Detecting Textual Adversarial Examples Based on Distributional Characteristics of Data Representations

Na Liu[1] **Mark Dras**[1] **Wei Emma Zhang**[2]
[1] School of Computing, Macquarie University
[2] School of Computer Science, The University of Adelaide
na.liu8@students.mq.edu.au,mark.dras@mq.edu.au
wei.e.zhang@adelaide.edu.au

Abstract

Although deep neural networks have achieved state-of-the-art performance in various machine learning tasks, adversarial examples, constructed by adding small non-random perturbations to correctly classified inputs, successfully fool highly expressive deep classifiers into incorrect predictions. Approaches to adversarial attacks in natural language tasks have boomed in the last five years using character-level, word-level, phrase-level, or sentence-level textual perturbations. While there is some work in NLP on defending against such attacks through proactive methods, like adversarial training, there is to our knowledge no effective general reactive approaches to defence via detection of textual adversarial examples such as is found in the image processing literature. In this paper, we propose two new reactive methods for NLP to fill this gap, which unlike the few limited application baselines from NLP are based entirely on distribution characteristics of learned representations: we adapt one from the image processing literature (Local Intrinsic Dimensionality (LID)), and propose a novel one (MultiDistance Representation Ensemble Method (MDRE)). Adapted LID and MDRE obtain state-of-the-art results on character-level, word-level, and phrase-level attacks on the IMDB dataset as well as on the later two with respect to the MultiNLI dataset. For future research, we publish our code [1].

1 Introduction

Highly expressive deep neural networks are fragile against adversarial examples, constructed by carefully designed small perturbations of normal examples, that can fool deep classifiers to make wrong predictions (Szegedy et al., 2013). Crafting adversarial examples in images involves adding small non-random perturbations to many pixels in inputs that should be correctly classified by a target model. These perturbations can force high-efficacy models into incorrect classifications and are often imperceptible to humans (Szegedy et al., 2013; Goodfellow et al., 2014; Moosavi-Dezfooli et al., 2016; Papernot et al., 2016b; Carlini and Wagner, 2017b; Chen et al., 2018). However, when adversarial examples have been studied in the context of text, to our knowledge, only Miyato et al. (2016) aligns closely with the original intuition of adversarial examples in applying perturbations to word embeddings, which are inputs of deep neural nets. Rather, most adversarial attack techniques use more practical semantics-preserving textual changes other than embedding perturbations, at character-level, word-level, phrase-level, or sentence-level (Pruthi et al., 2019; Jia and Liang, 2017; Alzantot et al., 2018; Ribeiro et al., 2018; Ren et al., 2019; Iyyer et al., 2018; Yoo and Qi, 2021; Li et al., 2020, 2021; Jin et al., 2020); see Table 1. This variety increases the difficulty of detecting textual adversarial examples.

Generating adversarial examples to attack deep neural nets and protecting deep neural nets from adversarial examples have been extensively studied in image classification tasks (Szegedy et al., 2013; Goodfellow et al., 2014; Moosavi-Dezfooli et al., 2016; Papernot et al., 2016b; Carlini and Wagner, 2017b; Chen et al., 2018; Papernot et al., 2016a; Feinman et al., 2017; Ma et al., 2018; Lee et al., 2018). However, in the natural language domain, only crafting of adversarial examples has been comprehensively considered (Jia and Liang, 2017; Alzantot et al., 2018; Ribeiro et al., 2018; Ren et al., 2019; Iyyer et al., 2018). Defence against textual adversaries, primarily through increasing the robustness of deep neural networks, is much less studied (Jia et al., 2019; Pruthi et al., 2019). In the image processing space, Cohen et al. (2020) refers to these as *proactive* defence methods, and Carlini and Wagner (2017a) notes that they can be evaded by optimization-based attacks, such as constructing new loss functions; in the NLP space,

[1] Code available at https://github.com/NaLiuAnna/MDRE

Proceedings of the 7th Workshop on Representation Learning for NLP, pages 78 - 90
May 26, 2022 ©2022 Association for Computational Linguistics

	Example	Prediction
Original	**This** is a **story** of two **misfits who** don't **stand** a **chance alone**, but **together** they are **magnificent**.	Positive
Character-level (Pruthi et al., 2019)	**TZyTis** is a **sotry** of two **misifts** who don't **stad** a **ccange alUone**, but **tpgthr** they are **mgnificent**.	Negative
Word-level (Alzantot et al., 2018)	This is a **conte** of two **who** don't **stands** a **opportunities** alone, but together they are **opulent**.	Negative
Phrase-level (Iyyer et al., 2018)	Why don't you have two misfits who don't stand a chance alone, but together they're beautiful.	Negative
Sentence-level (Jia and Liang, 2017)	This is a story of two misfits who don't stand a chance alone, but together they are magnificent. **ready south hundred at size expected worked whose turn poor.**	Negative

Table 1: Examples of textual adversarial instances on a sentiment analysis task

Yoo and Qi (2021) observes that generating word-level textual adversaries for proactive adversarial training are computationally expensive because of necessary search and constraints based on sentence encoding. Consequently, Feinman et al. (2017); Ma et al. (2018); Lee et al. (2018); Papernot and McDaniel (2018) explore *reactive* defence methods (Cohen et al., 2020) in the image processing space: these focus on distinguishing real from adversarial examples, in order to detect them before they are passed to neural networks. These reactive defences have been explored in only a limited way in the NLP space (Mozes et al., 2021). Importantly, these few methods rely on procedures like testing word substitutions, quite unlike those in the image processing space, which are functions of the learned representations.

The contributions of this paper are two textual adversarial reactive detectors as follows:

- Adapting the Local Intrinsic Dimensionality (LID) method from image processing to the text domain.
- Proposing a MultiDistance Representation Ensemble Method (MDRE).

Both of them are based on distribution differences of semantic representations between normal examples and adversarial examples. They achieve state-of-the-art results across a range of attack methods and domains.

2 Related Work

In this section, we briefly review state-of-the-art work on defending neural networks against both image and textual adversarial examples.

