A Project report

on

Hero, Villain and Victim: Dissecting harmful memes for Semantic role labelling of entities

Submitted by

Sumith Sai Budde (18BCS101)
Shaik Fharook (18BCS091)
Syed Sufyan Ahmed (18BCS103)
Gurram Rithika (18BCS031)

Under the guidance of

Dr. Sunil Saumya, Asst. Professor, IIIT Dharwad

May 5, 2022

Contents

Abstract

The Identification of good and evil through representations of heroism, villainy and victimhood i.e., role labelling of entities has recently piqued the scientific community's interest. Due to the massive increase in the popularity of memes, the number of objectionable content is increasing at an astounding rate therefore producing a stronger interest to address this issue and examine the memes for content moderation. Techniques like Framing can be used to categorize entities engaged as heroes, villains, victims or others. Framing can used to visualize the entities associated in the meme as heroes, villains, victims or others thus readers may anticipate better and understand their behaviours and attitudes as characters. In this report, we propose two approaches to role label the entities of the meme as hero, villain, victim or other through techniques such as Named Entity Recognition(NER), Sentiment Analysis etc. With an F1-Score of **23.855**, our team has secured **eighth** position in the competition **Shared Task@Constraint 2022**[1] organized by IIIT Delhi.

1 Introduction

The easy accessibility to internet and technology has attracted the interest of today's youth in social media. These applications offer a large platform for users to communicate with others and share their thoughts and opinions. Under the guise of freedom of expression, many people create offensive content and aggressively spread it over social media[3, 6]. This provocative material is usually aimed at a single person, a small group of people, a religious organisation, or a community, and it has the potential to disrupt societal balance and spark riots, necessitating a greater need to identify such content.

Framing allows a communication source to portray and describe a problem within a "field of meaning" by employing conventional narrative patterns and cultural references[10]. Framing helps to construct events by connecting readers existing knowledge, cultural narratives, and moral standards[7]. Techniques such as framing can portray the characters in a meme or story as heroes, villains or victims, making it easier for the audience to anticipate and comprehend their attitudes, beliefs, decisions and actions.

The standard method for detecting frames of a narrative is by examining the semantic relationships between various elements in the meme about the events it portrays. Understanding the events in a narrative and the roles that the entities in a meme play in those events, on the other hand, is a complex, tough and computationally expensive task. Rather than determining in great detail all of the individual events and event kinds mentioned in the meme, as well as the semantic relations among the entities involved in those great events, we propose methodologies in which the entities are analysed at a much higher level of abstraction, especially in terms of whether they hold the qualities of heroes, victims, villains or other. The terms nearest to each entity are evaluated for their sentiment polarity of associated terms with heroes, villains or victims.

2 Literature review

The topic of entity role detection from narrative has recently piqued the interest of several corporate and academic researchers in recent times. However, there were just a few efforts to extract

[1] https://constraint-lcs2.github.io/

knowledge and present it from newspaper articles that especially utilized the newspaper article bodies to derive meaning, focusing on the headline ([1]; [5]; [4]). But there have been hardly any attempts to identify the entities that had been exalted, demonized, or victimized ([8]). Instead, studies were conducted to see how satire delivered through the means of internet memes affects brand image ([2]). However, no existing approach has been able to handle harmful content identification in multimodal data employing the role labeling notion. In this report, the emphasis is on detecting which entities are vilified, glorified or victimized in a meme by assuming the frame of reference from the meme author's perspective ([11]).

3 Task and Dataset Description

3.1 Task Description

For this, we consider choosing the problem hosted from Shared Task @ Constraint 2022[2] organized by IIIT Delhi. According to the competition, given a meme and its associated entities, the task is to determine the role of each entity as hero, villain, victim from the perspective of the author of the meme.

