuss possible directions for the future.

One of the contributions of this thesis is to learn sentence representations in an unsupervised fashion i.e., to use unannotated stance examples. Since Contrastive Learning requires stance annotations for the examples to effectively build the triplets to learn the representations in a contrastive fashion, the unannotated stance examples are clustered to have intermediate stance labels required for Contrastive Learning. The clustering requires the user metadata to effectively cluster the stance examples. However, the stance datasets in the literature available from social media and debate forums are mostly annotated as described in Section 2.3. The experiment to learn sentence representations from the unannotated stance examples is limited to the DebateForum dataset. The sentence representation learning framework we proposed for the unannotated stance examples can be leveraged for the detecting stance for the polarized issues discussed in social media and debate forums.

6.2 Future Work

In this thesis, we considered the binary stances examples topics mainly i.e. for/against, support/refute, or agree/disagree. The proposed methodology leverages the Contrastive Learning framework which is conditioned to work with two stance labels examples to identify whether the author of the text is in favor of or against the topic of discussion. However, social media such as Twitter and online forums like Reddit will have threads discussing topics having more than two stances such as for/against/neither, or support/refute/comment. For example, authors posting unrelated comments in the discussion forum for the topic ‘Climate Change is a Real Concern’ would identify the stance of the authors as a ‘comment’, and thus the data for the topic will have more than two stances including ‘comment’ as another stance. Also, we analyzed the reliability or robustness of the models with three adversarial attack strategies (spelling errors, including tautology and synonym replacements). However, the text posted in the debate forums and social media may contain word-level errors such as repetition of crucial words in the context of the topic.

In future work, we propose to accommodate more than two stance labels in the proposed methodology and to experiment with other adversarial perturbation strategies for the reliability measure of the Stance Detection model.

This thesis is intended to be the first step of a research effort to provide a better Stance Detection system to identify authors stances on a controversial discussion topic. Hopefully, the contributions made here will result in improved systems for users and assist researchers in such development.

Bibliography

[1] D. W. Otter, J. R. Medina, and J. K. Kalita, "A survey of the usages of deep learning for natural language processing," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 2, pp. 604–624, 2021. DOI: 10.1109/TNNLS.2020.2979670.

[2] S. Mohammad, S. Kiritchenko, P. Sobhani, X. Zhu, and C. Cherry, "Semeval-2016 task 6: Detecting stance in tweets," in *Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016)*, 2016, pp. 31–41.

[3] A. v. d. Oord, Y. Li, and O. Vinyals, *Representation learning with contrastive predictive coding*, 2018. DOI: 10.48550/ARXIV.1807.03748. [Online]. Available: https://arxiv.org/abs/1807.03748.

[4] A. Vaswani *et al.*, "Attention is all you need," 2017. DOI: 10.48550/ARXIV.1706.03762. [Online]. Available: https://arxiv.org/abs/1706.03762.

[5] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," 2018. DOI: 10.48550/ARXIV.1810.04805. [Online]. Available: https://arxiv.org/abs/1810.04805.

[6] Y. Zhu *et al.*, "Aligning books and movies: Towards story-like visual explanations by watching movies and reading books," 2015. DOI: 10.48550/ARXIV.1506.06724. [Online]. Available: https://arxiv.org/abs/1506.06724.

[7] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, *Distilbert, a distilled version of bert: Smaller, faster, cheaper and lighter*, 2019. DOI: 10.48550/ARXIV.1910.01108. [Online]. Available: https://arxiv.org/abs/1910.01108.

[8] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, *Albert: A lite bert for self-supervised learning of language representations*, 2019. DOI: 10.48550/ARXIV.1909.11942. [Online]. Available: https://arxiv.org/abs/1909.11942.

[9] Y. Liu *et al.*, *Roberta: A robustly optimized bert pretraining approach*, 2019. DOI: 10.48550/ARXIV.1907.11692. [Online]. Available: https://arxiv.org/abs/1907.11692.

[10] M. Ostendorff, N. Rethmeier, I. Augenstein, B. Gipp, and G. Rehm, *Neighborhood contrastive learning for scientific document representations with citation embeddings*, 2022. DOI: 10.48550/ARXIV.2202.06671. [Online]. Available: https://arxiv.org/abs/2202.06671.