Image Adversarial Defences: Adversarial training (Goodfellow et al., 2014) using adversarial examples to augment training data or adding an adversarial objective to a loss function, and defensive distillation framework (Papernot et al., 2016a) which transfers knowledge between same structured teacher and student models, are two effective proactive defence methods. For reactive defences, Feinman et al. (2017); Ma et al. (2018); Papernot and McDaniel (2018); Lee et al. (2018) have all proposed approaches that use the learned representations of the classifier that the attacker is trying to fool, and then with a variety of techniques to identify characteristics of the adversarial examples' learned representations that permit the detection of whether a data point is adversarial or original; these techniques involve kernel density estimations in a feature space of a last hidden layer and Bayesian uncertainty estimates, Local Intrinsic Dimensionality, Deep k-Nearest Neighbors, and Mahalanobis distance-based confidence scores respectively.

Textual Adversarial Defences: Adversarial training (Goodfellow et al., 2014) is a commonly used defence method to augment training data with adversarial examples and their correct labels, which has been effective in Li et al. (2016), Li et al. (2017), Ribeiro et al. (2018), and Ebrahimi et al. (2018), but has limited utility in Pruthi et al. (2019) and Jia and Liang (2017). Jia et al. (2019) applies interval bound propagation (IBP) to minimize an upper bound of possible candidate sentences' losses when facing word substitution adversaries. Jones et al. (2020) introduced robust encodings (RobEn) to cluster words and typos, and produced one encoding for each cluster to harness adversarial typos. Zhou et al. (2019) proposed the learning to discriminate perturbations (DISP) framework to block character-level and word-level adversarial perturbations by recognising and replacing perturbed words. Mozes et al. (2021) noticed and verified a characteristic of word-level adversaries that replacement words are less likely to occur than their substitutions, therefore, they constructed a rule-based, model-agnostic frequency-guided word substitutions (FGWS) algorithm, which is the only existing textual reactive defence method as far as we know.

3 Methods

3.1 Adapted Local Intrinsic Dimensionality (LID)

From among the reactive image processing methods, we selected the Local Intrinsic Dimensionality (LID) approach of Ma et al. (2018) as one that can be directly adapted to textual representations. The approach of Ma et al. (2018) uses LID to reveal the local distance distribution for a reference point representation to its neighbours, and uses outputs of each layer from the target deep neural network as an input point representations. LID was initially presented for dimension reduction (Houle et al., 2012). Ma et al. (2018) introduced LID to characterize the local data submanifolds in the vicinity of reference points and detect adversarial samples from their originals. The LID definition is as follows.

Definition 3.1 (Local Intrinsic Dimensionality (Ma et al., 2018))**.** Given a data sample $x \in X$, let $R > 0$ be a random variable denoting the distance from x to other data samples. If the cumulative distibution function $F(r)$ of R is positive and continuously differentiable at distance $r > 0$, the LID of x at distance r is given by:

$$LID_F(r) \triangleq \lim_{\epsilon \to 0} \frac{ln(F((1+\epsilon) \cdot r)/F(r))}{ln(1+\epsilon)} = \frac{r \cdot F'(r)}{F(r)} \tag{1}$$

whenever the limit exists.

To simplify computation, given a reference sample $x \sim \mathcal{P}$, where $\mathcal{P}$ represents the data distribution, the Maximum Likelihood Estimator of the LID at x is defined as follows (Ma et al., 2018):

$$\widehat{LID}(x) = -\left(\frac{1}{k} \sum_{i=1}^{k} \log \frac{r_i(x)}{r_k(x)} \right)^{-1} \tag{2}$$

where $r_i(x)$ is the distance between x and its ith nearest neighbor within a sample of points drawn from $\mathcal{P}$, k is the number of nearest neighbors. Since the logarithmic function $f(x) = \log_a(x)$ for any base a and the negative reciprocal function $f(x) = -x^{-1}$ are monotonically increasing functions when their independent variables are positive, if neighbors of a reference sample x are compact, its estimated LID from Equation (2) is smaller, otherwise, its estimated LID is bigger.

When building a binary classifier to detect adversarial examples using LID in Ma et al. (2018), the inputs are lists of estimated LID from the Equation (2) of different layers' outputs from the target deep neural net, and adversarial and normal examples are two categories of the classifier.

To adapt this to textual representations, we implement same technique — a detection classifier based on LID characterizations derived from different layers' outputs of a deep neural net — but apply this to a Transformer. Here we use $\text{BERT}_{\text{BASE}}$ model (Devlin et al., 2019), although in principle any would be suitable. The x in the Equation (2) is a representation of an input text from a layer's hidden state of the first token of the target ($\text{BERT}_{\text{BASE}}$) model, since self-attention layers are essential modules of a transformer, and the last layer hidden state of the first token is typically used as a component to build a pooled output, a text representation for a classifier. Therefore, an input of a detection classifier for an example is a 12-dimensional vector, where each element illustrates the corresponding layer's estimated LID from the $\text{BERT}_{\text{BASE}}$ model.

3.2 MultiDistance Representation Ensemble Method (MDRE)

Adversarial examples are constructed by adding imperceptible non-random perturbations to inputs of correctly classified test examples to fool highly expressive deep neural nets into incorrect classifications (Szegedy et al., 2013). Motivated by the reasoning behind LID expressed in Equation (2), by Feinman et al. (2017)'s intuition that adversarial samples lie off the true data manifold, and by (Lee et al., 2018)'s recognition that they are out-of-distribution samples by a class-conditional distribution, we assume that samples with a same predicted label from a deep neural net lie on a data submanifold; an adversarial example is generated because perturbations cause a correctly predicted example to transfer from one data submanifold to another, making it an out-of-distribution sample relative to training examples from its data submanifold. Consequently, we posit that it is likely that the Euclidean distance between an adversarial example x' and the nearest neighbor of x' among training examples with the same predicted label as x' is bigger than the Euclidean distance between its corresponding original normal test example x and x's nearest neighbor among training examples with the same predicted label as x.