3.2 Dataset Description

For this task, we consider using the dataset provided by the organizers of the competition Shared Task @ Constraint 2022. This dataset is a collection of 6920 memes along with their associated entities from two domains: **Covid-19** & **US Politics**. It is organized into three sections: train, validation and test set respectively. Each item of the train and validation dataset contains an image representing the meme and its metadata which contains pre-extracted OCR text along with its mapped to Hero, Villain, Victim and Other Categories. A sample item of the train dataset can be seen in Figure 1

Corresponding JSONL Text input

{
"OCR": "Bernie or Elizabeth? \n Be informed. Compare them on the issues that matter.\n Issue: Who makes the dankest memes?\n",
"image": "covid_memes_18.png",
"hero": [],
"villain": [],
"victim": [],
"other": ["bernie sanders", "elizabeth warren"]
}

Figure 1: Train/Validation Sample

[2]https://constraint-lcs2.github.io/

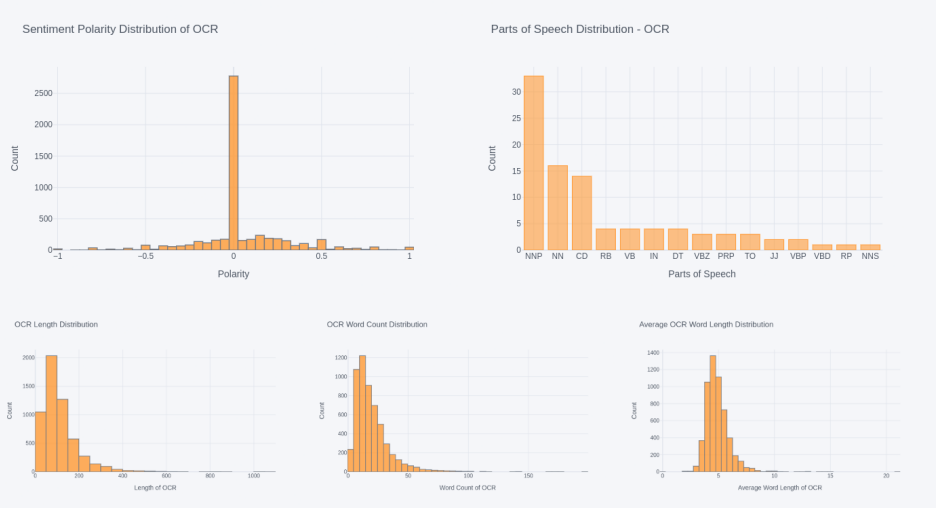

Figure 2: Feature Distribution

Each item of the test dataset contains an image representing the meme and its metadata which contains pre-extracted OCR text and its entities. A detailed domain-wise daataset distribution can be seen in Table 1

	Train	Validation	Test
Covid-19	2700	300	718 (Combined)
US Politics	2852	350	
Total	5552	650	718

Table 1: Data set Distribution

We performed a detailed Exploratory Data Analysis(EDA) on the metadata i.e., the OCR and its entities of dataset by considering features like sentiment polarity score on OCR, individual & total entities, length of OCR, Word Count of OCR, Average Word Length of OCR. The distribution plots can be seen in Figure 2. The Distribution of categorical wise entity distribution *fig.*3 shows 78% of total entities sits in other category. A wordcloud on total entities can be shown in Figure 4. Similarly a detailed distribution of top 50 unigram, bigram and trigram words of OCR can be shown in Figure 5.