[11] S. Mohammad, S. Kiritchenko, P. Sobhani, X. Zhu, and C. Cherry, "Semeval-2016 task 6: Detecting stance in tweets," 2016. [Online]. Available: http://dx.doi.org/10.18653/v1/s16-1003.

[12] P. Sobhani, D. Inkpen, and X. Zhu, "A dataset for multi-target stance detection," 2017. [Online]. Available: http://dx.doi.org/10.18653/v1/e17-2088.

[13] K. Darwish, W. Magdy, and T. Zanouda, "Improved stance prediction in a user similarity feature space," 2017. [Online]. Available: http://dx.doi.org/10.1145/3110025.3110112.

[14] N. Landwehr, M. Hall, and E. Frank, "Logistic model trees," *Machine Learning*, vol. 59, no. 1, 161–205, 2005, ISSN: 1573-0565. DOI: 10.1007/s10994-005-0466-3.

[15] W. Ferreira and A. Vlachos, "Emergent: A novel data-set for stance classification," 2016. [Online]. Available: http://dx.doi.org/10.18653/v1/n16-1138.

[16] M. Matero, N. Soni, N. Balasubramanian, and H. A. Schwartz, "Melt: Message-level transformer with masked document representations as pre-training for stance detection," 2021. [Online]. Available: http://dx.doi.org/10.18653/v1/2021.findings-emnlp.253.

[17] M. Hardalov, A. Arora, P. Nakov, and I. Augenstein, "Cross-domain label-adaptive stance detection," 2021. [Online]. Available: http://dx.doi.org/10.18653/v1/2021.emnlp-main.710.

[18] K. Joseph *et al.*, "(mis)alignment between stance expressed in social media data and public opinion surveys," pp. 312–324, Nov. 2021. DOI: 10.18653/v1/2021.emnlp-main.27. [Online]. Available: https://aclanthology.org/2021.emnlp-main.27.

[19] Z. Zhang, J. Li, F. Fukumoto, and Y. Ye, "Abstract, rationale, stance: A joint model for scientific claim verification," 2021. [Online]. Available: http://dx.doi.org/10.18653/v1/2021.emnlp-main.290.

[20] S. Jayaram and E. Allaway, "Human rationales as attribution priors for explainable stance detection," 2021. [Online]. Available: http://dx.doi.org/10.18653/v1/2021.emnlp-main.450.

[21] S. Yang and J. Urbani, "Tribrid: Stance classification with neural inconsistency detection," 2021. [Online]. Available: http://dx.doi.org/10.18653/v1/2021.emnlp-main.547.

[22] A. Aldayel and W. Magdy, "Your stance is exposed! analysing possible factors for stance detection on social media," *Proceedings of the ACM on Human-Computer Interaction*, vol. 3, no. CSCW, 1–20, 2019. DOI: 10.1145/3359307.

[23] A. Rashed, M. Kutlu, K. Darwish, T. Elsayed, and C. Bayrak, "Embeddings-based clustering for target specific stances: The case of a polarized turkey," May 2020.

[24] P. Wei, J. Lin, and W. Mao, "Multi-target stance detection via a dynamic memory-augmented network," 2018. [Online]. Available: http://dx.doi.org/10.1145/3209978.3210145.

[25] Y. Li, C. Zhao, and C. Caragea, "Improving stance detection with multi-dataset learning and knowledge distillation," 2021. [Online]. Available: http://dx.doi.org/10.18653/v1/2021.emnlp-main.511.

[26] S. Mohammad, S. Kiritchenko, P. Sobhani, X. Zhu, and C. Cherry, "A dataset for detecting stance in tweets," in *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, Portorož, Slovenia: European Language Resources Association (ELRA), May 2016, pp. 3945–3952. [Online]. Available: https://aclanthology.org/L16-1623.

[27] B. Schiller, J. Daxenberger, and I. Gurevych, "Stance detection benchmark: How robust is your stance detection?" *KI - Künstliche Intelligenz*, vol. 35, no. 3–4, 329–341, 2021. DOI: 10.1007/s13218-021-00714-w.

