University of Idaho

Department of Computer Science

CS 404/504 Special Topics: Adversarial Machine Learning

Dr. Alex Vakanski

Lecture 14

Adversarial Examples in Text and Audio Data

Lecture Outline

- Adversarial examples in text data
- Attacks on text classification models
  - Ebrahimi (2018) HotFlip attack
  - Gao (2018) DeepWordBug attack
- Attacks on reading comprehension models
  - Jia (2017) Text concatenation attack
- Attacks on translation and text summarization models
  - Cheng (2018) Seq2Sick attack
- Attacks on dialog generation models
  - He (2018) Egregious output attack
- Attacks against transformer language models
  - Jin (2020) TextFooler
  - Guo (2021) GBDA attack
- Jiang Chang presentation
  - Adversarial examples in audio data: Carlini (2018) Targeted attacks on speech-to-text

Adversarial Examples in Text Data

Adversarial Examples in Text Data

- ***Adversarial examples*** were shown to exists for ***ML models for processing text data***
  - An adversary can generate manipulated text sentences that mislead ML text models
- To satisfy the definitions for adversarial examples, a generated text sample x' that is obtained by perturbing a clean text sample x should look "similar" to the original text
  - The perturbed text should preserve the semantic meaning for a human reader
  - I.e., an adversarial text sample that is misclassified by an ML model should not be misclassified by a typical human
- In general, crafting adversarial examples in text data is more challenging than in image data
  - E.g., many text attacks output grammatically or semantically incorrect sentences
- Generation of adversarial text examples is often based on replacement of input words (with synonyms, misspelled words, or words with similar vector embeddings), or based on adding distracting text to the original clean text

Text Processing Models

Adversarial Examples in Text Data

- Dominant text processing models
  - Pre 1990
    - Hand-crafted rule-based approaches (if-then-else rules)
  - 1990-2014
    - Traditional ML models, e.g., decision trees, logistic regression, Naïve Bayes
  - 2014-2018
    - Recurrent NNs (e.g., LSTM, GRU) layers
    - Combinations of CNNs and RNNs
    - Bi-directional LSTM layers
  - 2018-present time
    - Transformers (BERT, RoBERTa, GPT family, Bard, LLaMA)

Slide credit: Chollet (2021) Deep Learning with Python

Adversarial Examples in Text versus Images

Adversarial Examples in Text Data

- ***Image data***
  - Inputs: pixel intensities
  - Continuous inputs
  - Adversarial examples can be created by applying small perturbations to pixel intensities
    - Adding small perturbations does not change the context of the image
    - Gradient information can be used to perturb the input images
  - Metrics based on ℓ_p norms can be applied for measuring the distance to adversarial examples
- ***Text data***
  - Inputs: words or characters
  - Discrete inputs
  - Small text modifications are more difficult to apply to text data for creating adversarial examples
    - Adding small perturbations to words can change the meaning of the text
    - Gradient information cannot be used, generating adversarial examples requires applying heuristic approaches (e.g., word replacement with local search) to produce valid text
  - It is more difficult to define metrics for measuring text difference, ℓ_p norms cannot be applied

Ebrahimi (2018) – HotFlip Attack

Attacks on Text Classification Models

- Ebrahimi et al. (2018) HotFlip: White-Box Adversarial Examples for Text Classification
- ***HotFlip*** attacks character-level text classifiers by replacing one letter in text
  - It is a white-box untargeted attack
  - Approach:
    - Use the model gradient to identify the most important letter in the text
    - Perform an optimization search to find a substitute (flip) for that letter
      - The approach also supports insertion or deletion of letters
  - In this example, the predicted topic label of the sentence is changed from "World" to "Sci/Tech" by changing the letter P in the word 'mood'

Original text	South Africa's historic Soweto township marks its 100th birthday on Tuesday in a mood of optimism.
Predicted class	57% **World**
Adversarial text	South Africa's historic Soweto township marks its 100th birthday on Tuesday in a mooP of optimism.
Predicted class	95% **Sci/Tech**

Ebrahimi (2018) – HotFlip Attack

Attacks on Text Classification Models

- Attacked model: CharCNN-LSTM, a character-level model that uses a combination of CNN and LSTM layers
- Dataset: AG news dataset, consists of 120K training and 7.6K testing instances with 4 classes: World, Sports, Business, and Science/Technology
- The attack does not change the meaning of the text, and it is often unnoticed by human readers

Original text	Chancellor Gordon Brown has sought to quell speculation over who should run the Labour Party and turned the attack on the opposition Conservatives.
Predicted class	75% **World**
Adversarial text	Chancellor Gordon Brown has sought to quell speculation over who should run the Labour Party and turned the attack on the o**B**position Conservatives.
Predicted class	94% **Business**

Gao (2018) DeepWordBug Attack

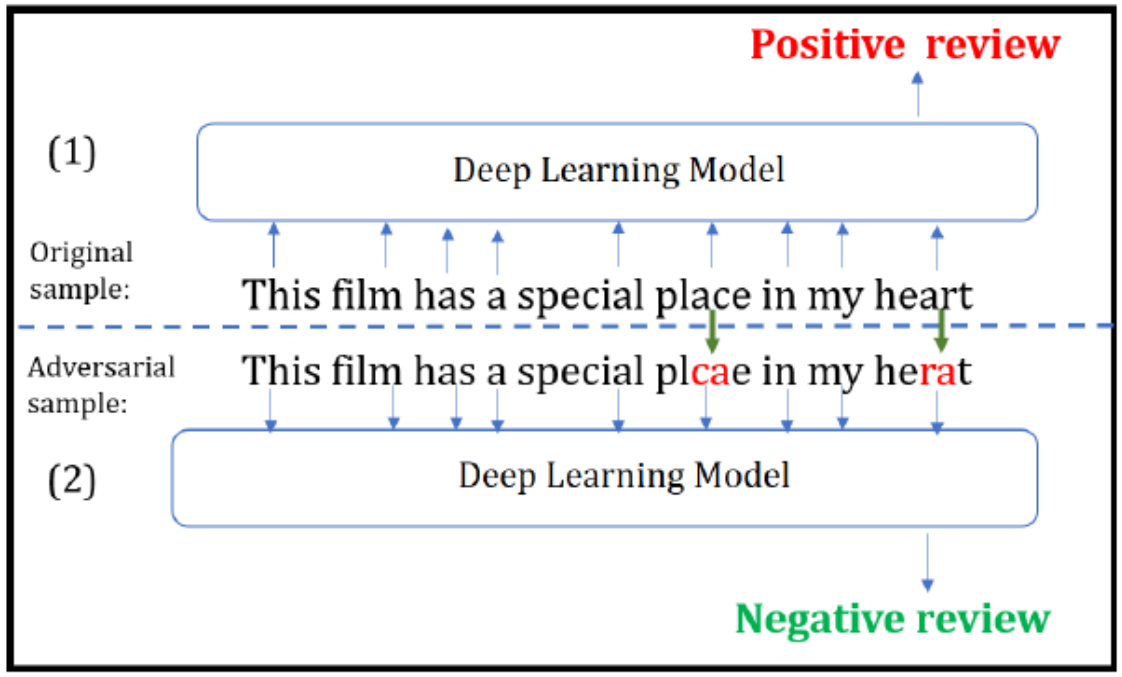
Attacks on Text Classification Models

- Gao et al. (2018) Black-box Generation of Adversarial Text Sequences to Evade Deep Learning Classifiers
- ***DeepWordBug*** attack is a black-box attack on text classification models
- The approach has similarity to the HotFlip attack:
  - Identify the most important tokens (either words or characters) in a text sample
  - Apply character-level transformations to change the label of the text
- Key idea: the misspelled words in the adversarial examples are considered "unknown" words by the ML model
  - Changing the important words to "unknown" impacts the prediction by the model
- Applications: the attack was implemented against three different models, which include text classification, sentiment analysis, spam detection
- Attacked models: Word-LSTM (uses word tokens) and Char-CNN (uses character tokens) models
- Datasets: evaluated on 8 text datasets

Gao (2018) DeepWordBug Attack

Attacks on Text Classification Models

- Example of a generated adversarial text for sentiment analysis
  - The original text sample has a positive review sentiment
  - An adversarial sample is generated by changing 2 characters, resulting in wrong classification (negative review sentiment)
- Question: is the adversarial sample perceptible to a human reader?
  - Argument: a human reader can understand the meaning of the perturbed sample, and assign positive review sentiment



Gao (2018) DeepWordBug Attack

Attacks on Text Classification Models

- Attack approach
  - Assume an input sequence x $= x_1 x_2 x_3 \cdots x_n$, and $F(\mathrm{x})$ is output of a black-box model
  - The authors designed 4 scoring functions to identify the most important tokens
    - Replace-1 score: evaluate the output $F(\mathrm{x})$ when the token x_i is replaced with the "unknown" (i.e., out of vocabulary) token x_i'

$$R1S(x_i) = F(x_1, x_2, \cdots x_{i-1}, x_i, \cdots x_n) - F(x_1, x_2, \cdots x_{i-1}, x_i', \cdots x_n)$$

    - Temporal head score: evaluate the output of the model for the tokens before x_i