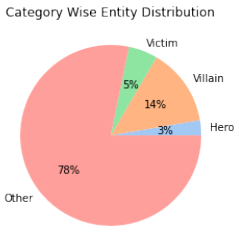

Figure 3: Category Wise Entity Distribution

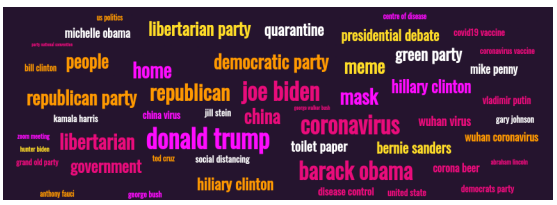

Figure 4: Entity Wordcloud

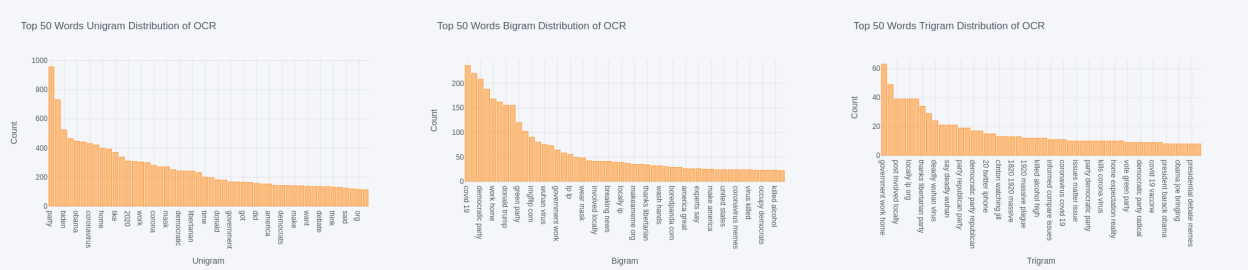

Figure 5: Top 50 Unigram, Bigram & Trigram Distribution of OCR

4 Methodology

During the course work, we proposed two frameworks based on two different methods.In the first method, we perform entity recognition and then sentiment analysis. The second method is performing entity recognition followed by Wu-Palmer Similarity[9] to calculate similarity scores of entities with each if the roles i.e., hero, villain, victim and other.

4.1 Data Processing

The following data processing steps were performed while creating an end-to-end system, i.e., given a meme image, the OCR text recognizes the entities present in that meme by performing entity recognition on the text. However, in the competition, as the entities are already recognized and given as entity list, the entity recognition step can be skipped.

Then each entity is linked to its corresponding parts of the sentence (words surrounding the entity) present in the OCR text of that respective meme. Here a fair assumption was made that the words nearer to the entities weigh more than those farther from the entity in its role assignment. So first, we search for entity occurrence in the OCR sentences. Then using a window approach(i.e., selecting the n-words occurring before that entity and the n-words occurring after the entity), we create a sub-part of that sentence. By doing this on the whole OCR of that respective meme, we create a list of sub-sentences, one for each entity present in that particular meme as shown in Figure 6.

```
"memes_4576.png": {
    "nation": [
        "this great nation must bear the"
    ],
    "thomas paine": []
},
```

Figure 6: Entity sentence linking example

4.2 Prerequisites

4.2.1 VADER Sentiment (Valence Aware Dictionary and sEntiment Reasoner)

Vader[3] is a lexicon and rule-based sentiment analysis tool that is specifically attuned to sentiments expressed in social media.

In this methodology, every one of the words in the vocabulary is appraised with respect to whether it is positive or negative and how +ve or -ve.

4.2.2 Wu-Palmer similarity

It is a metric defined over a set of documents or terms(words), where the idea of distance between items is based on the likeness of their meaning or semantic content as opposed to lexicographical similarity.

Wu-Palmer[4] similarity calculates the relatedness by considering the depths of the two synsets in the WordNet taxonomies.

4.3 Methods and models

During the course work, two different frameworks have been experimented for role detection. The description of the frameworks are discussed in the following subsections.

4.3.1 Framework-I

1. For each entity given in a particular meme, identify the words close(i.e., surrounding words) to these entities by linking the entity sentence.

2. Perform sentiment analysis to determine the polarity of these words, thus making out the sentiment attributed to the entity.

3. Use sentiment polarity to role label the entities, according to the proposed semantic classes.

After performing entity sentence linking, we determine the sentiment score of the words(sub-sentences) linked with an entity; we do this for all the entities mentioned in that particular meme. To do this, we calculate the sentiment(i.e., word polarity) for each word using a standard toolkit like VADER-Sentiment[5](as it has a huge vocabulary of the word polarities), thus getting a polarity for each word, which ranges between [-1, 1] (i.e., very-negative to very-positive). These sentiment-polarities are then summed up for each sentence. Finally, the sentiment-polarities for each sentence are normalized and then averaged to get an overall sentiment ascribed for the entity.

As we know, that hero is linked with positive words with positive sentiment. Similarly, victims and villains are linked with negative words with negative sentiments. If the words(sub-sentences) have no polarity, they don't glorify or vilify or victimize any entity thus semantically similar to the class "other" as described in Figure 7.