In natural language processing, most inputs of deep neural networks are learned representations by representation learning models nowadays. Even though current methods of representation learn-

Algorithm 1 MultiDistance Representation Ensemble Method (MDRE)

Input:

$\mathbb{D} = \{\boldsymbol{X}^{(train)}, \boldsymbol{X}^{(norm)}, \boldsymbol{X}^{(adv)}\}$: a dataset; there are k examples in $\boldsymbol{X}^{(norm)}$ and $\boldsymbol{X}^{(adv)}$

H: an array containing m representation learning models

$g : \mathbb{R}^m \rightarrow \{0, 1\}$: a binary classification model (MDRE)

$f : \mathbb{R}^n \rightarrow \mathbb{R}^l$: a deep neural net that is the target model for an adversarial attack

Output:

Detection accuracy of MDRE: acc

Initializing inputs and labels of g: $\boldsymbol{X} = zeros[2k, m]$, $\boldsymbol{y} = zeros[2k]$
Computing examples' predictions from f of $\mathbb{D}$: $\{\hat{\boldsymbol{y}}^{(train)}, \hat{\boldsymbol{y}}^{(norm)}, \hat{\boldsymbol{y}}^{(adv)}\}$
for $j \in \{0, \cdots, m-1\}$ **do**
    Computing examples' representations from $H[j]$ of $\mathbb{D}$: $\{\boldsymbol{V}_j^{(train)}, \boldsymbol{V}_j^{(norm)}, \boldsymbol{V}_j^{(adv)}\}$
    for $i \in \{0, \cdots, k-1\}$ **do**
        Calculating $d_j^{(norm)}, d_j^{(adv)}$ for examples $\boldsymbol{X}_i^{(norm)}, \boldsymbol{X}_i^{(adv)}$
        $\boldsymbol{X}[i, j] = d_j^{(norm)}$, $\boldsymbol{y}[i] = 0$
        $\boldsymbol{X}[k+i, j] = d_j^{(adv)}$, $\boldsymbol{y}[k+i] = 1$
    end for
end for
Training g by randomly choosing 80% of $\{(\boldsymbol{X}_{i,:}, \boldsymbol{y}_i)\}_{i=0}^{2k-1}$
acc = test accuracy of g using the rest 20% of $\{(\boldsymbol{X}_{i,:}, \boldsymbol{y}_i)\}_{i=0}^{2k-1}$

ing are effective in various tasks (Devlin et al., 2019; Liu et al., 2019; Yang et al., 2019; Lewis et al., 2020), semantic meanings and semantic differences between texts from humans' perspective are not perfectly captured by textual representation vectors (Liu et al., 2020). In addition, as mentioned in Section 1, most textual adversarial generation algorithms do not modify representations, which are input feature vectors of deep neural networks, but modify original texts. Therefore, the assumed characteristic of adversaries in the last paragraph that the Euclidean distances between adversarial examples and their nearest neighbors among training examples in the same submanifolds are bigger than normal examples, may lose efficiency in textual adversarial detection scenarios. To build a stronger reactive classifier, we use ensemble learning to combine distances between representations learned from multiple representation learning models. We construct a more effective MultiDistance Representation Ensemble Method (MDRE), as illustrated in Algorithm 1.

The MDRE is a supervised binary classification model $g : \mathbb{R}^m \rightarrow \{0, 1\}$. m is the number of representation learning models; g can be any binary classification model, such as logistic regressions or deep neural nets; $\{0, 1\}$ is the output label set, with 1 corresponding to adversarial examples, 0 to normal examples.

The input of MDRE is a matrix $\boldsymbol{X}$ and each row vector of $\boldsymbol{X}$ is $\boldsymbol{X}_{i,:} = (d_0, d_1 \cdots, d_{m-1}) \in \mathbb{R}^m$. The element of this vector $d_j, 0 \leq j \leq m-1$ is a Euclidean distance between a semantic representation of a normal or adversarial example $\boldsymbol{v}$ and a representation of its nearest neighbour among training examples with the same predicted label as $\boldsymbol{v}$ through the j-th representation learning model $H[j]$, as $d_j^{(norm)}$ or $d_j^{(adv)}$ in Algorithm 1. To find a nearest neighbour, we compare Euclidean distances between $\boldsymbol{v}$ and all representations among training examples with the same predicted label as $\boldsymbol{v}$ through $H[j]$. In Algorithm 1, $\boldsymbol{X}^{(norm)}$ consists of normal test examples corresponding to the elements of $\boldsymbol{X}^{(adv)}$, where the elements of $\boldsymbol{X}^{(norm)}$ have correct predictions from the target model f, but $\boldsymbol{X}^{(adv)}$ elements have incorrect predictions from f. The training and testing process of MDRE is same as the process of the selected model g.

4 Evaluation

In this section, we evaluate the utilities of the adapted LID and MDRE by using character-level, word-level, and phrase-level upstream attacks on sentiment analysis and natural language inference tasks, and comparing against several baselines: a language model, DISP (Zhou et al., 2019), and

FGWS (Mozes et al., 2021). The experimental results demonstrate that the adapted LID and MDRE outperforms these methods on sentiment analysis and natural language inference tasks for word-level and phrase-level attacks.

4.1 Experimental Setup

4.1.1 Tasks

We apply our approaches and baselines to sentiment analysis and natural language inference tasks, since they are two most commonly used datasets in textual adversarial example generation. The sentiment analysis task has been the most widely used testbed for generating textual adversarial examples (Pruthi et al., 2019; Alzantot et al., 2018; Ribeiro et al., 2018; Ren et al., 2019; Iyyer et al., 2018; Yoo and Qi, 2021; Li et al., 2020), making this the natural domain for these experiments; they have also been popularly applied to the natural language inference task (Alzantot et al., 2018; Iyyer et al., 2018; Yoo and Qi, 2021; Li et al., 2020, 2021; Jin et al., 2020), so we choose this to explore the generality of our methods.

We use the IMDB dataset (Maas et al., 2011) in the sentiment analysis task, which contains 50,000 movie reviews, divided into 25,000 training examples and 25,000 test examples, labelled for positive or negative sentiment. The average number of words per review in the IMDB dataset is 262 when using the Natural Language Toolkit (NLTK) (Bird et al., 2009) to tokenize examples. We set a maximum sequence length of the IMDB dataset to 512 for all following models.