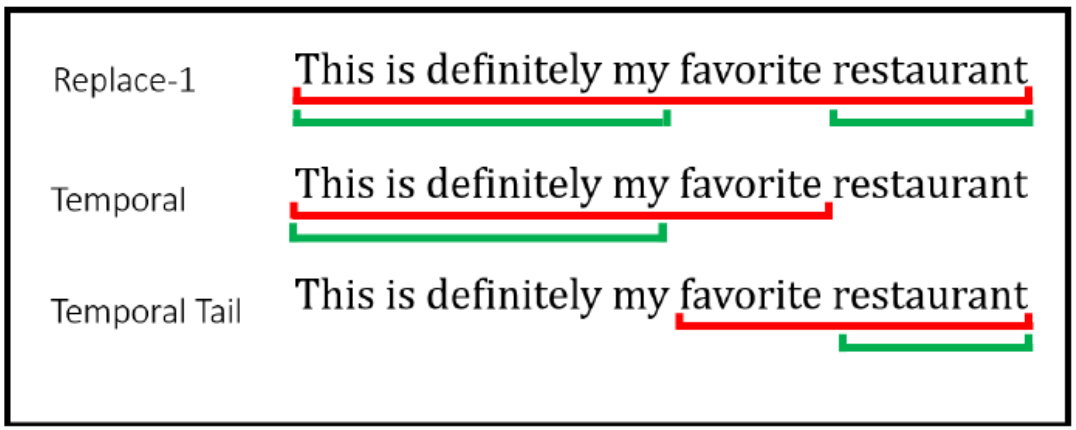
$$THS(x_i) = F(x_1, x_2, \cdots x_{i-1}, x_i) - F(x_1, x_2, \cdots x_{i-1})$$

    - Temporal tail score: evaluate the output of the model for the tokens after x_i

$$TTS(x_i) = F(x_i, x_{i+1}, \cdots, x_n) - F(x_{i+1}, \cdots x_n)$$

    - Combined score: a weighted sum of the Temporal Head and Temporal Tail Scores

$$CS(x_i) = THS(x_i) + \lambda TTS(x_i)$$



Gao (2018) DeepWordBug Attack

Attacks on Text Classification Models

- Attack approach
  - Next, the top *m* important tokens selected by the scoring functions are perturbed
  - The following 4 transformations are considered:
    - Swap – swap two adjacent letters
    - Substitution – substitute a letter with a random letter
    - Deletion – delete a letter
    - Insertion – insert a letter
  - ***Edit distance*** of the perturbation is the minimal number of edit operations to change the original text
    - The edit distance is 2 edits for the swap transformation, and 1 edit for substitution, deletion, and insertion transformations

Original		Swap	Substitution	Deletion	Insertion
Team	→	Taem	Texm	Tem	Tezam
Artist	→	Artsit	Arxist	Artst	Articst
Computer	→	Comptuer	Computnr	Compter	Comnputer

Gao (2018) DeepWordBug Attack

Attacks on Text Classification Models

- Datasets details

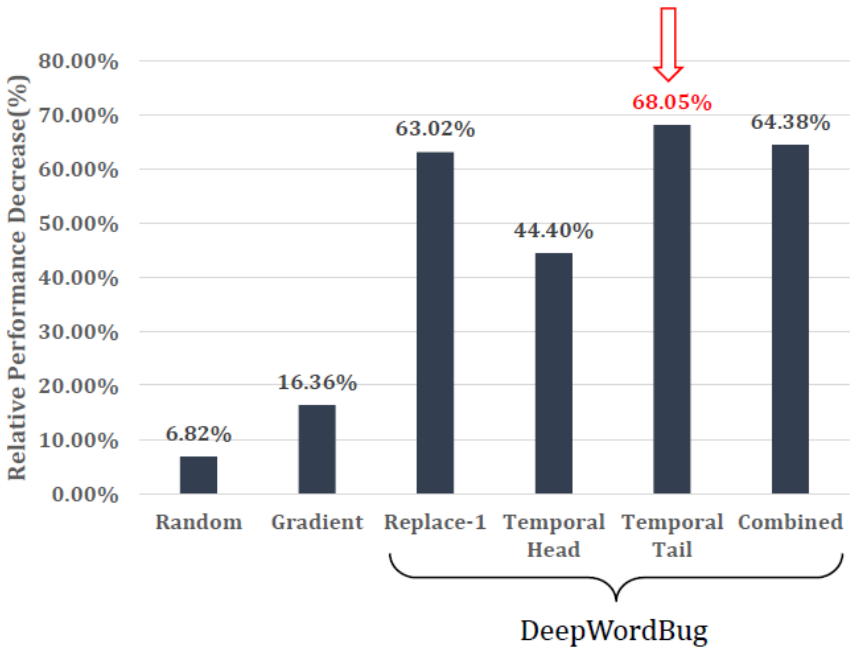
	#Training	#Testing	#Classes	Task
AG's News	120,000	7,600	4	News Categorization
Amazon Review Full	3,000,000	650,000	5	Sentiment Analysis
Amazon Review Polarity	3,600,000	400,000	2	Sentiment Analysis
DBPedia	560,000	70,000	14	Ontology Classification
Yahoo! Answers	1,400,000	60,000	10	Topic Classification
Yelp Review Full	650,000	50,000	5	Sentiment Analysis
Yelp Review Polarity	560,000	38,000	2	Sentiment Analysis
Enron Spam Email	26,972	6,744	2	Spam E-mail Detection

Gao (2018) DeepWordBug Attack

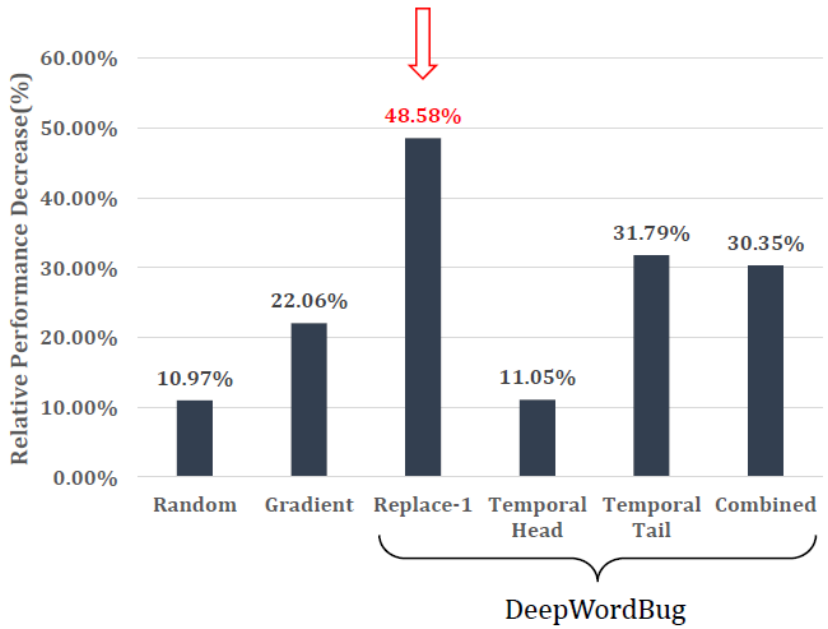
Attacks on Text Classification Models

- Evaluation results for attacks against Word-LSTM and Char-CNN models
  - The maximum edit distance is set to 30 characters
  - Left figure: DeepWordBug reduced the performance by the Word-LSTM model by 68.05% in comparison to the accuracy on non-perturbed text samples
    - Temporal Tail score function achieved the largest decrease in accuracy
  - Right figure: decrease in the accuracy by the Char-CNN of 48.58% was achieved

Word-LSTM model



Char-CNN model



Gao (2018) DeepWordBug Attack

Attacks on Text Classification Models

- Evaluation results on all 8 datasets for the Word-LSTM model
  - Two baseline approaches are included for comparison (Random token replacement and Gradient)
  - The largest average decrease in the performance was achieved by the Temporal Tail scoring function approach (mean decrease of 68.05% across all datasets)