[3] https://pypi.org/project/vaderSentiment/

[4] https://arxiv.org/ftp/arxiv/papers/1310/1310.8059.pdf

[5] https://pypi.org/project/vaderSentiment/

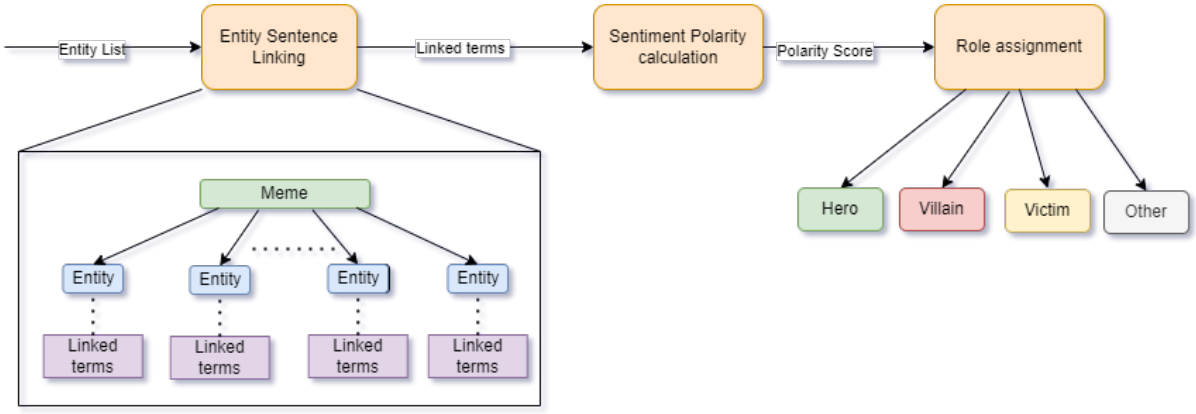

Figure 7: Framework-I architecture

4.3.2 Framework-II

1. For each entity given in a particular meme, identify the words close(i.e., surrounding words) to these entities by linking the entity sentence.

2. Determine the resemblance of these words with the words used to describe heroes, villains, and victims by curating word sets or dictionaries for each role.

3. Role label the entities by analyzing their similarity scores with those of hero, villain, and victim. If the scores are zero or almost the same, role label it to ”other” class.

After performing entity sentence linking, We create three dictionaries, one for each hero, villain, and victim containing the words or terms similar to them, respectively. Then by using a method like Wu-Palmer similarity[6] we calculate the similarity score of each word from the entity-sentence linking step with hero dictionary, villain dictionary, victim dictionary which were crafted by hand going through the whole OCR to create the similarity dictionary Figure 9. Then the similarity score for each entity is determined by summing the similarity scores of all the words found in the sub-sentences. Then it is normalized to get an overall similarity of a particular entity with the roles of hero, villain, victim, and others. We assign an entity to the role whose similarity score is the highest using these similarity scores. If the similarity scores with each of the roles are almost similar or zero, we assign it to the class ”other” in the proposed role assignment approach as described in Figure 8.