[28] G. Gorrell *et al.*, "SemEval-2019 task 7: RumourEval, determining rumour veracity and support for rumours," in *Proceedings of the 13th International Workshop on Semantic Evaluation*, Minneapolis, Minnesota, USA: Association for Computational Linguistics, Jun. 2019, pp. 845–854. DOI: 10.18653/v1/S19-2147. [Online]. Available: https://aclanthology.org/S19-2147.

[29] C. Conforti, J. Berndt, M. T. Pilehvar, C. Giannitsarou, F. Toxvaerd, and N. Collier, "Will-they-won't-they: A very large dataset for stance detection on twitter," 2020. [Online]. Available: http://dx.doi.org/10.18653/v1/2020.acl-main.157.

[30] A. Baheti, M. Sap, A. Ritter, and M. Riedl, "Just say no: Analyzing the stance of neural dialogue generation in offensive contexts," 2021. [Online]. Available: http://dx.doi.org/10.18653/v1/2021.emnlp-main.397.

[31] R. Dong, Y. Sun, L. Wang, Y. Gu, and Y. Zhong, "Weakly-guided user stance prediction via joint modeling of content and social interaction," 2017. [Online]. Available: http://dx.doi.org/10.1145/3132847.3133020.

[32] K. Darwish, P. Stefanov, M. Aupetit, and P. Nakov, "Unsupervised user stance detection on twitter," Apr. 2019.

[33] S. Sun and W. Li, "Alleviating exposure bias via contrastive learning for abstractive text summarization," Aug. 2021.

[34] Y. Shao, S. Gouws, D. Britz, A. Goldie, B. Strope, and R. Kurzweil, "Generating high-quality and informative conversation responses with sequence-to-sequence models," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, Copenhagen, Denmark: Association for Computational Linguistics, Sep. 2017, pp. 2210–2219. DOI: 10.18653/v1/D17-1235. [Online]. Available: https://aclanthology.org/D17-1235.

[35] S. Cao and L. Wang, "Cliff: Contrastive learning for improving faithfulness and factuality in abstractive summarization," 2021. [Online]. Available: http://dx.doi.org/10.18653/v1/2021.emnlp-main.532.

[36] H. Wu, T. Ma, L. Wu, T. Manyumwa, and S. Ji, "Unsupervised reference-free summary quality evaluation via contrastive learning," 2020. [Online]. Available: http://dx.doi.org/10.18653/v1/2020.emnlp-main.294.

[37] Z. Yue, B. Kratzwald, and S. Feuerriegel, "Contrastive domain adaptation for question answering using limited text corpora," 2021. [Online]. Available: http://dx.doi.org/10.18653/v1/2021.emnlp-main.754.

[38] H. Xu *et al.*, "Videoclip: Contrastive pre-training for zero-shot video-text understanding," 2021. [Online]. Available: http://dx.doi.org/10.18653/v1/2021.emnlp-main.544.

[39] C. Liu, R. Wang, J. Liu, J. Sun, F. Huang, and L. Si, "Dialoguecse: Dialogue-based contrastive learning of sentence embeddings," 2021. [Online]. Available: http://dx.doi.org/10.18653/v1/2021.emnlp-main.185.

[40] J. Zhang *et al.*, "Few-shot intent detection via contrastive pre-training and fine-tuning," 2021. [Online]. Available: http://dx.doi.org/10.18653/v1/2021.emnlp-main.144.

[41] Z. Wu, S. Wang, J. Gu, M. Khabsa, F. Sun, and H. Ma, "Clear: Contrastive learning for sentence representation," Dec. 2020.

[42] Y. Du *et al.*, "Constructing contrastive samples via summarization for text classification with limited annotations," 2021. [Online]. Available: http://dx.doi.org/10.18653/v1/2021.findings-emnlp.118.

[43] X. Dong, A. T. Luu, M. Lin, S. Yan, and H. Zhang, "How should pre-trained language models be fine-tuned towards adversarial robustness?" In *Advances in Neural Information Processing Systems*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P. S. Liang, and J. W. Vaughan, Eds., vol. 34, Curran Associates, Inc., 2021, 4356–4369. [Online]. Available: https://proceedings.neurips.cc/paper/2021/file/22b1f2e0983160db6f7bb9f62f4dbb39-Paper.pdf.