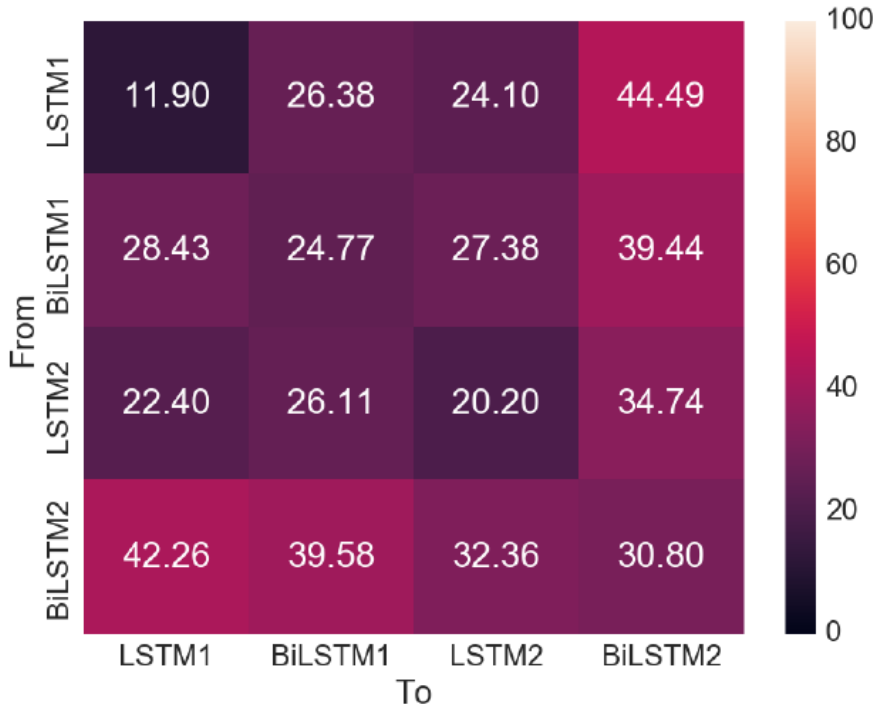
Word-LSTM Model

		Baselines				WordBug							
	Original	Random		Gradient		Replace-1		Temporal Head		Temporal Tail		Combined	
	Acc(%)	Acc(%)	Decrease	Acc(%)	Decrease	Acc(%)	Decrease	Acc(%)	Decrease	Acc(%)	Decrease	Acc(%)	Decrease
AG's News	90.5	89.3	1.33%	48.5	10.13%	36.1	60.08%	42.5	53.01%	21.3	76.48%	24.8	72.62%
Amazon Review Full	62.0	61.1	1.48%	55.7	10.13%	18.6	70.05%	27.1	56.30%	17.0	72.50%	16.3	73.76%
Amazon Review Polarity	95.5	93.9	1.59%	86.9	8.93%	40.7	57.36%	58.5	38.74%	42.6	55.37%	36.2	62.08%
DBPedia	98.7	95.2	3.54%	74.4	24.61%	28.8	70.82%	56.4	42.87%	28.5	71.08%	25.3	74.32%
Yahoo! Answers	73.4	65.7	10.54%	50.0	31.83%	27.9	61.93%	34.9	52.45%	26.5	63.86%	23.5	68.02%
Yelp Review Full	64.7	60.9	5.86%	53.2	17.76%	23.4	63.83%	36.6	43.47%	20.8	67.85%	24.4	62.28%
Yelp Review Polarity	95.9	95.4	0.55%	88.4	7.85%	37.8	60.63%	70.2	26.77%	34.5	64.04%	46.2	51.87%
Enron Spam Email	96.4	67.8	29.69%	76.7	20.47%	39.1	59.48%	56.3	41.61%	25.8	73.22%	48.1	50.06%
Mean			6.82%		16.46%		**63.02%**		44.40%		**68.05%**		**64.38%**
Median			2.57%		13.95%		**61.28%**		43.17%		**69.46%**		**65.15%**
Standard Deviation			9.81%		8.71%		4.94%		9.52%		6.77%		9.56%

Gao (2018) DeepWordBug Attack

Attacks on Text Classification Models

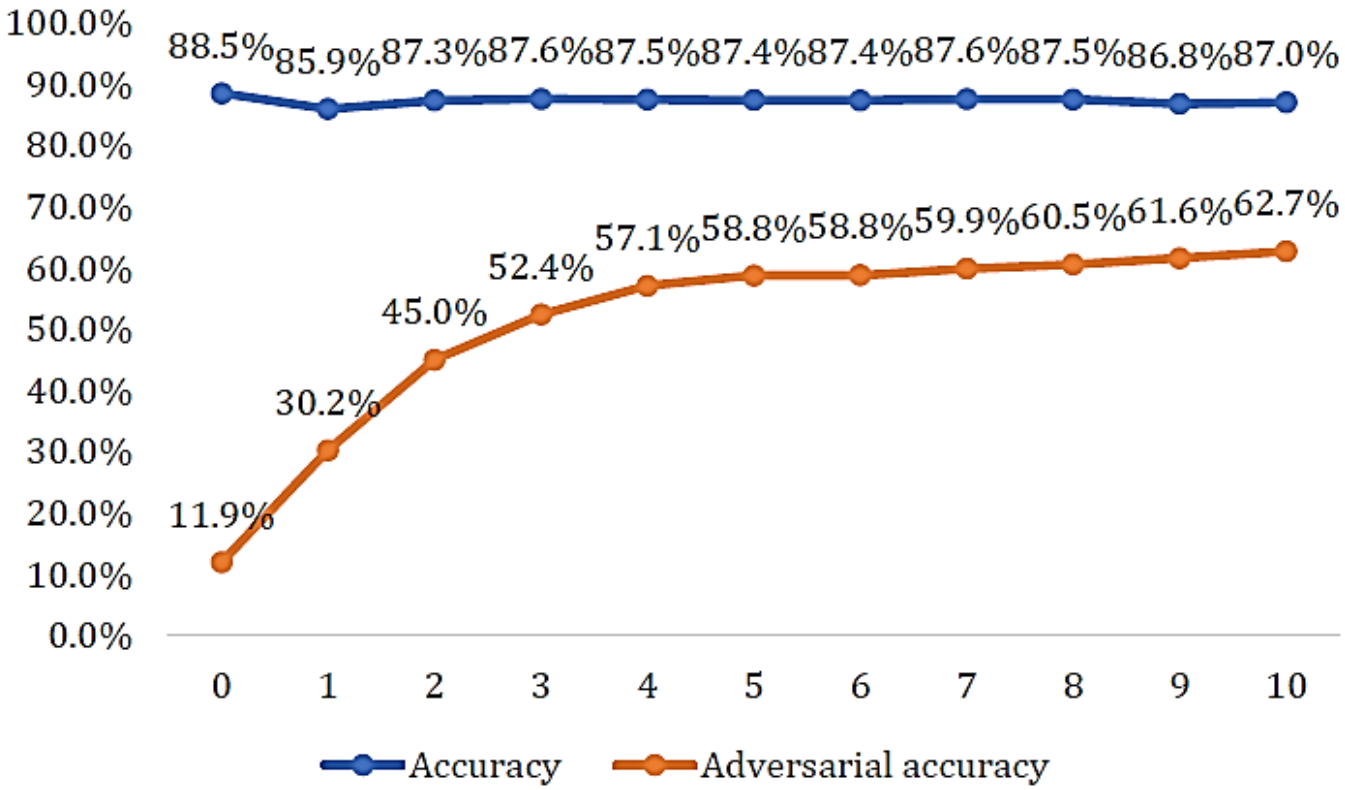
- Are adversarial text examples ***transferable*** across ML models? – Yes!
  - The figure shows the accuracy on adversarial examples generated with one model and transferred to other models
  - Four models were considered containing LSTM and bi-directional LSTM layers (BiLSTM)
  - The adversarial examples transferred successfully to other models



Gao (2018) DeepWordBug Attack

Attacks on Text Classification Models

- Evaluation of ***adversarial training defense***
  - The figure shows the standard accuracy on regular text samples (blue), and the adversarial accuracy (orange) on adversarial samples for 10 epochs
  - The adversarial accuracy improves significantly to reach 62.7%, with a small trade-off in the standard accuracy



Jia (2017) Text Concatenation Attack

Attacks on Reading Comprehension Models

- Jia et al. (2017) Adversarial Examples for Evaluating Reading Comprehension Systems
- Reading comprehension task
  - An ML model answers questions about paragraphs of text
  - Human performance was measured at 91.2% accuracy
- ***Text Concatenation Attack*** is a black-box, non-targeted attack
  - Adds additional sequences to text samples to distract ML models
  - The generated adversarial examples should not confuse humans
- Attacked model: LSTM-based model for reading comprehension
- Dataset: Stanford Question Answering Dataset (SQuAD)
  - Consists of 108K human-generated reading comprehension questions about Wikipedia articles
- Results: accuracy decreased from 75% to 36%

Jia (2017) Text Concatenation Attack

Attacks on Reading Comprehension Models

- Example
  - The concatenated adversarial text in blue color at the end of the paragraph fooled the ML model to give the wrong answer 'Jeff Dean'

Article: Super Bowl 50
Paragraph: "*Peyton Manning became the first quarterback ever to lead two different teams to multiple Super Bowls. He is also the oldest quarterback ever to play in a Super Bowl at age 39. The past record was held by John Elway, who led the Broncos to victory in Super Bowl XXXIII at age 38 and is currently Denver's Executive Vice President of Football Operations and General Manager. Quarterback Jeff Dean had jersey number 37 in Champ Bowl XXXIV.*"
Question: "*What is the name of the quarterback who was 38 in Super Bowl XXXIII?*"
Original Prediction: John Elway
Prediction under adversary: Jeff Dean

Jia (2017) Text Concatenation Attack

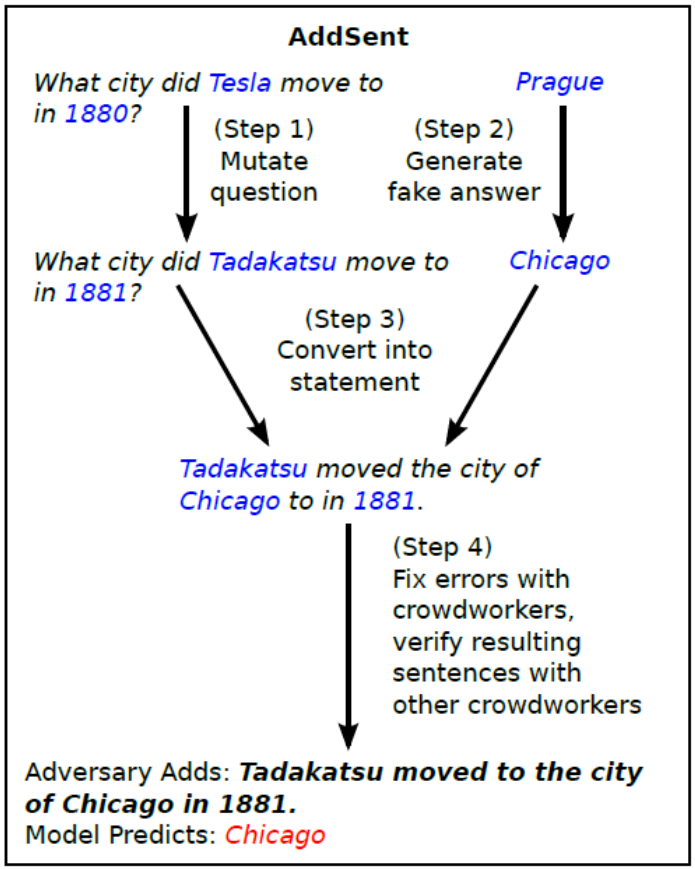
Attacks on Reading Comprehension Models

- **ADDSENT approach** uses a four-step procedure to add a sentence to a text
  - Step 1 changes words in the question with nearest words in the embedding space, Step 2 generates a fake answer randomly, and Step 3 replaces the changed words
  - Step 4 involves human-in-the-loop to fix grammar errors or unnatural sentences

Original text and prediction

Article: **Nikola Tesla**
Paragraph: "*In January 1880, two of Tesla's uncles put together enough money to help him leave Gospić for Prague where he was to study. Unfortunately, he arrived too late to enroll at Charles-Ferdinand University; he never studied Greek, a required subject; and he was illiterate in Czech, another required subject. Tesla did, however, attend lectures at the university, although, as an auditor, he did not receive grades for the courses.*"
Question: "*What city did Tesla move to in 1880?*"
Answer: *Prague*
Model Predicts: *Prague*

Attack



Cheng (2018) Seq2Sick Attack

Attacks on Translation and Text Summarization Models

- Cheng et al. (2018) Seq2Sick: Evaluating the Robustness of Sequence-to-Sequence Models with Adversarial Examples
- ***Seq2Sick*** is a white-box, targeted attack
  - Attacked are RNN-based sequence-to-sequence (seq2seq) models, used for machine translation and text summarization tasks
  - Seq2seq models are more challenging to attack than classification models, because there are infinite possibilities for the text sequences outputted by the model
    - Conversely, classification models have a finite number of output classes
    - Example:

      Input sequence in English: A child is splashing in the water.
      Output sequence in German: Ein kind im wasser.