To test the robustness of our methods, the Multi-Genre NLI (MultiNLI) corpus (Williams et al., 2018) and its mismatched test examples, which are derived from sources that differ from the training examples, are used in the natural language inference task. The MultiNLI dataset includes 392,702 training examples and 9,832 mismatched testing examples in which global_label fields are not "-", with three classes: entailment, neutral, and contradiction. The average and maximum word numbers of the MultiNLI dataset are 34 and 416 respectively, using NLTK word tokenizer. We set the maximum sequence length for this dataset to 256.

4.1.2 Attack Methods

We implement three widely used attack methods using character-level, word-level, and phrase-level perturbations to construct adversarial examples. For all types of attacks, we take the $\text{BERT}_{\text{BASE}}$ model as the target model, indicating that adversaries have different predictions with their originals by the $\text{BERT}_{\text{BASE}}$ model.

Character-level. The character-level attack is from Pruthi et al. (2019), which applies swapping, dropping, adding, and keyboard mistakes to a randomly selected word of an original example.

- Swapping: swapping two adjacent internal characters.
- Dropping: removing an internal character.
- Adding: internally inserting a new character.
- Keyboard mistakes: substituting an internal character with one of its adjacent characters in keyboards.

Here, we set maximum numbers of perturbations to half of the maximum sequence lengths of datasets; consequently, for the IMDB dataset, the maximum number of attacks is 256, and for the MultiNLI dataset is 128. If after achieving this number, the prediction of the perturbed text is still consistent with the original example, these attacks fail, and no character-level adversarial example constructed for this original example.

Word-level. We use a method from Alzantot et al. (2018), which is an effective and widely cited word-level threat method. Their approach randomly selects a word in a sentence, replaces it with its synonymous and context fitted word according to the GloVe word vectors (Pennington et al., 2014), counter-fitting word vectors (Mrkšić et al., 2016), and the Google 1 billion words language model (Chelba et al., 2013), and applies population-based genetic algorithms from the natural selection using a combination of crossover and mutation to generate next adversarial generations.

While effective, the initial algorithm is somewhat inefficient and computationally expensive. In implementing this method, Jia et al. (2019) found that computing scores from the Google 1 billion words language model (Chelba et al., 2013) for each iteration in this approach causes its inefficiency; to improve this, they used a faster language model and prevented semantic drift, which is synonyms picked from previous iterations also apply the language model to select words from their neighbour lists. In our experiments, we adapt these modifications by using a faster Transformer-XL architecture (Dai et al., 2019) pretrained on the WikiText-103 dataset (Merity et al., 2016), and not allowing the semantic drift, so that we compute all test examples words' neighbours before attacks.

Dataset	Training.	Validation.	Testing.	Correctly Predicted Test Examples	Adversarial/Original Examples		
					character-level	word-level	phrase-level
IMDB	20,000	5,000	25,000	23,226	12,299	9,627	6,315
MultiNLI	314,162	78,540	9,832	8,062	7,028	3,240	4,340

Table 2: The number of examples used in experiments

In this attack, we also set maximum numbers of perturbations, which are one fifth of the maximum sequence lengths; therefore, for the IMDB dataset is 102, and for the MultiNLI dataset is 51. For an original test example, if the number of attacks reaches this threshold but predictions do not change, no corresponding adversarial example is constructed for this original example.

Phrase-level. The phrase-level attack is from Ribeiro et al. (2018), which uses translators and back translators to generate adversarial examples. As far as we know, this is the only phrase-level perturbation technique that can be used for paragraph-length text. Their approach — termed semantically equivalent adversaries (SEAs) — translates an original sentences into multiple pivot languages, then translates them back to the source language. If there is a back translated sentences that is semantically equivalent to the original sentences, measured by a semantic score greater than a threshold, and it has a different prediction with the original sentences, then it is an adversarial example. Otherwise, this original example has no relevant adversaries.

4.1.3 Target Model

The $\text{BERT}_{\text{BASE}}$ model is implemented as a target model for these three attacks, by which adversarial examples are misclassified. We apportion training sets on both datasets into training subsets and validation subsets, with an 80-20 split. After training, the models achieve 92.90% test accuracy on the IMDB dataset, and for the MultiNLI mismatched test set is 82.01%. The correctly predicted test examples are preserved for subsequent attack processes. After attacks, adversarial examples and their corresponding normal test examples maintain for following detectors as negative and positive examples; in this, we follow the experimental setup used for evaluating reactive defences in the image processing literature (Ma et al., 2018) with an 80/20 training/test split. The number of examples used on the IMDB and MultiNLI datasets and number of originals and adversaries after attacks are shown in Table 2.

4.1.4 Detection Methods

We evaluate three baselines in addition to the adapted LID and MDRE in these experiments.

A language model. The first baseline is built from a language model since even though most attack algorithms intend to construct semantically and syntactically similar adversaries, many textual adversaries are abnormal and ungrammatical, as shown in Table 1. We use the Transformer-XL model pretrained on the WikiText-103 dataset from Hugging Face transformers (Wolf et al., 2020), and obtain language model scores for texts as the product of words prediction proportion scores. We construct a detection classifier by using a logistic regression model with language model scores as inputs; the model acts to learn a threshold on scores to distinguish adversarial examples.

Learning to Discriminate Perturbations (DISP) (Zhou et al., 2019). Our second baseline is the DISP framework, which is the only comparable technique for detecting textual adversarial examples across character-level and word-level attacks to our knowledge. DISP consists of three components: perturbation discriminator, embedding estimator, and hierarchical navigable small word graphs. The perturbation discriminator identifies a set of character-level or word-level perturbed tokens; the embedding estimator predicts embeddings for each perturbed token; then, hierarchical navigable small word graphs map these embeddings to actual words to correct adversarial perturbations. DISP is not itself designed as a adversarial example detector, but we adapt it for that task: if an adversarial example rectified by DISP predicts the same class as the target model predicts for the corresponding initial original example, or the prediction of a normal (non-adversarial) example rectified by DISP isn't changed, we consider DISP to have been successful in its detection. Otherwise, it is not. Since DISP is designed for character-level and word-level attacks, we do not apply it to phrase-level attacks.

