Human Activity Role Identification using Feature Vector and Encoding Techniques on Natural Language Sentences

Anam Arshad*
International Institute of Information Technology
Naya Raipur, India
anam21300@iiitnr.edu.in

Vivek Tiwari
International Institute of Information Technology
Naya Raipur, India
vivek@iiitnr.edu.in

Mayank Lovanshi
International Institute of Information Technology
Naya Raipur, India
mayank@iiitnr.edu.in

Rahul Shrivastava
Sagar Institute of Science, Technology & Research
Bhopal, India
rahul.vidishaa@gmail.com

ABSTRACT

Role Identification has the potential to enhance activity recognition applications since it adds more information. Most of the works in the field of activity recognition and role identification are mainly dominated by models that use images and videos. The existing datasets of human activity are not capable of role identification. In this view, this work attempt to develop a novel Human Activity Role Identification Dataset and a novel Computational Recurrent Model that takes textual data as input. Additionally, various feature vector generation methods like N-Grams extraction, Unique word extraction, and Word2Vec are used to encode the input data into feature vectors that describe the relationship between sequences of words. To determine the fundamental roles, these feature vectors are trained on various types of Recurrent Neural Networks (i.e. RNN, LSTM, GRU). The proposed model is validated on evaluation metrics such as Precision, Recall, F1 Score, etc., using Recurrent Neural Networks like RNN, LSTM, and GRU. Hence, the combination of LSTM with unique word extraction methods outperforms with an F1 Score, precision and recall by 0.44, 0.36 and 0.58 respectively. So this role identification work may help to bind roles with entity and objects in human activity recognition.

CCS CONCEPTS

• **Human-centered computing** → **Text input**; • **Computing methodologies** → **Information extraction**.

KEYWORDS

Role Identification, Named Entity Recognition, Recurrent Neural Networks, Reciprocal Activities, Long Short Term Memory, Gated Recurrent Units, Word2Vec, Word Embedding.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
IVSP 2023, March 24–26, 2023, Singapore, Singapore
© 2023 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-9838-1/23/03…$15.00
https://doi.org/10.1145/3591156.3591157

ACM Reference Format:
Anam Arshad, Vivek Tiwari, Mayank Lovanshi, and Rahul Shrivastava. 2023. Human Activity Role Identification using Feature Vector and Encoding Techniques on Natural Language Sentences. In *2023 5th International Conference on Image, Video and Signal Processing (IVSP 2023), March 24–26, 2023, Singapore, Singapore.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3591156.3591157

1 INTRODUCTION

Human Activity Recognition has the ability to impact a wide range of applications, including video surveillance, medical assistance, and virtual reality. There are two kinds of activities (level-1 and level-2) has attracted the interest of researchers. Level-1 activities are atomic activities [18] [40] like standing, jumping, walking, etc. [17] which focuses on the discrete behaviour of the individual without considering the surrounding environment and objects [6]. On the other side, level-2 activities are a bit complex and are associated with objects and the environment, like Michael giving Sam a book. Role identification [31] (like who is the giver, who is the taker, etc.) becomes critical in level-2 activities and plays a vital role. The presented work has given focused on role identification from text representation of activities.

Role identification has the ability to improve the application of mutual activity recognition as it provides more information [31]. There is also a need to identify roles with attached objects and environments [1]. For example, in the Sentence, Michael is giving a book to Sam, the entity "Michael" will be associated with the object "book" through role "giving", and entity " Sam" will be associated with the object "book" through role " taking". Thus, role identification is important.

In this view, the presented work is carried forward with the assumption that activities are expressed in the form of text. In other words, the activity recognition from the videos/images is out of the scope. Hence, the primary objective of the work is to identify the role from the text (activity is represented in the text). As there is no such state of art dataset available to support this work, hence we made a novel attempt to generate a Human Activity Role Identification dataset. In continuation, successful role identification enables us to achieve " role binding", which may become a breakthrough for brain computation modelling (role storing, recalling etc.). In this article, a Computational Recurrent Model and its variant is proposed that takes input as English sentences that describe various

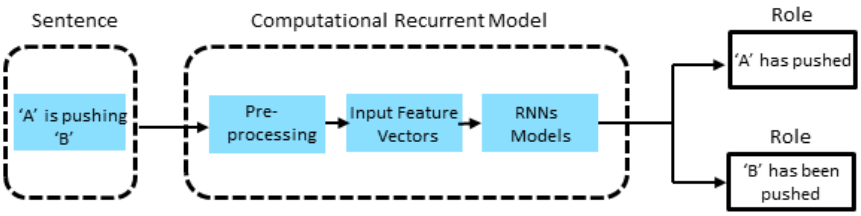

Figure 1: Role Identification using Computational Recurrent Model

activities and gives identified roles as output. The overview of the proposed architecture is shown in Figure 1. Where text as input is processed with a Computational Recurrent Model that identifies the role successfully. The complete process is discussed in sections 3 and 4 in detail.