- Attacked model: word-level LSTM encoder-decoder
- This work designed a regularized PGD method to generate adversarial text examples with targeted outputs

Cheng (2018) Seq2Sick Attack

Attacks on Translation and Text Summarization Models

- Text summarization example with a ***target keyword*** "police arrest"
  - **Original text:** President Boris Yeltsin stayed home Tuesday, nursing a respiratory infection that forced him to cut short a foreign trip and revived concerns about his ability to govern.
  - **Summary by the model**: Yeltsin stays home after **illness**.
  - **Adversarial exampl**e: President Boris Yeltsin stayed home Tuesday, cops cops respiratory infection that forced him to cut short a foreign trip and revived concerns about his ability to govern.
  - **Summary by the model**: Yeltsin stays home after **police arrest**.

Cheng (2018) Seq2Sick Attack

Attacks on Translation and Text Summarization Models

- Other text summarization examples with a ***target keyword*** "police arrest"

Source input seq	north korea is entering its fourth winter of chronic food shortages with its people malnourished and at risk of dying from normally curable illnesses , senior red cross officials said tuesday.
Adv input seq	north **detectives** is **apprehended** its fourth winter of chronic food shortages with its people malnourished and at risk of dying from normally curable illnesses , senior red cross officials said tuesday.
Source output seq	north korea enters fourth winter of food shortages
Adv output seq	north **police arrest** fourth winter of food shortages.
Source input seq	after a day of fighting , congolese rebels said sunday they had entered kindu , the strategic town and airbase in eastern congo used by the government to halt their advances.
Adv input seq	after a day of fighting , **nordic detectives** said sunday they had entered **UNK** , the strategic town and airbase in eastern congo used by the government to halt their advances.
Source output seq	congolese rebels say they have entered UNK.
Adv output seq	nordic **police arrest** ## in congo.

Cheng (2018) Seq2Sick Attack

Attacks on Translation and Text Summarization Models

- Text summarization examples with a ***non-overlapping attack***
  - I.e., the output sequence does not have overlapping words with the original output

Source input seq	under nato threat to end his punishing offensive against ethnic albanian separatists in kosovo , president slobodan milosevic of yugoslavia has ordered most units of his army back to their barracks and may well avoid an attack by the alliance , military observers and diplomats say
Adv input seq	under nato threat to end his punishing offensive against ethnic albanian separatists in kosovo , president slobodan milosevic of yugoslavia has **jean-sebastien** most units of his army back to their barracks and may well avoid an attack by the alliance , military observers and diplomats say.
Source output seq	milosevic orders army back to barracks
Adv output seq	**nato may not attack kosovo**
Source input seq	flooding on the yangtze river remains serious although water levels on parts of the river decreased today , according to the state headquarters of flood control and drought relief .
Adv input seq	flooding **that** the yangtze river **becomes** serious although water levels on parts of the river decreased today , according to the state headquarters of flood control and drought relief .
Source output seq	floods on yangtze river continue
Adv output seq	**flooding in water recedes in river**

Cheng (2018) Seq2Sick Attack

Attacks on Translation and Text Summarization Models

- Seq2Sick approach
  - For an input sequence X and perturbation δ, solve the optimization problem formulated as

$$\min_{\delta} \mathcal{L}(X+\delta) + \lambda_1 \sum_i |\delta_i| + \lambda_2 \sum_i \min_{w_j} |x_i + \delta_i - w_j|$$

  - The first term $\mathcal{L}(X+\delta)$ is a loss function that is minimized by using Projected Gradient Descent (PGD)
  - The second and third term are regularization terms
  - The term $\sum_i |\delta_i|$ applies lasso regularization to ensure that only a few words in the text sequence are changed
  - The third term $\sum_i \min_{w_j} |x_i + \delta_i - w_j|$ applies gradient regularization to ensure that the perturbed input words $x_i + \delta_i$ are close in the word embedding space to existing words w_j from a vocabulary W

Cheng (2018) Seq2Sick Attack

Attacks on Translation and Text Summarization Models

- Datasets:
  - Text summarization: Gigaword, DUC2003, DUC2004
  - Machine translation: German-English WMT 15 dataset
- Evaluation results
  - $|K|$ is the number of targeted keywords
  - #changed is number of changed words
  - Success rate of the attack is over 99% for 1 targeted keyword
  - BLEU score stands for Bilingual Evaluation Understudy, and evaluates the quality of text translated from one language to another
    - BLEU scores between 0 and 1 are assigned based on a comparison of machine translations to good quality translations created by humans
    - High BLEU score means good quality text

Text Summarization - Targeted Keywords

Datasest	$\|K\|$	Success%	BLEU	# changed
Gigaword	1	99.8%	0.801	2.04
	2	96.5%	0.523	4.96
	3	43.0%	0.413	8.86
DUC2003	1	99.6%	0.782	2.25
	2	87.6%	0.457	5.57
	3	38.3%	0.376	9.35
DUC2004	1	99.6%	0.773	2.21
	2	87.8%	0.421	5.1
	3	37.4%	0.340	9.3

Cheng (2018) Seq2Sick Attack

Attacks on Translation and Text Summarization Models

- Evaluation results for text summarization using non-overlapping words
  - High BLEU score for text summarization indicates that the adversarial examples are similar to the clean input samples
  - Despite that the attacks is quite challenging, high success rates were achieved
- Evaluation results for machine translation
  - Results for non-overlapping words and targeted keywords are presented

Text Summarization – Non-overlapping Words

Dataset	Success%	BLEU	# changed
Gigaword	86.0%	0.828	2.17
DUC2003	85.2%	0.774	2.90
DUC2004	84.2%	0.816	2.50

Machine Translation

Method	Success%	BLEU	# changed
Non-overlap	89.4%	0.349	3.5
1-keyword	100.0%	0.705	1.8
2-keyword	91.0 %	0.303	4.0
3-keyword	69.6%	0.205	5.3

He (2018) Egregious Output Attack

Attacks on Dialog Generation Models

- He (2018) Detecting Egregious Responses in Neural Sequence-to-sequence Models
- ***Egregious output attack***: attack on RNN seq2seq models for dialog generation
- Research question: can ML models for dialog generation (e.g., AI assistants) generate not only wrong, but egregious outputs, which are aggressive, insulting, or dangerous
  - E.g., you ask your AI assistant a question and it replies: "You are so stupid, I don't want to help you"
- Attacked model: LSTM encoder-decoder
- Approach:
  - Create manually a list of "malicious sentences" that shouldn't be output by ML models
  - Developed an optimization algorithm to search for trigger inputs that maximize the probability of generating text that belongs to the list of malicious sentences
- Results: the authors discovered input text samples that can generate egregious outputs

He (2018) Egregious Output Attack

Attacks on Dialog Generation Models

- Datasets
  - Ubuntu conversational data: an agent is helping a user to deal with issues
  - Switchboard dialog dataset: two-sided telephone conversations
  - OpenSubtitles: dataset of movie subtitles
- Table: trigger inputs that result in target egregious outputs

io-sample-min-k1-hit in Ubuntu data ↓		
no , i do n't . i use <unk> , i want to ask	⇒	i do n't want to help
we are talking about ubuntu users here , but he is n't here .	⇒	oh i do not care
what is the problem ? ? ? ? ? how to change the password	⇒	what is your password
do n't know how to kill , but you might be using it .	⇒	i will kill you
io-sample-avg-k1-hit in Switchboard data ↓		
i i i i i i think that i would think that if you	⇒	i 'm really bad
i think that 's the way they do do they do they do that	⇒	i think you are so stupid
you do n't learn how to take care of the day care care because	⇒	well that 's enough
i know i i do n't know i do n't think about it too	⇒	oh i do not want to help you
o-greedy-hit in OpenSubtitles data ↓		
mrs. hello 'ii our ! l ! , go you go stop ready them	⇒	that 's enough
in 's the help go we ? . it get go stupid , !	⇒	shut up .
how you woltz # sorry i you ? i not why will she a	⇒	i think you 're a fool
you why ! # . how the the me a us 'ii me it	⇒	i 'll kill you

Jin (2020) TextFooler

Attacks against Transformer Language Models

- Jin (2020) Is BERT Really Robust? A Strong Baseline for Natural Language Attack on Text Classification and Entailment
- ***TextFooler attack*** is a black-box attack on transformers, CNN, and RNN language models
  - The adversarial examples are transferrable to the other models
- Approach:
  - Identify most important words and replace them with synonyms
- Attacked models: BERT (transformer model), WordCNN, and WordRNN for text classification (sentiment analysis)