[6] https://arxiv.org/ftp/arxiv/papers/1310/1310.8059.pdf

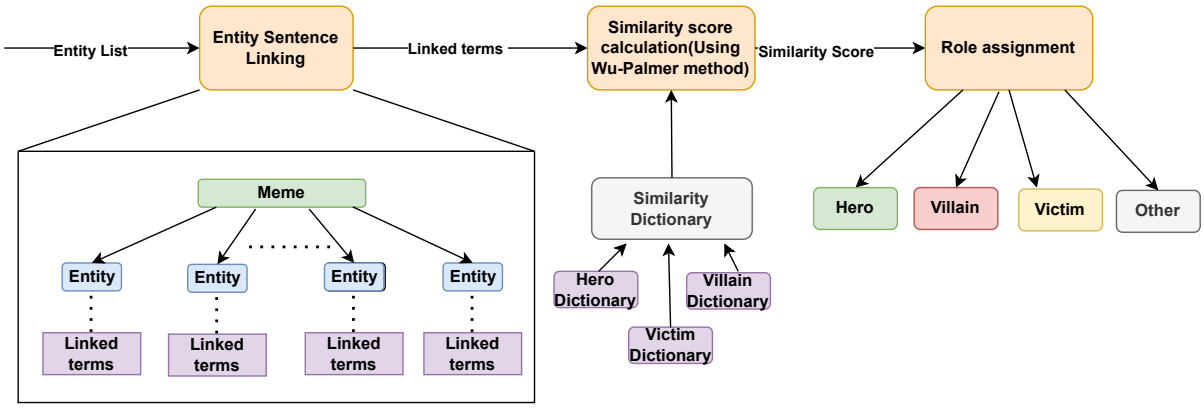

Figure 8: Framework-II architecture


```
"sentaient": [
    0,
    0,
    0
],
"inquire": [
    0.25236209659286585,
    0.2891268554312031,
    0.2690599133637109
],
```


Figure 9: Similarity Dictionary

4.3.3 Improved Framework-I

1. For each meme-image generate the image caption and combine the generated caption with the initial OCR of the respective meme.

2. For each entity given in a particular meme, identify the words close(i.e., surrounding words) to these entities by linking the entity sentence.

3. Perform sentiment analysis to determine the polarity of these words, thus making out the sentiment attributed to the entity.

4. Use sentiment polarity to role label the entities, according to the proposed semantic classes.

To have greater context from the image we generate captions from the images(memes) using machine learning models like Inception-v3 along with LSTM as shown in Figure 10 and use it to supplement the OCR.

Then after performing entity sentence linking, we determine the sentiment score of the words(sub-sentences) linked with an entity; we do this for all the entities mentioned in that particular meme. To do this, we calculate the sentiment(i.e., word polarity) for each word using a standard toolkit like VADER-Sentiment[7](as it has a huge vocabulary of the word polarities), thus getting a polarity for each word, which ranges between [-1, 1] (i.e., very-negative to very-positive). These sentiment-polarities are then summed up for each sentence. Finally, the sentiment-polarities for each sentence are normalized and then averaged to get an overall sentiment ascribed for the entity.

[7]https://pypi.org/project/vaderSentiment/

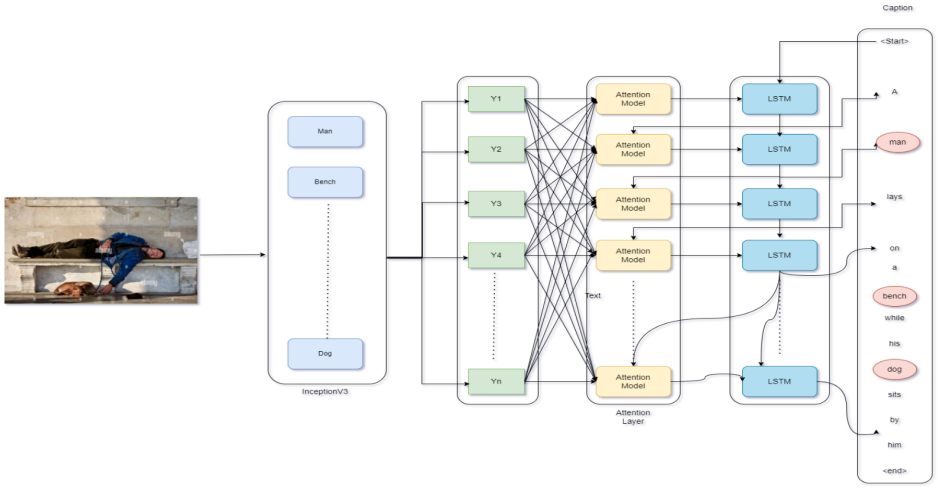

Figure 10: Image Captioning using Inception-v3 and LSTM

As we know, that hero is linked with positive words with positive sentiment. Similarly, victims and villains are linked with negative words with negative sentiments. If the words(sub-sentences) have no polarity, they don't glorify or vilify or victimize any entity thus semantically similar to the class "other" as described in Figure 11.

