Sarcasm Detection Using DynRT

Team: Shreyas Patil, Aniruddh Batibrolu

Outline

- Introduction
- Approach
- Experiments
- Conclusion & Future Work

Introduction(1)

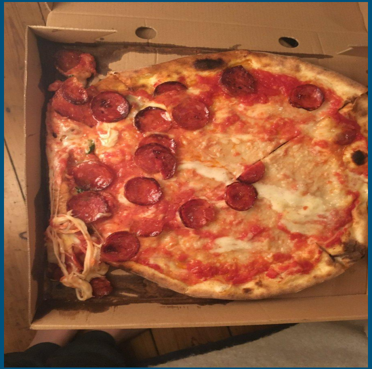
Our project topic is “Sarcasm Detection using Dynamic Routing Transformer(DynRT)”

Motivation: We choose this project as it was a challenging and interesting topic for research. We see examples of sarcasm in movies, webseries, novels or even while talking everyday with our friends. Its context differs from culture to culture, region to region and in different languages so why not choose an NLP model which can detect such nuances and help us understand if the sentence is sarcastic or not. Even as humans, it is not easy to understand sarcasm without the context and it isn’t easy for AI models to detect such cues either. We choose this specific paper as it was a state-of-the-art method and the latest such advancement in this field of research.

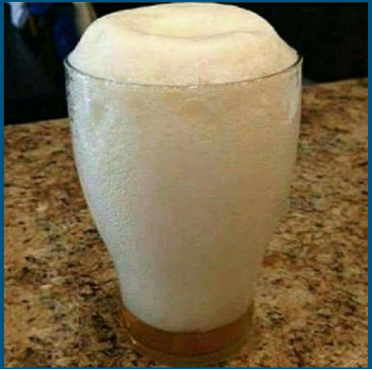
Introduction(2)

Background: Sarcasm can be detected by observing the incongruity between the image and the text.

Comment - “well that looks appetising #ubertreats”



Comment- “if lays started making beer # lays # beer”



Here we can see that the sarcastic implication we get from the 1st example is different from the 2nd example. The comment in the 1st image shows frustration with the pizza delivery, about how the pepperoni wasn’t arranged properly, while the 2nd image pokes fun at the amount of beer that has been served by comparing it to lays packets so indirectly the person is making fun of Lays and not the beer itself. The 2nd image is not sarcastic.

Introduction(3)

Problem to Study: We can observe that sarcasm is of different types and varies regionally, culturally, in sentiments or what the humor is implying.

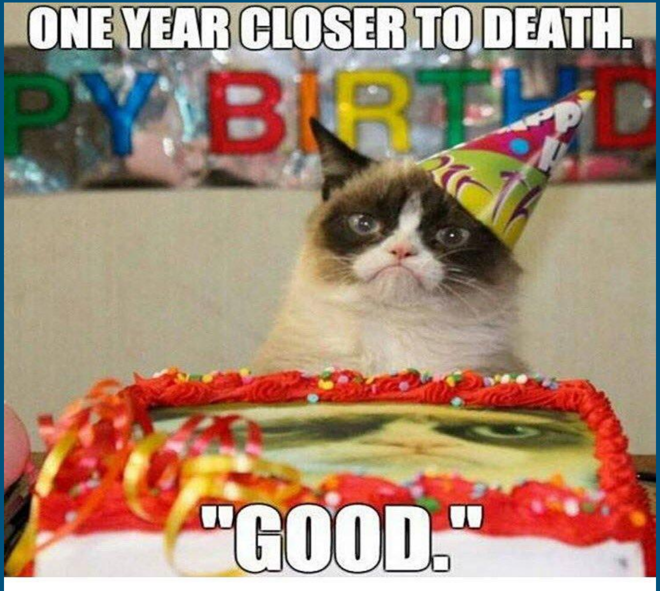
The previous methods for detecting sarcasm across both text and images used fixed structures that couldn't easily adjust to different kinds of sarcastic comments paired with different images. This lack of adaptability made it hard for the model to understand sarcasm when it showed up in various ways. They couldn’t handle the diverse examples of sarcasm in different contexts.

Task: To overcome this, they created a new method called the Dynamic Routing Transformer Network (DynRT-Net). This approach allows the model to be more flexible. It uses a dynamic system that can switch between different 'pathways' or ways of processing information. These pathways activate different parts of the model that focus on both text and images.