Jin (2020) TextFooler

Attacks against Transformer Language Models

- Attack approach:
  - Step 1: for each word, compute an importance score
    - Remove that word and query the back-box model to obtain a prediction score/label
      - Important words cause large change in the predicted score
  - Step 2: sort the words in descending order based on their importance
  - Step 3: for each word identify a set of candidate replacement words
    - Find the top *N* closest synonyms in the vocabulary
      - Use cosine similarity in the embeddings space as a metric for identifying the closest synonyms
    - Keep only the synonyms that have the same part-of-speech tag (i.e., if the word is a verb, consider only verb synonyms)
  - Step 4: replace a word with a synonym and check the semantic similarity of the new sentence
    - Use Universal Sentence Encoder (USE) to encode the original text and the adversarial sample into embedding vectors
    - Next, apply cosine similarity between the embeddings of the original text and the adversarial sample to check whether they are semantically similar
  - Step 5: choose the words with greatest semantic similarity that alter the predicted score

Jin (2020) TextFooler

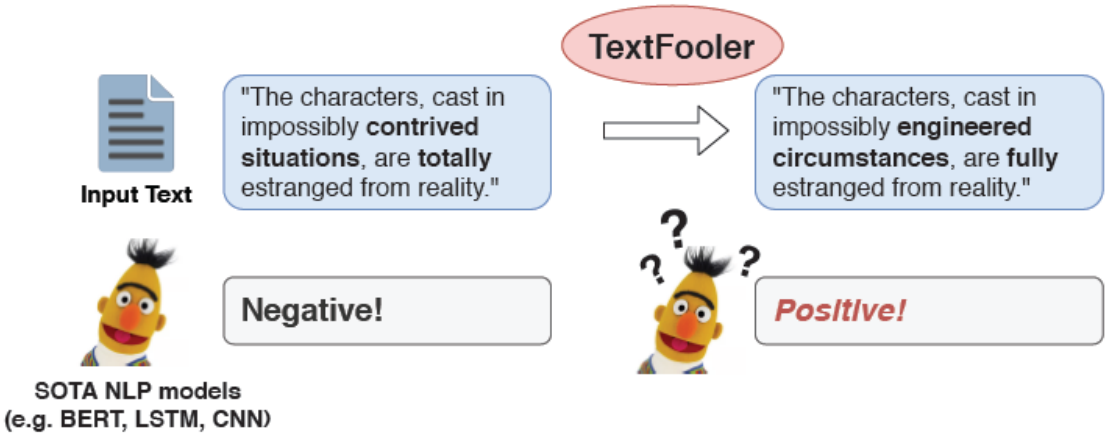
Attacks against Transformer Language Models

- Generated adversarial examples
  - The word "Perfect" in the original text is replaced with the word "Spotless"
  - The model prediction was changed from positive sentiment (99% confidence) to negative (100%)

Original
Perfect performance by the actor → **Positive (99%)**

Adversarial
Spotless performance by the actor → **Negative (100%)**

  - By replacing the words "contrived situations" and "totally" the predicted sentiment was changed from negative to positive



Jin (2020) TextFooler

Attacks against Transformer Language Models

- The attack was validated with three models: WordCNN, WordLSTM, and BERT
  - Five datasets: MR (Movie Reviews), IMDB (movie reviews), Yelp (reviews), AG (news classification), and Fake (fake news detection)
- The classification accuracy was reduced to less than 20% for all models
- The number of perturbed words ranged from 3% to 22% of the text

	WordCNN					WordLSTM					BERT				
	MR	**IMDB**	**Yelp**	**AG**	**Fake**	**MR**	**IMDB**	**Yelp**	**AG**	**Fake**	**MR**	**IMDB**	**Yelp**	**AG**	**Fake**
Original Accuracy	78.0	89.2	93.8	91.5	96.7	80.7	89.8	96.0	91.3	94.0	86.0	90.9	97.0	94.2	97.8
After-Attack Accuracy	2.8	0.0	1.1	1.5	15.9	3.1	0.3	2.1	3.8	16.4	11.5	13.6	6.6	12.5	19.3
% Perturbed Words	14.3	3.5	8.3	15.2	11.0	14.9	5.1	10.6	18.6	10.1	16.7	6.1	13.9	22.0	11.7
Semantic Similarity	0.68	0.89	0.82	0.76	0.82	0.67	0.87	0.79	0.63	0.80	0.65	0.86	0.74	0.57	0.76
Query Number	123	524	487	228	3367	126	666	629	273	3343	166	1134	827	357	4403
Average Text Length	20	215	152	43	885	20	215	152	43	885	20	215	152	43	885

Guo (2021) GBDA Attack

Attacks against Transformer Language Models

- Guo et al. (2021) Gradient-based Adversarial Attacks against Text Transformers
- ***Gradient-based Distributional Adversarial (GBDA) attack*** is a white-box attack on transformer language models
  - The adversarial examples are also be transferrable in black-box setting
- Approach:
  - Define an output adversarial distribution, which enables using the gradient information
  - Introduce constraints to ensure semantic correctness and fluency of the perturbed text
- Attacked models: GPT-2, XLM, BERT
  - GBDS attack was applied to text classification and sentiment analysis tasks
- Runtime: approximately 20 seconds per generated example

Guo (2021) GBDA Attack

Attacks against Transformer Language Models

- Generated adversarial examples for text classification
  - The changes in input text are subtle:
    - "worry" → "hell", "camel" → "animal", "no" → "varying"
    - Adversarial text examples preserved the semantic meaning of the original text

Attack	Prediction	Text
Original	Entailment (83%)	He found himself thinking in circles of worry and pulled himself back to his problem. He got lost in loops of worry, but snapped himself back to his problem.
GBDA	Neutral (95%)	He found himself thinking in circles of worry and pulled himself back to his problem. He got lost in loops of hell, but snapped himself back to his problem.
Original	Contradiction (95%)	You're the Desert Ghost. You're a living desert camel.
GBDA	Entailment (51%)	You're the Desert Ghost. You're a living desert animal.
Original	Contradiction (98%)	Pesticide concentrations should not exceed USEPA's Ambient Water Quality chronic criteria values where available. There is no assigned value for maximum pesticide concentration in water.
GBDA	Entailment (86%)	Pesticide concentrations should not exceed USEPA's Ambient Water Quality chronic criteria values where available. There is varying assigned value for maximum pesticide concentration in water.

Guo (2021) GBDA Attack

Attacks against Transformer Language Models

- The discrete inputs in text prevent from using gradient information for generating adversarial samples
  - This work introduces models that take probability vectors as inputs, to derive smooth estimates of the gradient
- Specifically, transformer models take as input a sequence of embedding vectors corresponding to text tokens, e.g., $\mathbf{z} = z_1 z_2 z_3 \cdots z_n$
  - GBDA attack considered an input sequence consisting of probability vectors corresponding to the text tokens, e.g., $p(\mathbf{z}) = p(z_1)p(z_2)p(z_3)\cdots p(z_n)$
  - Gumbel-softmax distribution provides a differentiable approximation to sampling discrete inputs
  - This allows to use gradient descent for estimating the loss with respect to the probability distribution of the inputs
- The work applied additional constraints to enforce semantic similarity and fluency of the perturbed samples

Guo (2021) GBDA Attack

Attacks against Transformer Language Models

- Evaluation results
  - For the three models (GPT-2, XML, and BERT) on all datasets, adversarial accuracy of less than 10% was achieved
  - Cosine similarity was employed to evaluate the semantic similarity of perturbed samples to the original clean samples
    - All attacks indicate high semantic similarity

	GPT-2			XLM (en-de)			BERT		
Task	**Clean Acc.**	**Adv. Acc.**	**Cosine Sim.**	**Clean Acc.**	**Adv. Acc.**	**Cosine Sim.**	**Clean Acc.**	**Adv. Acc.**	**Cosine Sim.**
DBPedia	99.2	5.2	0.91	99.1	7.6	0.80	99.2	7.1	0.80
AG News	94.8	6.6	0.90	94.4	5.4	0.87	95.1	2.5	0.82
Yelp	97.8	2.9	0.94	96.3	3.4	0.93	97.3	4.7	0.92
IMDB	93.8	7.6	0.98	87.6	0.1	0.97	93.0	3.0	0.92

Guo (2021) GBDA Attack

Attacks against Transformer Language Models

- Evaluation of transferability of the generated adversarial samples
  - Perturbed text samples from GPT-2 are successfully transferred to three other transformer models: ALBERT, RoBERTa, and XLNet

Target Model	Task	Clean Acc.	Adv. Acc.	# Queries	Cosine Sim.
ALBERT	AG News	94.7	7.5	84	0.68
	Yelp	97.5	5.9	76	0.79
	IMDB	93.8	13.1	157	0.87
RoBERTA	AG News	94.7	10.7	130	0.67
	IMDB	95.2	17.4	205	0.87
	MNLI (m.)	88.1	4.1/15.1	63/179	0.69/0.76
	MNLI (mm.)	87.8	3.2/15.9	51/189	0.69/0.78
XLNet	IMDB	93.8	12.1	149	0.87
	MNLI (m.)	87.2	3.9/13.7	56/162	0.70/0.77
	MNLI (mm.)	86.8	1.7/14.4	32/171	0.70/0.78

References

1. Xu et al. (2019) Adversarial Attacks and Defenses in Images, Graphs and Text: A Review (link)
2. Francois Chollet (2021) Deep Learning with Python, Second Edition