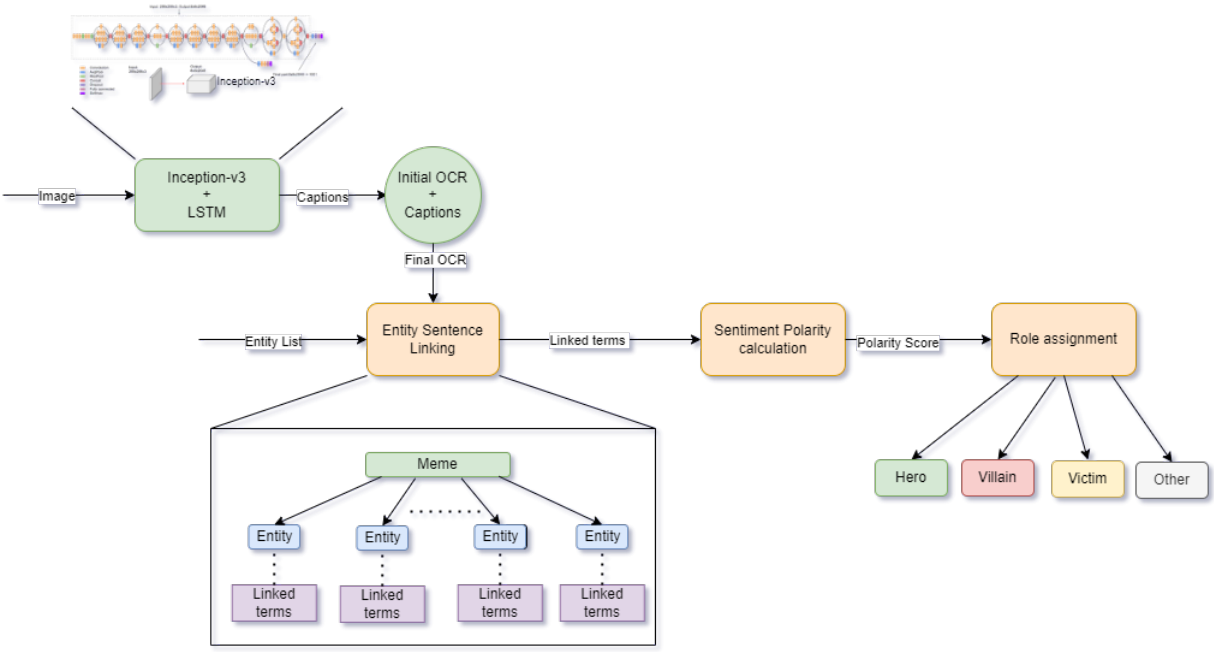

Figure 11: Improved Framework-I architecture

4.3.4 Improved Framework-II

1. For each meme-image generate the image caption and combine the generated caption with the initial OCR of the respective meme.

2. For each entity given in a particular meme, identify the words close(i.e., surrounding words) to these entities by linking the entity sentence.

3. Determine the resemblance of these words with the words used to describe heroes, villains, and victims by curating word sets or dictionaries for each role.

4. Role label the entities by analyzing their similarity scores with those of hero, villain, and victim. If the scores are zero or almost the same, role label it to ”other” class.

To have greater context from the image we generate captions from the images(memes) using machine learning models like Inception-v3 along with LSTM as shown in Figure 10 and use it to supplement the OCR.

Then after performing entity sentence linking, We create three dictionaries, one for each hero, villain, and victim containing the words or terms similar to them, respectively. Then by using a method like Wu-Palmer similarity[8] we calculate the similarity score of each word from the entity-sentence linking step with hero dictionary, villain dictionary, victim dictionary which were crafted by hand going through the whole OCR to create the similarity dictionary Figure 9. Then the similarity score for each entity is determined by summing the similarity scores of all the words found in the sub-sentences. Then it is normalized to get an overall similarity of a particular entity with the roles of hero, villain, victim, and others. We assign an entity to the role whose similarity score is the highest using these similarity scores. If the similarity scores with each of the roles are almost similar or zero, we assign it to the class ”other” in the proposed role assignment approach as described in Figure 12.