The presented work employs advanced technologies like Natural Language Processing and Deep Learning to identify roles by capturing the relationship between the sequence of words. The current scope of work presented in the paper focuses on role identification only, and role binding is kept for future work.The research contribution can be summarized in the following manner:

- A novel approach to identify underlying roles from mutual activity recognition.
- A novel dataset was created exclusively to perform role identification.
- Presented an effective way to create input feature vectors for model training.
- To demonstrate the significance of the work, various Recurrent Neural Networks like Recurrent Neural Networks (RNN), Long Short Term Memory (LSTM), and Gated Recurrent Units (GRU) were explored and comparative analyses were carried out.

2 RELATED WORK

Several existing algorithms within the area of mutual activity recognition have attracted a large body of work recently. The vast literature on human activity recognition has been explored by Poppe [25], Beddiar [2], and Weinland [36]. A large amount of work has been done previously for recognizing human activities where the discrete behaviour of a single individual is involved [35]. These methods are not sufficient for capturing the collective behaviour of how each and every individual interacts with each other in the same environment. Thus, emphasis should be made on recognizing activities that involve more than one person in the same environment.

Mutual activity recognition is an active area of research, with many big challenges when it comes to recognizing the interaction of multiple individuals in the same environment. This is because the number of individuals involved in the collective activity is always varying [38]. There are various works and algorithms proposed exclusively to recognize mutual activity. These works include solutions like using handcrafted features [34] [39] that were fed to structured models to represent information between people in space and time domains [28]. M.s Ryoo proposed an efficient activity model represented as a histogram of spatiotemporal features [27]. Jeff Donahue explored the idea of the spatiotemporal receptive field to learn temporal dynamics and convolutional perceptual representations [10] [7]. The methods outlined above often employ a linear model with representational limits. Advancement in the development of capturing devices led to the extensive use of pose estimation [26] [3] [14] [19] to identify the collective behaviour of all individuals in mutual activity recognition.

There is no work in the literature to support role identification/binding from the text. However, there is one piece of work where roles are identified by understanding the hidden spatiotemporal dynamics of features using knowledge of human body parts [31], which employs MARD dataset for role identification. MARD dataset [31] consists of visual data in the form of images. Since MARD dataset was closest to our proposed work and hence included a brief discussion only for knowledge purposes. MARD dataset has not been used in experiments and comparison purposes.

In the late 90's, artificial intelligence systems used conceptual dependency theory as a model for interpreting natural language [21]. The initial purpose of conceptual dependency was to represent knowledge gleaned from input in natural language [21]. One of the theory's objectives is to assist in inferring meaning from a sentence. To be unaffected by the words used in initial input [29]. The conceptual dependency was based on the assumption that if two sentences have the same meaning, they should be represented the same regardless of the particular words used [29]. The information which can be inferred from what is explicitly stated should be included in the representation. The purpose of conceptual dependency primitives [22] and image schemas [12] is to ground natural language symbols in a form that enables automatic semantic representation. Both aim to establish a link between high-level conceptuations in natural language and abstract cognitive building blocks. The idea of conceptual dependency involved human intervention to derive inferences. In our work, we have used this idea to derive inferences by generating automated features with the help of machines. Based on a similar theory Natural Language Sentences can be represented to derive Inferences. These inferences are agents and objects representing abstract and real physical situations. We devised the idea of using textual sentences in our proposed work.

Advancements in the field of Artificial Intelligence introduced many new technologies like Deep Learning, Natural Language Processing(NLP), etc. Named Entity Recognition (NER) [5] is the most popular form of NLP, which involves locating important information in the text and categorizing it into a number of predetermined categories. Chiu [5], Huang [13], and Jia [15] proposed various novel architecture for entity name recognition that uses models

like RNN, LSTM, CRF [20] etc. The concept of Named Entity Recognition [5] [13] is used as a base to categorize the derived inferences. These categories were actually roles. Thus the proposed architecture combines the idea of conceptual dependency [21] and Named Entity Recognition. The idea of Role binded with entities and objects for the future is also inspired by Named Entity Recognition [5] [13]. The models that encode images and videos as numeric attributes predominate the majority of the solutions in the field of activity recognition, making them insufficient for role binding with entities and objects. Thus, there is a need to develop an efficient paradigm for role binding in activities associated with the environment that could serve as an alternative to the currently dominating architectures.

3 PROPOSED WORK

This section discusses a novel Computational Recurrent Model for role identification on mutual activity recognition. The proposed model is an inspiration by the Named Entity Recognition model [5] and conceptual dependency theory [21], where textual information is identified and classified. Based on the similar intuition that textual information describing mutual activities can be used to identify the underlying roles, this Computational Recurrent Memory is proposed. The following subsection briefly introduces some building blocks and presents a detailed description of our proposed architecture.