Frequency-guided word substitutions (FGWS) (Mozes et al., 2021). Our third baseline is FGWS. Mozes et al. (2021) noticed, and

Dataset	Attack Method	$\text{BERT}_{\text{BASE}}$	$\text{RoBERTa}_{\text{BASE}}$	$\text{XLNet}_{\text{BASE}}$	$\text{BART}_{\text{BASE}}$
IMDB	Character-level	0.3656	0.8613	0.5770	0.8286
	Word-level	0.6999	0.8714	0.7918	0.8425
	Phrase-level	0.1827	0.3224	0.3289	0.3010
MultiNLI	Character-level	0.4848	0.7104	0.6670	0.6457
	Word-level	0.6864	0.7068	0.6870	0.6296
	Phrase-level	0.2795	0.3899	0.3698	0.3325

Table 3: The accuracy of adversarial examples

verified using hypothesis testing, that a characteristic of word-level adversaries was that replacement words are less likely to occur than their substitutions. They use this feature to construct a rule-based, model-agnostic frequency-guided word substitutions (FGWS) algorithm which distinguishes adversarial examples by replacing infrequent words with their higher frequency synonyms. If the replacements cause prediction confidence changes exceeding a threshold, these examples are deemed adversarial examples. FGWS is only designed to be applied to word-level attacks. They use WordNet (Fellbaum, 2005) and GloVe vectors (Pennington et al., 2014) to find neighbors of a word. A word frequency is its number of occurrences in the corresponding dataset's training examples; infrequent words are defined as those words whose frequencies are lower than a threshold. They set this threshold to be the frequency of the word at the $\{0$-th, 10-th, $\cdots$, 100-th$\}$ percentile of word frequencies in training set. If the prediction confidence differences between sequences with replaced words and their corresponding original sequences are higher than a threshold, the original sequences are assumed to be adversarial examples. They set this threshold to the 90%-th confidence difference between words substituted validation set and original validation set in their experiment.

Adapted Local Intrinsic Dimensionality (LID) Following the characterization of our adapted LID from Section 3.1, we use the $\text{BERT}_{\text{BASE}}$ model as in the above baselines. We implement a logistic regression model as the detection classifier as Ma et al. (2018), and the neighborhood size k is tuned using a grid search over 100, 1000, and the range [10, 42) with a step size 2.

MultiDistance Representation Ensemble Method (MDRE). In MDRE, we set $m = 4$, $H = [\text{BERT}_{\text{BASE}}, \text{RoBERTa}_{\text{BASE}}, \text{XLNet}_{\text{BASE}}, \text{BART}_{\text{BASE}}]$, and g is a logistic regression model. See Algorithm 1 for more information of notations.

4.2 Experimental Results

Before discussing the effectiveness of the detection classifiers, Table 4 and Table 3 show the accuracy of the sentiment analysis and natural language inference classifiers on normal and adversarial examples from four models with three types of attacks. The $\text{BERT}_{\text{BASE}}$ model is the target model in terms of generating all kinds of adversaries — that is, the adversarial examples are specifically designed to defeat the $\text{BERT}_{\text{BASE}}$ model — so all adversarial instance predictions are incorrect, therefore, the accuracy is 0. However, when we use a different random seed which also modify the order of training examples to fine-tune another $\text{BERT}_{\text{BASE}}$ model used for prediction, its parameters is different from the parameters of the $\text{BERT}_{\text{BASE}}$ model used before. The accuracy of adversaries slightly increases, indicating that BERT model parameters do not converge but fluctuate when using stochastic or mini-batch gradient descent.

Results for detection method accuracy are in Table 5. Adapted LID and MDRE work better than the baselines, except for DISP against character-level attacks on MultiNLI dataset, where the adapted LID is a close second. The detection accuracy on the MultiNLI dataset is lower than the IMDB dataset, although this is not a surprise. It uses the mismatched test set of the MultiNLI dataset which makes the task more challenging. The results show that the adapted LID and MDRE are sensitive to sample distributions, so if some normal test examples representations are from a different distribution of training samples representations, such as noise examples, they will influence their performance.

Adapted LID is often close to MDRE. It is higher

Dataset	$\text{BERT}_{\text{BASE}}$	$\text{RoBERTa}_{\text{BASE}}$	$\text{XLNet}_{\text{BASE}}$	$\text{BART}_{\text{BASE}}$
IMDB	0.9290	0.9532	0.9336	0.9429
MultiNLI	0.8201	0.8671	0.8630	0.8455

Table 4: The accuracy of normal test examples

Dataset	Detecting Method	Character-level Attack	Word-level Attack	Phrase-level Attack
IMDB	Language Model	0.4996	0.4966	0.4838
	DISP	0.8936	0.7714	—
	FGWS	—	0.7958	—
	LID	0.9142	**0.8406**	0.9093
	MDRE	**0.9193**	0.7562	**0.9505**
MultiNLI	Language Model	0.4932	0.4707	0.4997
	DISP	**0.7496**	0.6137	—
	FGWS	—	0.6128	—
	LID	0.7328	0.5849	0.6146
	MDRE	0.7016	**0.6319**	**0.6809**

Table 5: The accuracy for detection classifiers

in word-level attack on the IMDB dataset and character-level attack on the MultiNLI dataset, but it is lower on phrase-level attacks. Relative to its initial application on image classification tasks, the performance of the adapted LID approach is worse. Most accuracy of LID on image adversarial attacks on CIFAR-10, CIFAR-100, and SVHN datasets are over or near 90% (Cohen et al., 2020). However, in our experiments, the average accuracy of the adapted LID is about 77% (against majority class baseline of 50%). This reveals the difficulty of detecting textual adversarial examples.

The performance of the language model is similar to random guess, since the ratio between positive (normal) and negative (adversarial) examples is 1:1. We observed that language model prediction proportion scores are sensitive to the number of words in examples because each word scores is between 0 to 1 and more words leads to lower scores. In addition, in some contexts, scores for synonyms or typos which are out-of-dictionary words, are lower but close to scores of original words, which do not have the large differences that might be expected.