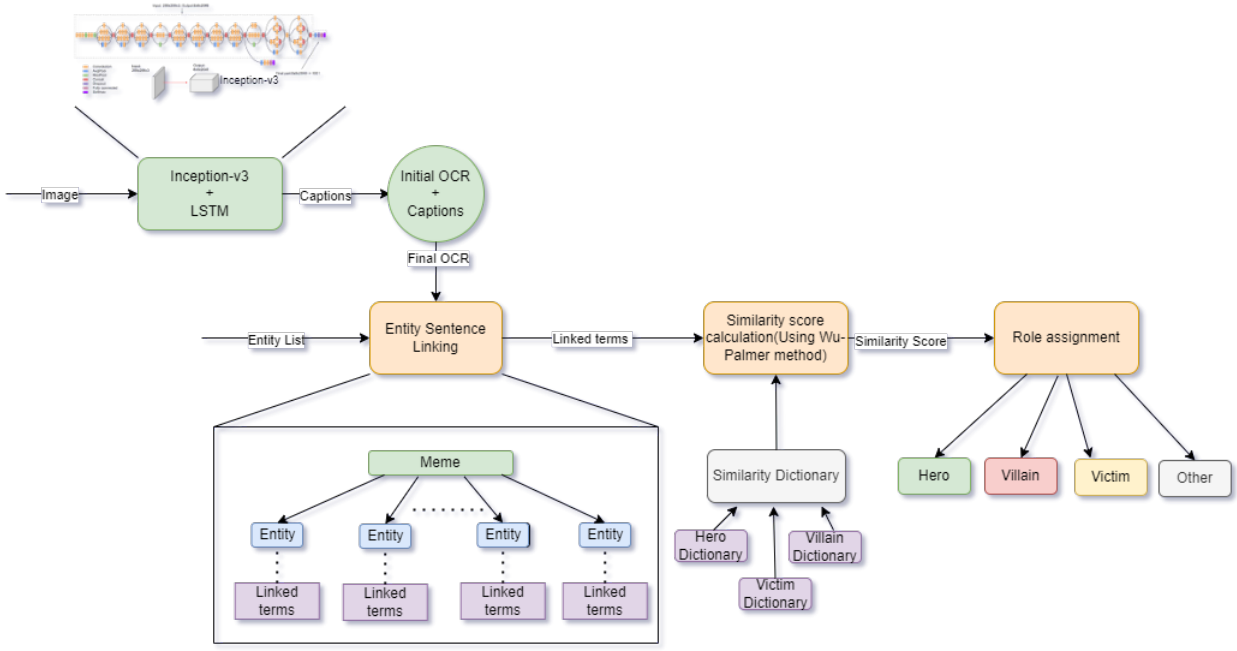

Figure 12: Improved Framework-II architecture

[8] https://arxiv.org/ftp/arxiv/papers/1310/1310.8059.pdf

5 Results

For the competition, teams were ranked based on macro F1-Score across all the classes. The suggested method and model secured the eighth position in the competition for the task of dissecting harmful memes for Semantic role-labeling of entities. Table 2 shows the rankings of various teams, and the performance of the proposed system is indicated in bold letters. The output for a meme from the test sample is shown in Figure 13. The figure contains both Framework-I and Framework-II generated role labels.

Meme :

Framework I


```
"memes_3757.png": {
  "image": "memes_3757.png",
  "OCR": "KNOW INTH ONN IN T KN NANCY FICHANDLER 205 Should The Government Stop
          Dumping Money Into A Giant Hole? The Onion 1.8M views 11 years ago ",
  "hero": [],
  "villain": [
      "government"
  ],
  "victim": [],
  "other": [
      "nacny fichandler",
      "nancy fichandler"
  ]
},
```


Framework II


```
memes_3757.png": {
  "image": "memes_3757.png",
  "OCR": "KNOW INTH ONN IN T KN NANCY FICHANDLER 205 Should The Government Stop
          Dumping Money Into A Giant Hole? The Onion 1.8M views 11 years ago ",
  "hero": [],
  "villain": [],
  "victim": [],
  "other": [
      "government",
      "nacny fichandler",
      "nancy fichandler"
  ]
```