[44] C. Zhang, X. Zhou, Y. Wan, X. Zheng, K.-W. Chang, and C.-J. Hsieh, "Improving the adversarial robustness of NLP models by information bottleneck," in *Findings of the Association for Computational Linguistics: ACL 2022*, Dublin, Ireland: Association for Computational Linguistics, May 2022, pp. 3588–3598. DOI: 10.18653/v1/2022.findings-acl.284. [Online]. Available: https://aclanthology.org/2022.findings-acl.284.

[45] B. Wang *et al.*, *Infobert: Improving robustness of language models from an information theoretic perspective*, 2020. DOI: 10.48550/ARXIV.2010.02329. [Online]. Available: https://arxiv.org/abs/2010.02329.

[46] M. Moradi and M. Samwald, "Evaluating the robustness of neural language models to input perturbations," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, Nov. 2021, pp. 1558–1570. DOI: 10.18653/v1/2021.emnlp-main.117. [Online]. Available: https://aclanthology.org/2021.emnlp-main.117.

[47] B. Alshemali and J. Kalita, "Improving the reliability of deep neural networks in nlp: A review," *Knowledge-Based Systems*, vol. 191, p. 105 210, 2020, ISSN: 0950-7051. DOI: https://doi.org/10.1016/j.knosys.2019.105210.

[48] J. Thorne, A. Vlachos, C. Christodoulopoulos, and A. Mittal, "Evaluating adversarial attacks against multiple fact verification systems," 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)*, Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 2944–2953. DOI: 10.18653/v1/D19-1292. [Online]. Available: https://aclanthology.org/D19-1292.

[49] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "Distilbert, a distilled version of bert: Smaller, faster, cheaper and lighter," *ArXiv*, vol. abs/1910.01108, 2019.

[50] A. Gokaslan and V. Cohen, *Openwebtext corpus*.

[51] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," *CoRR*, vol. abs/1810.04805, 2018. arXiv: 1810.04805. [Online]. Available: http://arxiv.org/abs/1810.04805.

[52] J. Li *et al.*, "Unsupervised belief representation learning with information-theoretic variational graph auto-encoders," 2022. DOI: 10.1145/3477495.3532072. [Online]. Available: https://doi.org/10.1145%2F3477495.3532072.

[53] K. S. Hasan and V. Ng, "Stance classification of ideological debates: Data, models, features, and constraints," in *Proceedings of the sixth international joint conference on natural language processing*, 2013, pp. 1348–1356.

[54] Q. Sun, X. Xi, J. Sun, Z. Wang, and H. Xu, "Stance detection with a multi-target adversarial attention network," *Transactions on Asian and Low-Resource Language Information Processing*, 2022.

[55] I. Habernal, H. Wachsmuth, I. Gurevych, and B. Stein, "The argument reasoning comprehension task: Identification and reconstruction of implicit warrants," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 2018, pp. 1930–1940.

[56] S. Chen, D. Khashabi, W. Yin, C. Callison-Burch, and D. Roth, *Seeing things from a different angle: Discovering diverse perspectives about claims*, 2019. DOI: 10.48550/ARXIV.1906.03538. [Online]. Available: https://arxiv.org/abs/1906.03538.

[57] D. Pomerleau and D. Rao, *Exploring how artificial intelligence technologies could be leveraged to combat fake news.* [Online]. Available: http://www.fakenewschallenge.org/.

[58] K. Kawintiranon and L. Singh, “Knowledge enhanced masked language model for stance detection,” in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Online: Association for Computational Linguistics, Jun. 2021, pp. 4725–4735. DOI: 10.18653/v1/2021.naacl-main.376. [Online]. Available: https://aclanthology.org/2021.naacl-main.376.

[59] J. Giorgi, O. Nitski, B. Wang, and G. Bader, *Declutr: Deep contrastive learning for unsupervised textual representations*, 2020. DOI: 10.48550/ARXIV.2006.03659. [Online]. Available: https://arxiv.org/abs/2006.03659.

[60] *Allennlp - allen institute for ai.* [Online]. Available: https://allenai.org/allennlp.

[61] Twitter, *Twitter. it's what's happening.* 2022. [Online]. Available: https://twitter.com/?lang=en.

[62] J. P. Kincaid, R. P. Fishburne Jr, R. L. Rogers, and B. S. Chissom, “Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel,” Naval Technical Training Command Millington TN Research Branch, Tech. Rep., 1975.