DISP effectively applies the bidirectional language model feature of the $\text{BERT}_{\text{BASE}}$ model and builds a powerful perturbation discriminator, which labels character-level or word-level perturbed tokens to 1, and unperturbed tokens to 0. The perturbation discriminator achieves F_1 scores of 95.06% on the IMDB dataset and 97.67% on the MultiNLI dataset, using their own adversaral attack methods. However, the embedding estimator predicts embeddings through inputting 5-grams with masked middle tokens to a $\text{BERT}_{\text{BASE}}$ model with one layer feed-forward head on top and outputting embeddings of these masked tokens from 300-dimensional pretrained FastText English word vectors (Mikolov et al., 2018). This is challenging and restricts the overall performance of DISP.

Intuitively, adversaries' predictions are different from their original counterparts, which are ordinary language; therefore, adversaries may contain rare and infrequent words. According to an English word frequency dataset,[2] some words frequencies in examples of Alzantot et al. (2018) are shown in Table 6. We can find that the intuition is correct

org.	**org. freq.**	**sub.**	**sub. freq.**
terrible	8,610,277	horrific	1,017,211
		horrifying	491,916
considered	57,378,298	regarded	6,892,622
kids	96,602,880	youngstars	—
runner	7,381,022	racer	3,625,077
battling	1,340,424	—	—
strives	1,415,683	—	—

Table 6: Original and modified sample words frequencies in examples of Alzantot et al. (2018)

that replacement words frequencies drop compared with substitutions; however, they may be higher than other normal words. Therefore, using one threshold makes it difficult to separate adversarially substituted words from all normal words. Alternative approaches to applying the characteristic of adversarial words frequencies may work better. We note that it is perhaps surprising, then, that our representation-based detection methods outperform FGWS that do incorporate frequency information from the raw text input. This underscores the usefulness of the distributional information available in the learned representations.

We show detection methods applied to examples from the MultiNLI dataset in the Appendix A supplement.

4.3 Ablation Analysis of MDRE

The key ideas behind MDRE is that (1) adversarial examples are out-of-distribution samples rela-

[2]The english word frequency: https://www.kaggle.com/rtatman/english-word-frequency

Dataset	Detecting Method	Character-level Attack	Word-level Attack	Phrase-level Attack
IMDB	$\text{MDRE}_{\text{BERT}}$	0.8941	0.7541	0.9129
	$\text{MDRE}_{\text{RoBERTa}}$	0.8606	0.6645	0.9287
	$\text{MDRE}_{\text{XLNet}}$	0.7226	0.5962	0.7819
	$\text{MDRE}_{\text{BART}}$	0.8951	0.6858	0.9327
MultiNLI	$\text{MDRE}_{\text{BERT}}$	0.6102	0.5903	0.6382
	$\text{MDRE}_{\text{RoBERTa}}$	0.6853	0.5903	0.6526
	$\text{MDRE}_{\text{XLNet}}$	0.6323	0.6227	0.6452
	$\text{MDRE}_{\text{BART}}$	0.6824	0.6366	0.6740

Table 7: The accuracy of detection classifiers for ablation analysis of MDRE

tive to training examples from their data submanifolds and (2) ensemble learning can help identify this. Therefore, we combine four representation learning models: $\text{BERT}_{\text{BASE}}$, $\text{RoBERTa}_{\text{BASE}}$, $\text{XLNet}_{\text{BASE}}$, and $\text{BART}_{\text{BASE}}$ to produce MDRE as described in Section 4.1.4. In order to explore the effects of these two components and each representation learning model, we apply $\text{MDRE}_{\text{BERT}}$, $\text{MDRE}_{\text{RoBERTa}}$, $\text{MDRE}_{\text{XLNet}}$, $\text{MDRE}_{\text{BART}}$ models, where $m = 1$, $H = [\text{BERT}_{\text{BASE}}]$, $[\text{RoBERTa}_{\text{BASE}}]$, $[\text{XLNet}_{\text{BASE}}]$, and $[\text{BART}_{\text{BASE}}]$ respectively.

The results are shown in Table 7 which reveals all models work in detecting textual adversarial examples: the detection accuracy on both the IMDB and MultiNLI datasets, and all upstream adversarial attacks is substantially higher than random guess (50%). Comparing with the results of MDRE on the IMDB and MultiNLI datasets from Table 5, ensemble learning helps to build a stronger detector except word-level attack on the MultiNLI dataset.

5 Conclusion and Future work

In this paper, we adapted Local Intrinsic Dimensionality (LID) method (Ma et al., 2018) from image processing and proposed a simple and general textual adversarial reactive detector, MultiDistance Representation Ensemble Method (MDRE), based on the distribution characteristics of adversarial examples representations, that they are out-of-distribution samples and lie off the true data manifold. The experimental results show adapted LID and MDRE achieve state-of-the-art results on detecting character-level, word-level, and phrase-level adversaries on the IMDB dataset as well as on the later two with respect to the MultiNLI dataset. The results show that it is possible to construct adversarial example detectors using only the learned representations, and not relying on various textual substitution processes as in the baselines.

As discussed in Section 3, adapted LID uses estimated Local Intrinsic Dimensionality on text representations form different layers outputs of a target model, and MDRE is implemented on Euclidean distances between samples' representations and representations of their nearest neighbors among the training examples with the same predicted labels from different representation learning models, to characterise representation distribution differences between adversarial examples and normal examples. In terms of future work and the LID approach, Athalye et al. (2018) found that in the image processing space, LID is vulnerable to their Backward Pass Differentiable Approximiation (BPDA) attack; it would be useful to investigate whether this is the case in the text space, and if so, other detection methods from image processing may be worth looking into. With respect to MDRE, as it is a kind of nearest-neighbour ensembling approach, looking into other possibilities falling within that space could be productive. More generally, exploring more effective distribution characteristics of data semantic representations among adversarial and normal examples, may help to build better detectors.