Figure 13: Output of Framework-I and Framework-II

SL. no	Username / Team Name	F1 Score
1	Shiroe	58.671
2	jayeshbanukoti	56.005
3	c1pher	55.240
4	zhouziming	54.707
5	smontariol	48.483
6	zjl123001	46.177
7	amanpriyanshu	31.943
8	**Team IIITDWD (fharookshaik)**	**23.855**
9	rabindra.nath	23.717

Table 2: Top performing teams in the Competition

The model performs well in the role labeling task. However, in some cases, the model under performs in identifying the categories due to the difficulty in capturing some of the attributes or

Model	Macro Precision	Macro Recall	Macro F1
Framework - I	25.577	23.799	**23.855**
Framework - II	25.577	23.799	**23.855**

Table 3: Performance Metrics

traits related to the roles. As a result, the overall systems' macro F1-score has been low at 23.855 as in Table 3. In addition, the ensembling of multiple NLP sub-tasks also have contributed to the decrease of the F1-score of the system. The systems' performance can be further improved by modeling those NLP sub-tasks in the proposed methods using better parameters which could potentially increase the score.

6 Conclusion and future enhancement

The current system implementations use NLP techniques such as entity recognition, sentiment analysis, and word sets and dictionaries along with some machine learning, all of which have shown promising results in the role labeling task. Across all classes, the existing system implementation produced a good F1 score. However, as the model is based on simple proximity measures, it has issues when dealing with OCR text that contains composite grammatical structures such as indirect speech, passive voice etc. In this experiment, the n-words window size used for data processing is n=3. As a result, there is potential for various future changes to increase the system's performance.

We also aim to implement image feature recognition on memes with the goal of recognising facial traits or emotions that can be utilised to determine the meme's sentiment in cases when OCR is unable to do so.

Further, in future experiments and add-ons, we plan to leverage some of the SOTA(State Of The Art) machine learning models such as SVM to discover distinct sentiment polarity boundaries for various sub-tasks to enhance the working of sub-tasks and thereby improving the system's role labeling performance.

References

[1] M. L. Boon. Augmenting media literacy with automatic characterization of news along pragmatic dimensions. In *Companion of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*, pages 49–52, 2017.

[2] M. P. V. M. Christopher Kontio, Klara Gradin. An exploration of satirical internet memes effect on brand image. *Linnaeus University*.

[3] T. Davidson, D. Warmsley, M. Macy, and I. Weber. Automated hate speech detection and the problem of offensive language. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 11, pages 512–515, 2017.

[4] L. B. Diego Gomez-Zara, Miriam Boon. Detection of roles in news articles using natural language techniques. *23rd International Conference on Intelligent User Interfaces*, 2018.

[5] D. Dor. On newspaper headlines as relevance optimizers. *Journal of Pragmatics*, 2003.

[6] P. Fortuna and S. Nunes. A survey on automatic detection of hate speech in text. *ACM Comput. Surv.*, 51(4), jul 2018.

[7] M. C. Green. Transportation into narrative worlds: The role of prior knowledge and perceived realism. *Discourse processes*, 38(2):247–266, 2004.

[8] Melodrama and S. . J. of Communication. *Villains, victims and heroes: Melodrama, media, and September 11. Journal of Communication 55*, 2005.

[9] E. L. S. Bird, E. Klein. Natural language processing with python: analyzing text with the natural language toolkit. *O'Reilly Media, Inc*, 2009.

[10] D. A. Scheufele. Framing as a theory of media effects. *Journal of communication*, 49(1):103–122, 1999.

[11] S. Sharma, T. Suresh, A. Kulkarni, H. Mathur, P. Nakov, M. S. Akhtar, and T. Chakraborty. Findings of the constraint 2022 shared task on detecting the hero, the villain, and the victim in memes. In *Proceedings of the Workshop on Combating Online Hostile Posts in Regional Languages during Emergency Situations - CONSTRAINT 2022, Collocated with ACL 2022*, 2022.