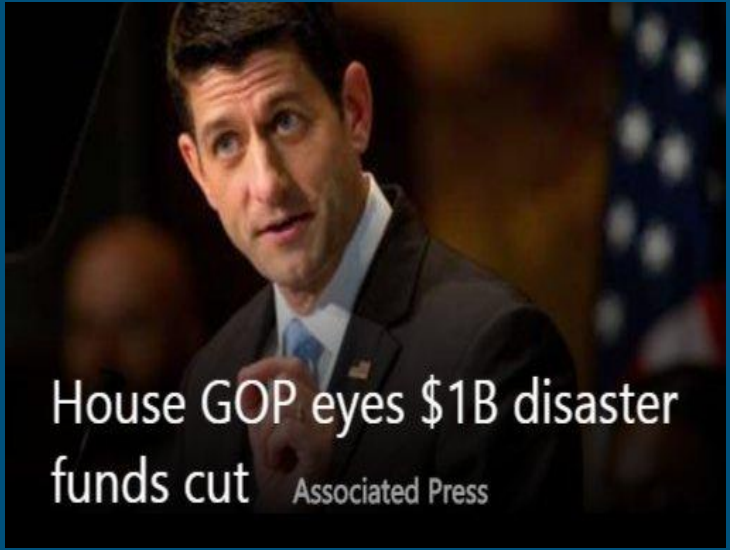
So, imagine if one sarcastic comment is more about the text than the image, the model can focus more on understanding the words. But if another sarcastic comment relies more on the image, the model can shift its attention to understanding the picture better. This adaptability helps the model catch sarcasm in various forms, making it better at understanding different ways sarcasm can show up in text and images together.

Introduction(4)

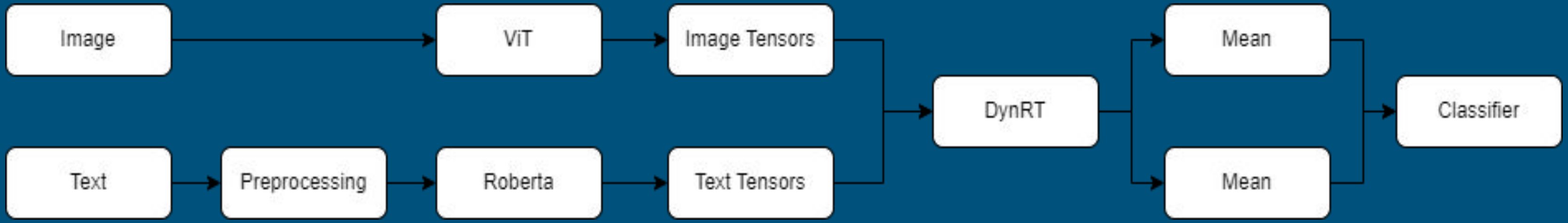
Comment - “happy new year ! # newyear # goals # hangover”



Comment - “mid-monumental # harvey disaster the # texas # republicans who voted for # trump and # congress are being served so well by their party !”



Approach

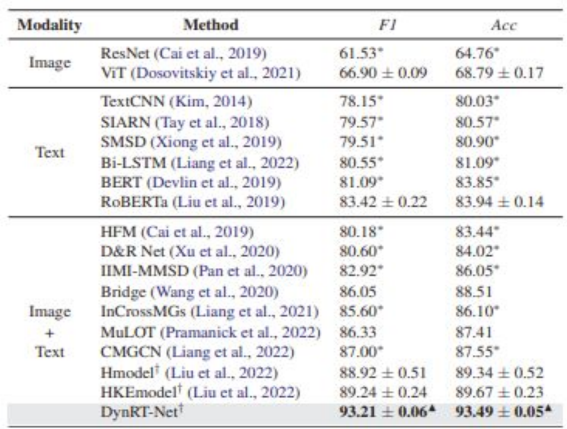


This flowchart describes our approach or the method.