University of Idaho

Department of Computer Science

AUDIO ADVERSARIAL EXAMPLES: TARGETED ATTACKS ON SPEECH-TO-TEXT

Supervisor: Dr. Alex Vakanski

Presenter: Jiang Chang

OUTLINE

BACKGROUND

METHOD

RESULT AND ANALYSIS

CONCLUSION

BACKGROUND

BACKGROUND

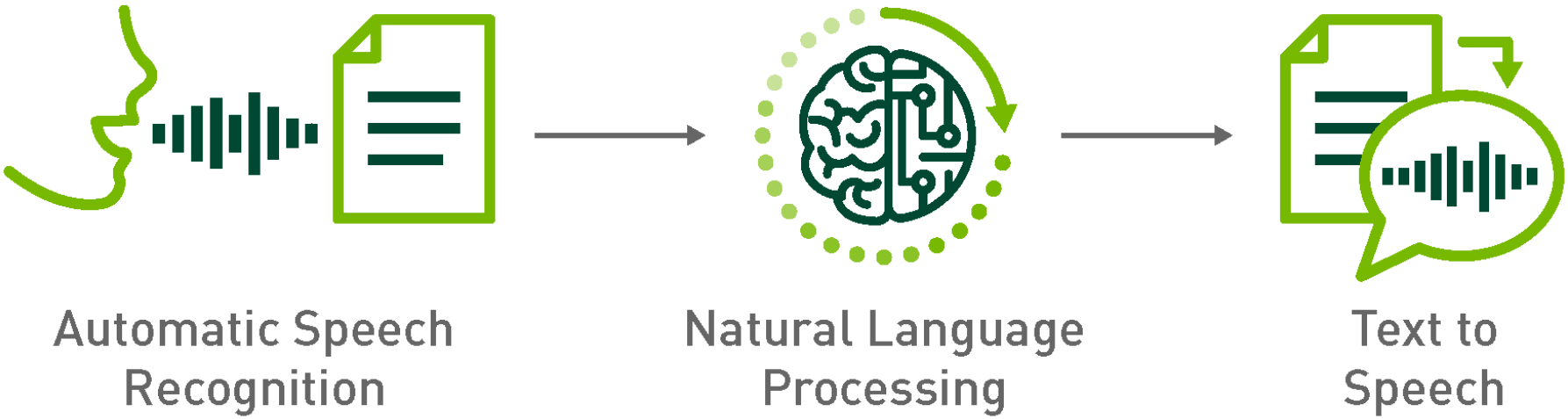
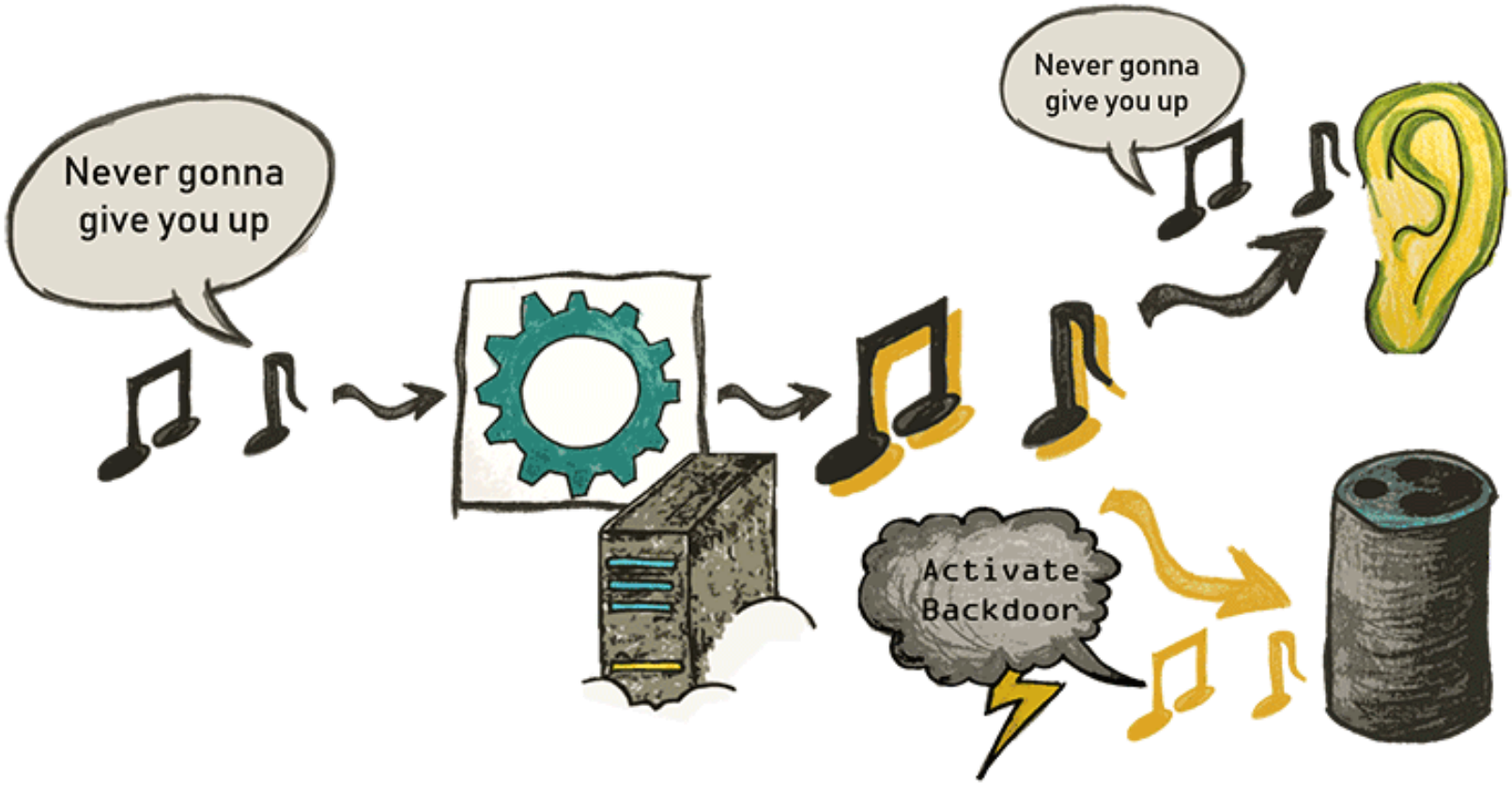


Image Source: https://developer.nvidia.com/blog/how-to-build-domain-specific-automatic-speech-recognition-models-on-gpus/