References

Moustafa Alzantot, Yash Sharma, Ahmed Elgohary, Bo-Jhang Ho, Mani Srivastava, and Kai-Wei Chang. 2018. Generating natural language adversarial examples. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2890–2896, Brussels, Belgium. Association for Computational Linguistics.

Anish Athalye, Nicholas Carlini, and David Wagner. 2018. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 274–283. PMLR.

Steven Bird et al. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.".

Nicholas Carlini and David Wagner. 2017a. Adversarial examples are not easily detected: Bypassing ten

detection methods. In *Proc. 10th AISec workshop*, pages 3–14.

Nicholas Carlini and David Wagner. 2017b. Towards evaluating the robustness of neural networks. In *Proc. IEEE S&P*, pages 39–57.

Ciprian Chelba et al. 2013. One billion word benchmark for measuring progress in statistical language modeling. *arXiv preprint arXiv:1312.3005*.

Pin-Yu Chen et al. 2018. Ead: elastic-net attacks to deep neural networks via adversarial examples. In *Proc. AAAI*.

Gilad Cohen et al. 2020. Detecting adversarial samples using influence functions and nearest neighbors. In *Proc. IEEE/CVF CVPR*, pages 14453–14462.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc Le, and Ruslan Salakhutdinov. 2019. Transformer-XL: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2978–2988, Florence, Italy. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing Dou. 2018. HotFlip: White-box adversarial examples for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 31–36, Melbourne, Australia. Association for Computational Linguistics.

Reuben Feinman et al. 2017. Detecting adversarial samples from artifacts. *arXiv preprint arXiv:1703.00410*.

Christiane Fellbaum. 2005. Wordnet and wordnets. In Alex Barber, editor, *ELL*, pages 2–665. Elsevier.

Ian J Goodfellow et al. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*.

M. E. Houle et al. 2012. Generalized expansion dimension. In *2012 IEEE 12th ICDM Workshops*, pages 587–594.

Mohit Iyyer, John Wieting, Kevin Gimpel, and Luke Zettlemoyer. 2018. Adversarial example generation with syntactically controlled paraphrase networks. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1875–1885, New Orleans, Louisiana. Association for Computational Linguistics.

Robin Jia and Percy Liang. 2017. Adversarial examples for evaluating reading comprehension systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2021–2031, Copenhagen, Denmark. Association for Computational Linguistics.

Robin Jia, Aditi Raghunathan, Kerem Göksel, and Percy Liang. 2019. Certified robustness to adversarial word substitutions. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4129–4142, Hong Kong, China. Association for Computational Linguistics.

Di Jin et al. 2020. Is bert really robust? a strong baseline for natural language attack on text classification and entailment.

Erik Jones, Robin Jia, Aditi Raghunathan, and Percy Liang. 2020. Robust encodings: A framework for combating adversarial typos. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2752–2765, Online. Association for Computational Linguistics.

Kimin Lee et al. 2018. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *arXiv preprint arXiv:1807.03888*.

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

Dianqi Li, Yizhe Zhang, Hao Peng, Liqun Chen, Chris Brockett, Ming-Ting Sun, and Bill Dolan. 2021. Contextualized perturbation for textual adversarial attack. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5053–5069, Online. Association for Computational Linguistics.

Linyang Li, Ruotian Ma, Qipeng Guo, Xiangyang Xue, and Xipeng Qiu. 2020. BERT-ATTACK: Adversarial attack against BERT using BERT. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6193–6202, Online. Association for Computational Linguistics.

Yitong Li, Trevor Cohn, and Timothy Baldwin. 2016. Learning robust representations of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1979–1985, Austin, Texas. Association for Computational Linguistics.

Yitong Li, Trevor Cohn, and Timothy Baldwin. 2017. Robust training under linguistic adversity. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 21–27, Valencia, Spain. Association for Computational Linguistics.

Yinhan Liu et al. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Zhiyuan Liu et al. 2020. *Representation learning for natural language processing*. Springer Nature.

Xingjun Ma et al. 2018. Characterizing adversarial subspaces using local intrinsic dimensionality. *arXiv preprint arXiv:1801.02613*.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

Stephen Merity et al. 2016. Pointer sentinel mixture models.

Tomas Mikolov, Edouard Grave, Piotr Bojanowski, Christian Puhrsch, and Armand Joulin. 2018. Advances in pre-training distributed word representations. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Takeru Miyato et al. 2016. Adversarial training methods for semi-supervised text classification. *arXiv preprint arXiv:1605.07725*.

Seyed-Mohsen Moosavi-Dezfooli et al. 2016. Deepfool: a simple and accurate method to fool deep neural networks. In *Proc. CVPR*, pages 2574–2582.

Maximilian Mozes, Pontus Stenetorp, Bennett Kleinberg, and Lewis Griffin. 2021. Frequency-guided word substitutions for detecting textual adversarial examples. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 171–186, Online. Association for Computational Linguistics.

Nikola Mrkšić, Diarmuid Ó Séaghdha, Blaise Thomson, Milica Gašić, Lina M. Rojas-Barahona, Pei-Hao Su, David Vandyke, Tsung-Hsien Wen, and Steve Young. 2016. Counter-fitting word vectors to linguistic constraints. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 142–148, San Diego, California. Association for Computational Linguistics.

Nicolas Papernot and Patrick McDaniel. 2018. Deep k-nearest neighbors: Towards confident, interpretable and robust deep learning. *arXiv preprint arXiv:1803.04765*.

Nicolas Papernot et al. 2016a. Distillation as a defense to adversarial perturbations against deep neural networks. In *Proc. IEEE S&P*, pages 582–597. IEEE.

Nicolas Papernot et al. 2016b. The limitations of deep learning in adversarial settings. In *2016 IEEE EuroS&P*, pages 372–387.

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

Danish Pruthi, Bhuwan Dhingra, and Zachary C. Lipton. 2019. Combating adversarial misspellings with robust word recognition. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5582–5591, Florence, Italy. Association for Computational Linguistics.