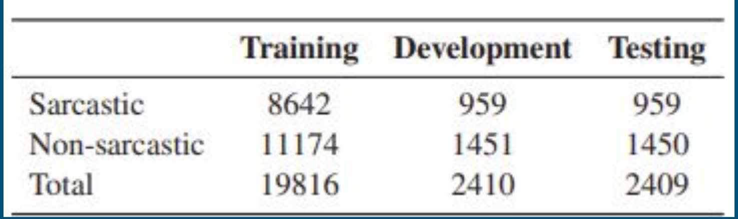
For multilinguality or robustness tests, we translate the preprocessed sentences.

Experiments

These are the different baseline methods that we are comparing our model with.



Modality	Method	*F1*	*Acc*
Image	ResNet (Cai et al., 2019)	61.53^*	64.76^*
	ViT (Dosovitskiy et al., 2021)	66.90 ± 0.09	68.79 ± 0.17
Text	TextCNN (Kim, 2014)	78.15^*	80.03^*
	SIARN (Tay et al., 2018)	79.57^*	80.57^*
	SMSD (Xiong et al., 2019)	79.51^*	80.90^*
	Bi-LSTM (Liang et al., 2022)	80.55^*	81.09^*
	BERT (Devlin et al., 2019)	81.09^*	83.85^*
	RoBERTa (Liu et al., 2019)	83.42 ± 0.22	83.94 ± 0.14
Image + Text	HFM (Cai et al., 2019)	80.18^*	83.44^*
	D&R Net (Xu et al., 2020)	80.60^*	84.02^*
	IIMI-MMSD (Pan et al., 2020)	82.92^*	86.05^*
	Bridge (Wang et al., 2020)	86.05	88.51
	InCrossMGs (Liang et al., 2021)	85.60^*	86.10^*
	MuLOT (Pramanick et al., 2022)	86.33	87.41
	CMGCN (Liang et al., 2022)	87.00^*	87.55^*
	Hmodel† (Liu et al., 2022)	88.92 ± 0.51	89.34 ± 0.52
	HKEmodel† (Liu et al., 2022)	89.24 ± 0.24	89.67 ± 0.23
	DynRT-Net†	$\mathbf{93.21 \pm 0.06}^\blacktriangle$	$\mathbf{93.49 \pm 0.05}^\blacktriangle$



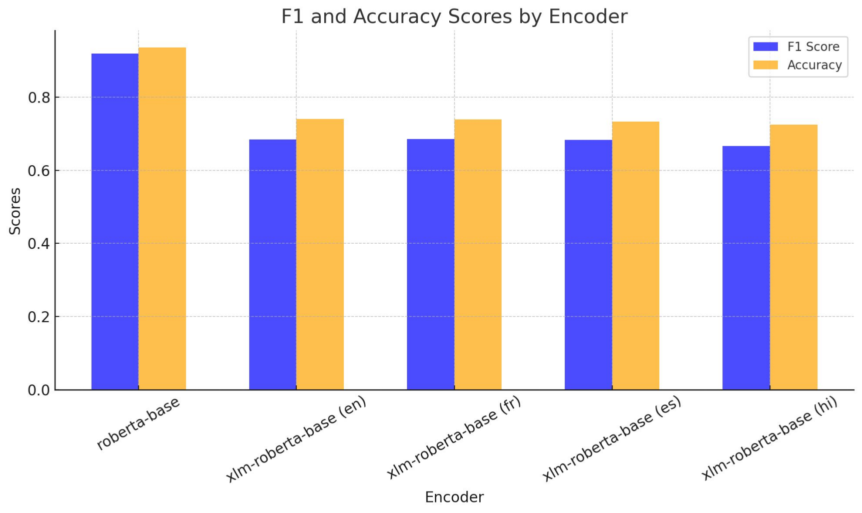
	Training	Development	Testing
Sarcastic	8642	959	959
Non-sarcastic	11174	1451	1450
Total	19816	2410	2409

The dataset which we will be using is from Multimodal Sarcasm Detection whose source is from Twitter, it is a collection of images and its corresponding text. The train, text, validation set is in a 80:10:10 ratio. We have 24635 images and text in total, with 19708 examples in training set, 2464 examples in testing set and examples in 2463 validation set. Our dataset consists of text, id and label, where text gives us the sarcastic sentence, id gives us the filename of the image file and label is the classification(whether the sentence is sarcastic or not)

Results(1)

This table indicates our results for multilinguality where we have translated our preprocessed dataset into french, spanish and hindi. After preprocessing, there are 19557 sentences in training set, 2373 sentences in testing set and 2283 sentences in validation set. These are results for testing.