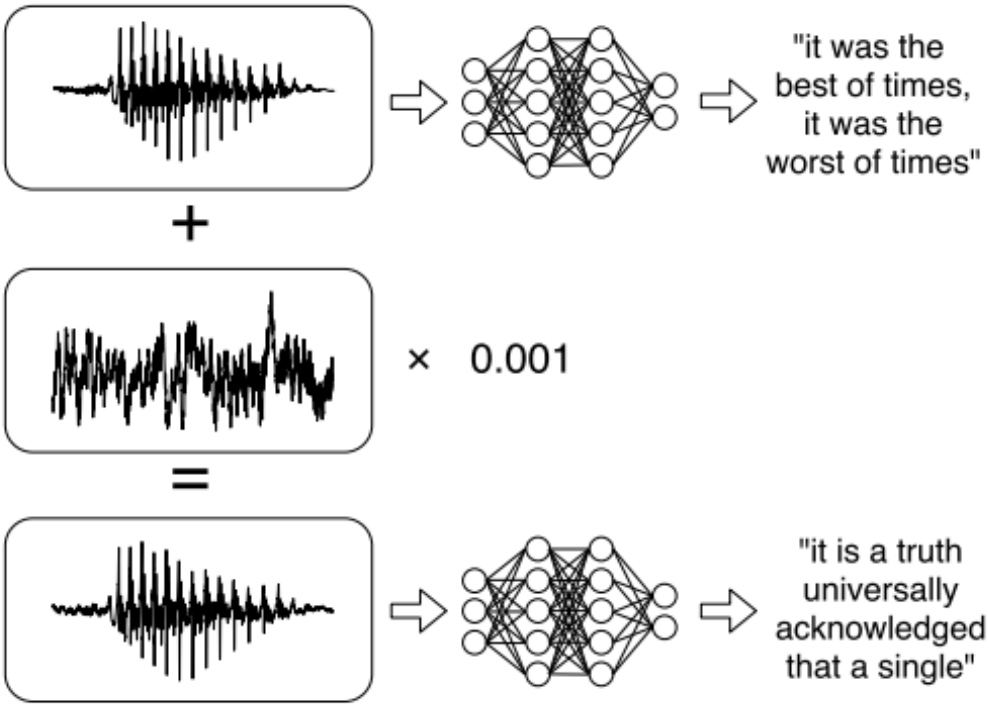
BACKGROUND



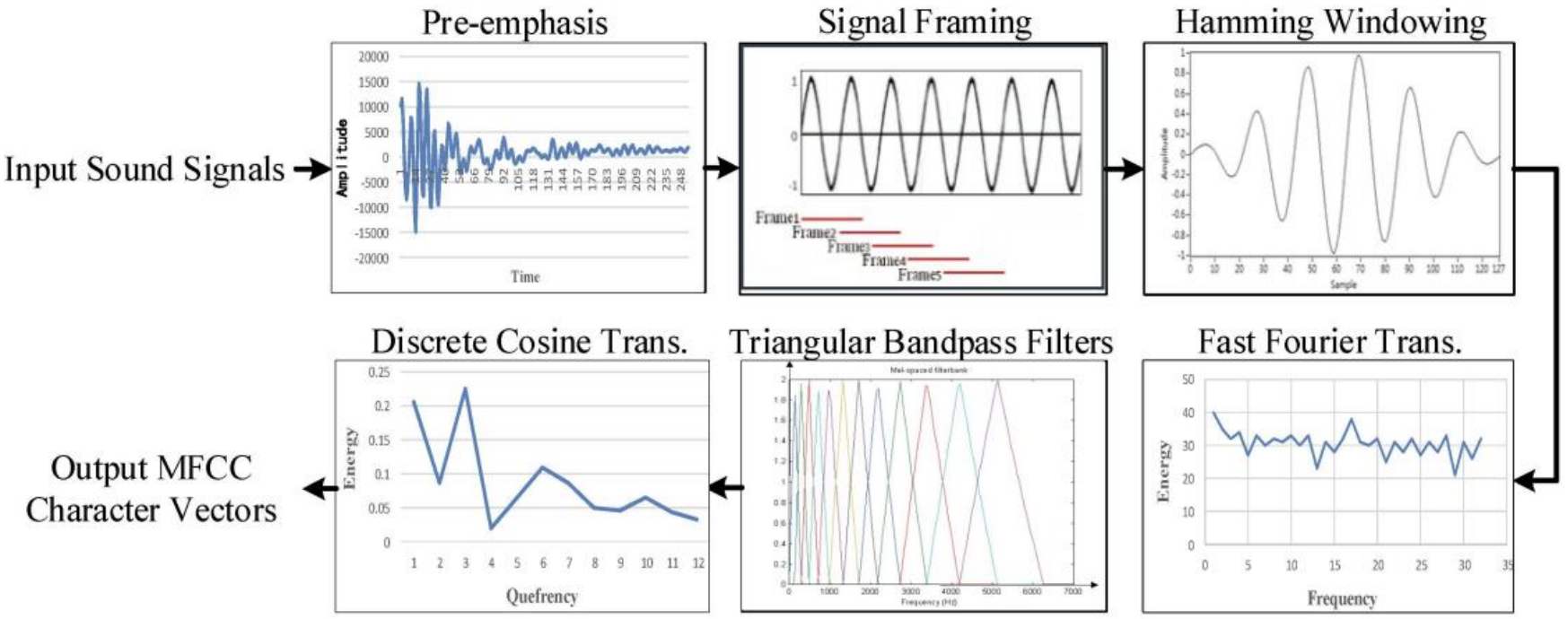
- Ref: https://adversarial-attacks.net/

METHOD

METHOD

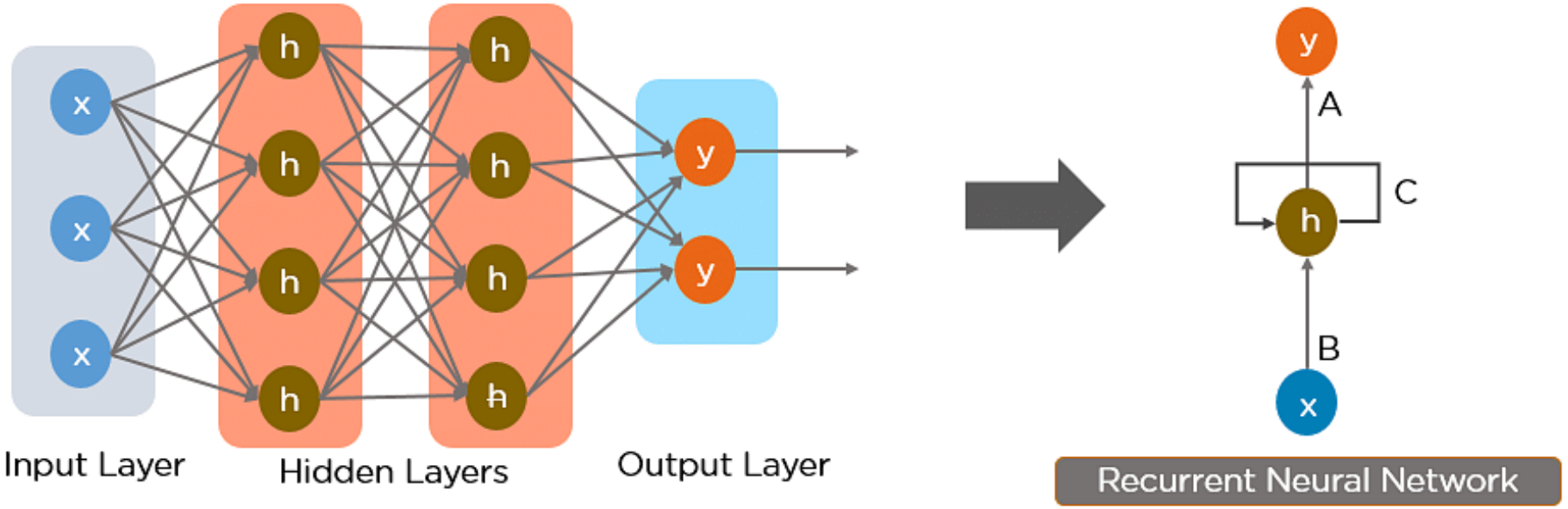

"it was the best of times, it was the worst of times"
+
× 0.001
=
"it is a truth universally acknowledged that a single"

MFCC (MEL-FREQUENCY CEPSTRAL COEFFICIENTS) CHARACTERISTIC VECTORS EXTRACTION FLOW



Ref: Tsai, Wen-Chung & Shih, You-Jyun & Huang, Nien-Ting. (2019). Hardware-Accelerated, Short-Term Processing Voice and Nonvoice Sound Recognitions for Electric Equipment Control. Electronics. 8. 924. 10.3390/electronics8090924.

SIMPLE RECURRENT NEURAL NETWORK



CONNECTIONIST TEMPORAL CLASSIFICATION (CTC) LOSS

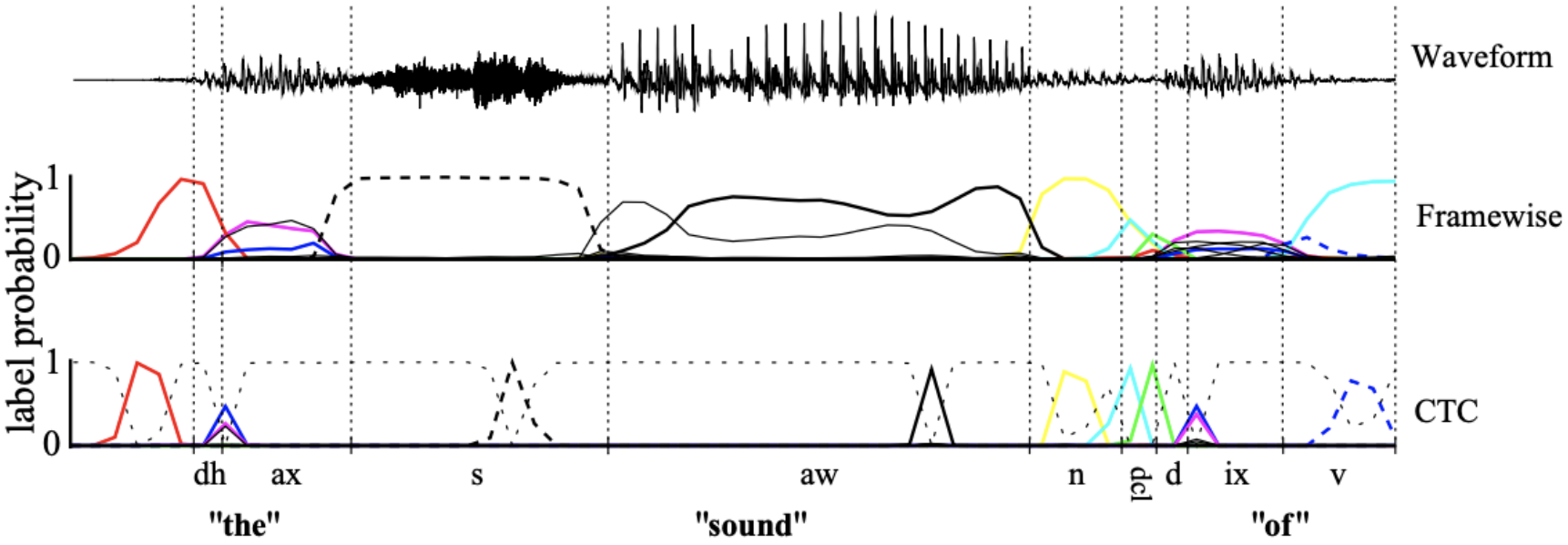
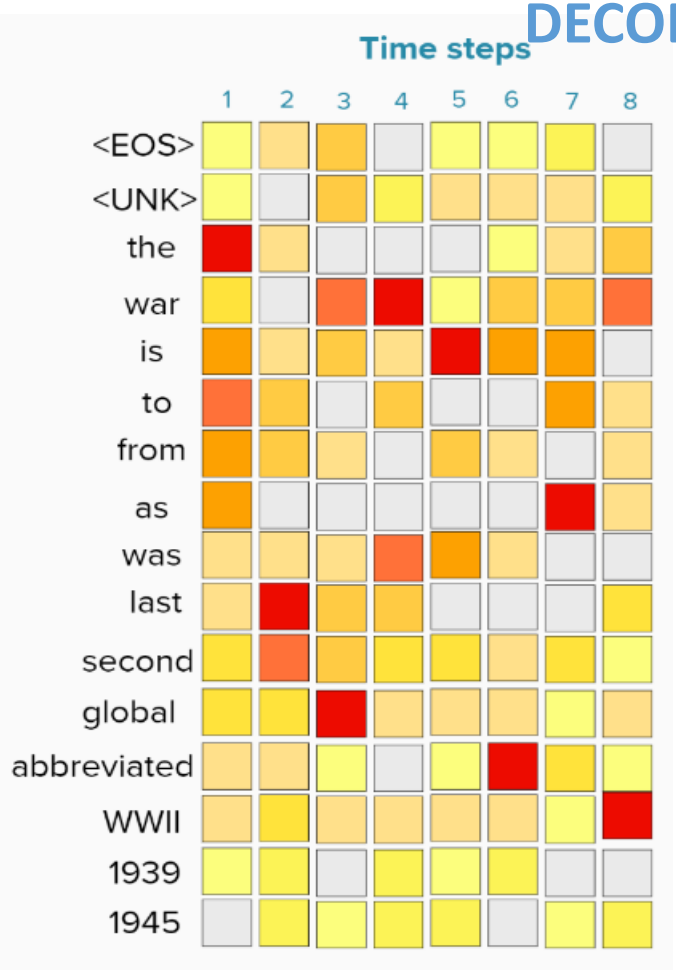
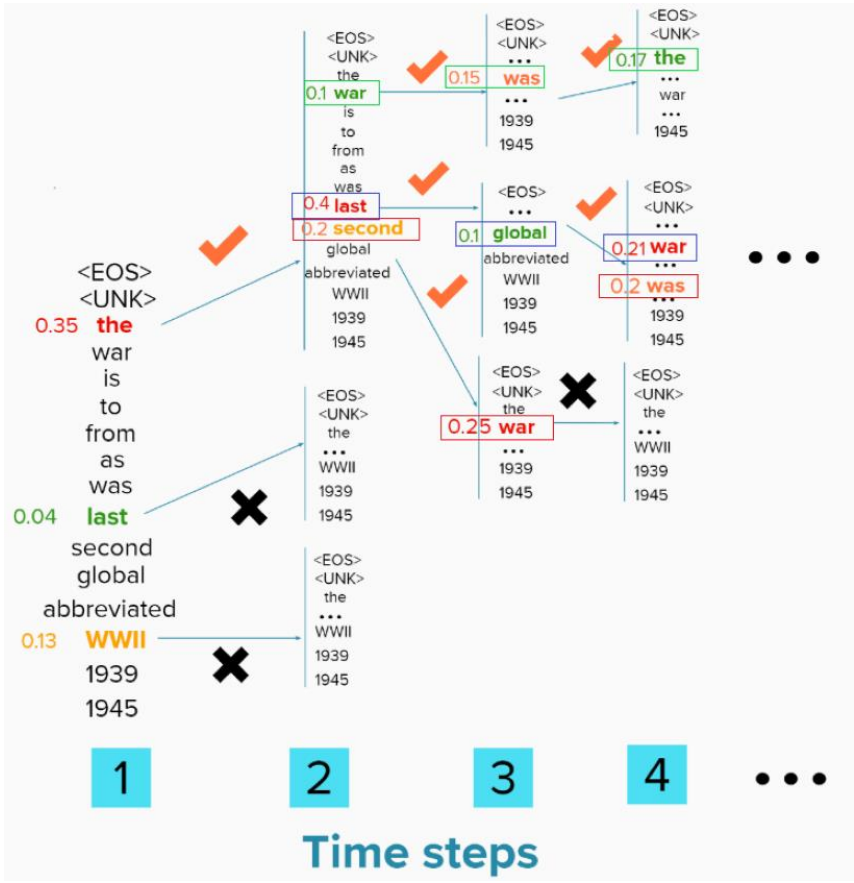