Shuhuai Ren, Yihe Deng, Kun He, and Wanxiang Che. 2019. Generating natural language adversarial examples through probability weighted word saliency. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1085–1097, Florence, Italy. Association for Computational Linguistics.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2018. Semantically equivalent adversarial rules for debugging NLP models. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 856–865, Melbourne, Australia. Association for Computational Linguistics.

Christian Szegedy et al. 2013. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Zhilin Yang et al. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *arXiv preprint arXiv:1906.08237*.

Jin Yong Yoo and Yanjun Qi. 2021. Towards improving adversarial training of NLP models. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 945–956, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Yichao Zhou, Jyun-Yu Jiang, Kai-Wei Chang, and Wei Wang. 2019. Learning to discriminate perturbations for blocking adversarial attacks in text classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4904–4913, Hong Kong, China. Association for Computational Linguistics.

A Experimental Results Samples

Samples of outputs produced by the word-level attack and four detection classifiers on the MultiNLI dataset are shown in Table 8, to illustrate where some detection methods work while others do not. The DISP and FGWS baselines both also produce 'corrected' text; their outputs are included here.

In our experiments, the best accuracy of FGWS is when the frequency threshold is 92 and the threshold for the difference in prediction confidence is about 0.1916, therefore, if a word appears in the MultiNLI dataset training set and its occurrence frequency in the training corpus is lower than 92, it will be replaced by another word that is semantically similar and has higher occurrence frequency in the training set. If after transformations, the difference in prediction confidence before and after exceeds 0.1916, this example is considered as an adversarial example.

In example (a), MDRE, adapted LID, and DISP are successful, but FGWS does not detect this word-level adversarial example, because the occurrence frequency of the substituted word *shopping* for *store* is 1153 which is higher than the threshold 92, but original words *mentioning* and *buffer* are replaced by *name* and *pilot* respectively, since their occurrence frequencies are 67 and 30 in the MultiNLI training set which are lower than the threshold 92. From the DISP output, we can see that it detects *shopping* as a problem word and it is substituted by *do*.

In example (b), only adapted LID is successful. This is an odd (but not atypical) example in that the premise is not grammatical in written English, which might cause its representation differ from normal examples and lead MDRE to predict wrong. However, the prediction confidence about the premise and the hypothesis are unrelated from $\text{BERT}_{\text{BASE}}$ model is 90.49%, therefore, the word-level adversarial method have to make many changes to both premise and hypothesis to fool the target classifier. All words occurrence frequencies are above the threshold 92. FGWS and DISP fail in detecting most substitution words in this adversarial example.

In example (c), only MDRE and FGWS are successful. As with example (a), there is only a single word change. Even though *dipped* is not an infrequent word, there are only 45 occurrences in the MultiNLI training corpus, which is lower than the threshold 92, so FGWS detects it. The language model detector doesn't detect these three adversarial examples, since it fails to learn a threshold on the language model scores to separate normal and adversarial examples, and predict nearly all examples as normal examples.

Original example prediction: Entailment
Premise: Finally, it might be worth mentioning that the program has the capacity to store in a temporary memory buffer about 100 words (proper names, for instance) that it has identified as not stored in its dictionary. **Hypothesis**: It's possible to **store** words in a temporary dictionary, if they don't appear in a regular dictionary.
Word-level adversarial example prediction: Neutral
Premise: Finally, it might be worth **mentioning** that the program has the capacity to store in a temporary memory **buffer** about 100 words (proper names, for instance) that it has identified as not stored in its dictionary. **Hypothesis**: It's possible to **shopping** words in a temporary dictionary, if they don't appear in a regular dictionary.
DISP output of this word-level adversarial example
Premise: Finally, it might be worth **that** that the program has the capacity to store in a temporary memory buffer about 100 words (proper names, for instance) that it has identified as not stored in its dictionary. **Hypothesis**: It's possible to **do** words in a temporary dictionary, if they don't appear in a regular dictionary.
FGWS output of this word-level adversarial example
Premise: Finally, it might be worth **name** that the program has the capacity to store in a temporary memory **pilot** about 100 words (proper names, for instance) that it has identified as not stored in its dictionary. **Hypothesis**: It's possible to shopping words in a temporary dictionary, if they don't appear in a regular dictionary.

(a) An example with MDRE, adapted LID, and DISP correct predictions;
FGWS and the language model incorrect predictions on the adversarial example

Original example prediction: Neutral
Premise: I've been going up as a progress in **school**, so I, it will be a good **change** for me. **Hypothesis**: I **think** further **change** can **help** me **improve** even more.
Word-level Adversarial Example prediction: Entailment
Premise: I've been going up as a progress in **teaching**, so I, it will be a good **amendment** for me . **Hypothesis**: I **thought** further **alter** can **support me improvement** even more.
DISP output of this word-level adversarial example
Premise: I've been going up as a progress in teaching, so I **think** it will be a good amendment for me. **Hypothesis**: I thought further **that** can support **and** improvement even more.
FGWS output of this word-level adversarial example
Premise: I've been going up as a progress in teaching, so I, it will be a good amendment for me. **Hypothesis**: I thought further alter can support me improvement even more.

(b) An example with adapted LID correct prediction;
MDRE, DISP, FGWS, and the language model incorrect predictions on the adversarial example

Original example prediction: Contradiction
Premise: Increased profit came from missing fewer sales by being in stock a higher percentage of the time. **Hypothesis**: Profits **declined** because less sales were missed.
Word-level adversarial example prediction: Entailment
Premise: Increased profit came from missing fewer sales by being in stock a higher percentage of the time . **Hypothesis**: Profits **dipped** because less sales were missed .
DISP output of this word-level adversarial example
Premise: Increased profit came from missing fewer sales by being in stock a higher percentage of the time . **Hypothesis**: Profits dipped because less sales were missed .
FGWS output of this word-level adversarial example
Premise: Increased profit came from missing fewer sales by being in stock a higher percentage of the time . **Hypothesis**: Profits **duck** because less sales were missed .

(c) An example with MDRE, FGWS correct predictions;
adapted LID, DISP, and the language model incorrect predictions on the adversarial example

Table 8: Examples of detection results on the MultiNLI dataset