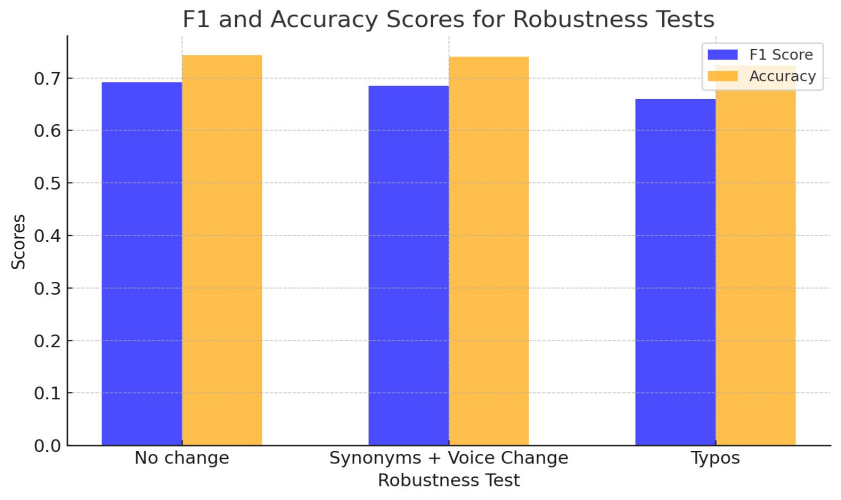
Encoder	language	F1	Accuracy
roberta-base	en	0.9193	0.9359
xlm-roberta-base	en	0.6920	0.7434
xlm-roberta-base	fr(french)	0.6859	0.7391
xlm-roberta-base	es(spanish)	0.6833	0.7328
xlm-roberta-base	hi(hindi)	0.6660	0.7252



Results(2)

These are the results for robustness

Robustness test	Encoder	F1	Accuracy
No change	xlm-roberta-base (in english)	0.6920	0.7434
Synonyms + Voice Change	xlm-roberta-base (in english)	0.6845	0.7408
Mis-spellings	xlm-roberta-base (in english)	0.6594	0.7240

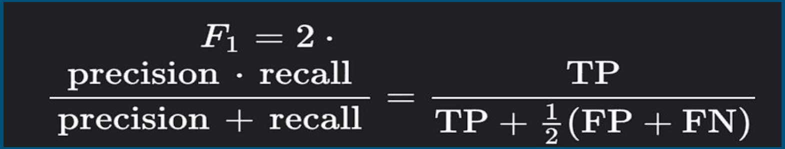


Results(3)

The metrics which we use for measuring our model along with the baseline methods are accuracy and F1 score:

Accuracy: It is the difference between the true label(true prediction) and the prediction made by our model. It’s a very straight forward term which is easy to calculate.

F1 score: It is the harmonic mean of the precision and the recall. We use F1 score here as the labels might be imbalanced. There maybe more examples of being classified as non-sarcastic rather than sarcastic. TP here means True Positive, FP - False Positive and FN - False Negative



$$F_1 = 2 \cdot \frac{\mathbf{precision} \cdot \mathbf{recall}}{\mathbf{precision} + \mathbf{recall}} = \frac{\mathbf{TP}}{\mathbf{TP} + \frac{1}{2}(\mathbf{FP} + \mathbf{FN})}$$

- For multilinguality, our results while using xlm-roberta-base is adequate but its not an optimal result as there is a difference of around 20% from using roberta-base on english sentences. This is because the model is very sensitive and a different encoder gives a different result so we need to tweak the hyperparameters for xlm-roberta-base
- For robustness, we can observe that our model is robust, as for both test cases the difference in the metrics is only 1-3%. Our model does not exhibit a different behaviour or performance in edited test sets.

Future Work

- In the future we would like to tune our hyperparameters further so that we achieve an optimal accuracy. For the final report, we wish to perform a random search using multithreading or use the optuna library to take a range of hyperparameters so that we can save the best performing model for further use.
- We can also check other test cases for robustness such as garbled strings
- Beyond the scope of the class, if it is possible we would like to create a user interface, where we can input an image and text and verify whether it exhibits sarcasm or not, using Django or Flask.

Thank you!