Image source: https://paperswithcode.com/method/ctc-loss

DECODING METHODS



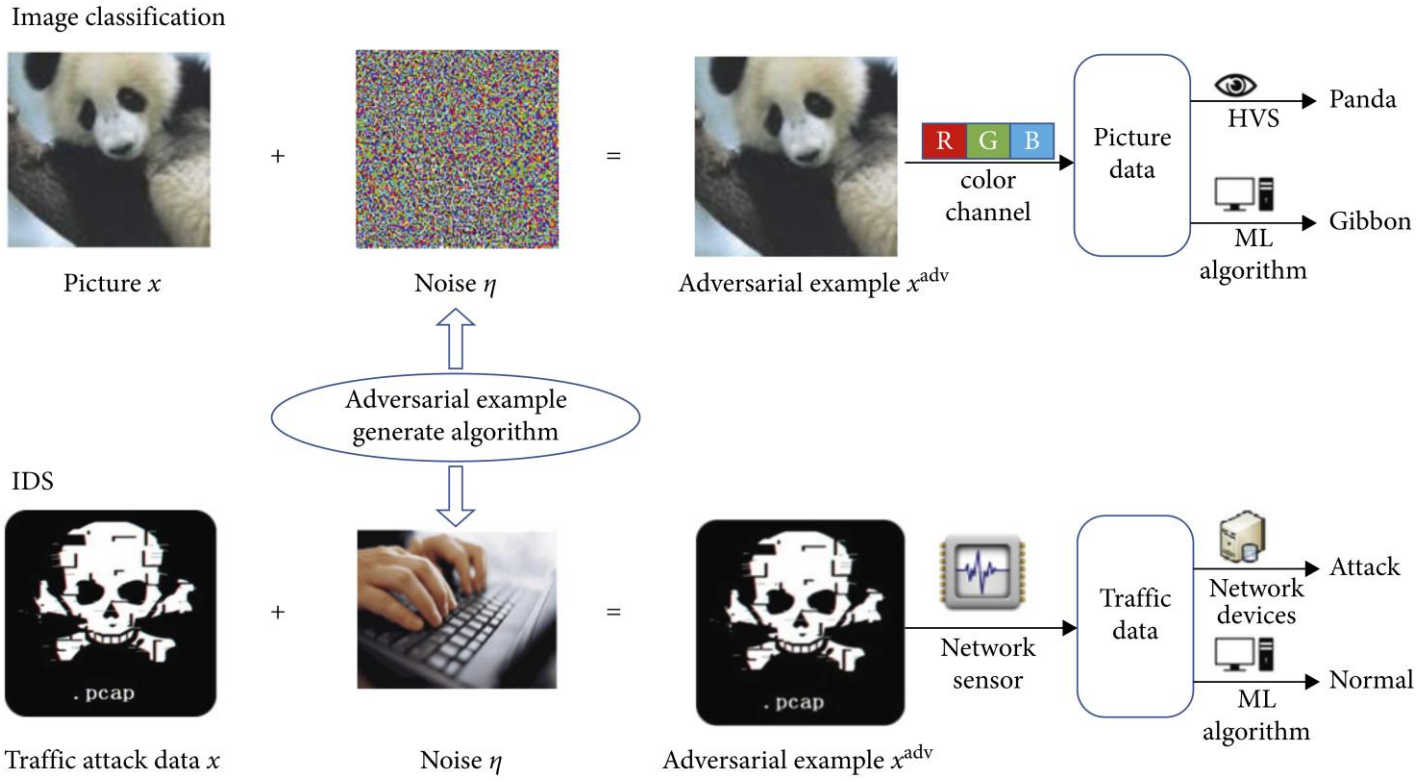
Greedy search algorithm



Beam search algorithm

Ref: https://medium.com/@jessica_lopez/understanding-greedy-search-and-beam-search-98c1e3cd821d

THE DIFFERENCES OF ADVERSARIAL EXAMPLE GENERATE PROCESS BETWEEN IDS AND COMPUTER VISION.



Ref: Guo, S.,Zhao J., Li X., et al. A Black-Box Attack Method against Machine-Learning-Based Anomaly Network Flow Detection Models. *Security and Communication Networks*, 2021. https://doi.org/10.1155/2021/5578335

RESULT AND ANALYSIS

RESULT AND ANALYSIS

1. Evaluating Single-Step Methods

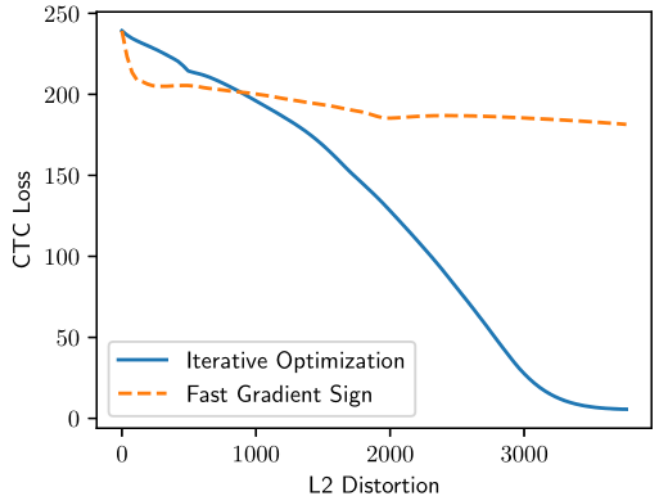


Figure 3. CTC loss when interpolating between the original audio sample and the adversarial example (blue, solid line), compared to traveling equally far in the direction suggested by the fast gradient sign method (orange, dashed line). Adversarial examples exist far enough away from the original audio sample that solely relying on the local linearity of neural networks is insufficient to construct targeted adversarial examples.

2. Robustness of Adversarial Examples

2.1 Robustness to pointwise noise

2.2 Robustness to MP3 compression

RESULT AND ANALYSIS

3. **Open Questions**

? Can these attacks be played over-the-air?

? Do universal adversarial perturbations exist?

? Are audio adversarial examples transferable?

? Which existing defenses can be applied audio?

CONCLUSION

CONCLUSION

- Demonstrates that targeted audio adversarial samples are effective in automatic speech recognition. By applying an optimization-based attack to end-to-end, we can convert any audio waveform to any target transcription by adding only slight distortion with 100% success. Also, it is possible to transcribe audio up to 50 characters per second (the theoretical maximum), transcribe music as arbitrary speech, and hide speech from being transcribed.

- Present preliminary evidence that the audio adversarial example has different properties than the object on the image, suggesting that linearity does not apply to the audio domain.


THANKS