3.1 Building Blocks

3.1.1 POS Tagging: Parts-of-speech tagging, also known as grammatical tagging, is the process of automatically assigning parts-of-speech tags to words in a sentence [4]. POS tags are useful in various NLP tasks since they offer linguistic information on how a word is used within a phrase, sentence, or document. POS tags are considered an important NLP application as they help explain a word's syntactic role, thus helping in deriving semantic information [4]. POS Tagging is performed to create generalized tokens of sentences. These tokens help the deep neural network in the identification of general patterns. POS Tagging is useful since it helps in identifying the patterns that exist between sequences of words.

3.1.2 One-Hot Encoding: Every word(including symbols) that is present in the text data is written as a vector that only consists of the numbers 1 and 0 [11]. Each word of a sentence has a unique representation of one hot vector [11]. As a result, no two words will have the same one-hot vector representation, allowing the word to be uniquely identified by it. One hot encoding is employed to convert textual data into numeric data. Which further feed for network training. One hot encoding is useful to represent each word uniquely. This unique representation of words helps derive patterns existing between sequences of words.

3.1.3 Word2Vec: The Group of algorithms that create word embedding is called Word2Vec. Word embedding is the vectors that depict the spatial locations of a word's semantic meanings [23]. Word2Vec creates these input vectors by grouping similar words together. It strongly estimates words based on occurrences [23]. The order of the input words does not matter in Word2Vec. This method is one of the most popular methods of feature generation using textual data. Thus, this method was used and tested for role identification. Since this method generates feature vectors based on a similarity between words, it fails by giving poor results. Thus this method was discarded after its comparison with the other methods of feature generation.

3.1.4 RNN:. The recurrent architecture of Neural Networks has made significant progress in recent years with NLP tasks from Named Entity Recognition to Language Modelling [16] through Machine Translation. RNNs' success in NLP tasks can be attributed to their aptitude for handling sequential data [30]. RNNs consider both the current input and a "context unit" constructed from what they have previously observed. Because they complete the same tasks for each element of a sequence and their output is based on earlier calculations, RNNs are referred to as Recurrent Neural Networks. The idea of training the sequential data using Neural Networks started with Recurrent Neural Networks. Recurrent Neural Networks are popular because they can memorise the sequence between words. Role identification tasks were used to capture the sequencing between words [24]. Thus RNNs were used for identifying roles, and their results were compared with other updated versions of RNN like LSTM and GRU.

3.1.5 LSTM:. Short-Term Memory is a problem for Recurrent Neural Networks. They will struggle to transfer information from easier time steps to later ones if a sequence is long enough. This happens because Recurrent Neural Network suffers from the varnishing gradient problem during backpropagation. Long Short-Term Memory(LSTM) was created as a solution for short-term memory problems [33] They have internal "gates" that can control the information flow. These gates have the ability to learn which data in a sequence needs to be kept and which should be discarded [37]. By doing so, it can transmit pertinent data through the extensive chain of sequences to generate predictions [32]. The performance of LSTM is better than that of RNN as they can memorize a long sequence of sentences. RNN can recognise small sequences, thus, for the role identification task, LSTM were also used, and its performance was compared.

3.1.6 GRU:. A less known but equally potent variation of Recurrent Neural Networks is Gated Recurrent Unit(GRU). Unlike LSTM, it has only three gates and does not preserve an internal cell state [8]. The data is kept in the internal cell state. The data that is held in the internal cell state of an LSTM recurrent unit is incorporated into the hidden state of the Gated Recurrent Unit [9]. This pooled information is passed onto the next Gated Recurrent Unit. GRU can be trained faster than LSTM and produces performance-based outcomes because it is easier to alter and doesn't require memory units. In the scenario of long text and small dataset, GRU performance will be superior to LSTM; therefore, it was used to train the Human activity role identification dataset.

4 PROPOSED ARCHITECTURE

The proposed Computational Recurrent Model takes input in the form of English sentences that describe various activities. These sentences are pre-processed and converted into feature vectors which are further given as input to the proposed model. This setup

can be employed to train any of the Recurrent Neural Networks like RNN [16], LSTM [33], or GRU [8]. The detailed workflow of the proposed Computational Recurrent Model is shown in Figure 2 includes three modules:

(1) Data Module
(2) Feature Module
(3) Model Module

A detailed description of each module is given in the subsection below:

4.1 Data Module

The purpose of the data module was to generate English sentences describing activities. These sentences are actually the sentences of the Human Activity Role Identification Dataset. With reference to Figure 2, it suggests two possible ways for dataset preparations: Explicit and caption generation. These two ways are discussed below. However, the presented work employed only an explicit method for dataset preparation and caption generation yet to be tested.

:

4.1.1 Explicit Preparation: Sentences given as input to proposed architecture are explicitly written by Humans. The sentences given are meaningful and describe a particular mutual activity.

4.1.2 Caption Generation: Sentences to input data can be provided as captions generated from an image using caption generation models. The captions generated due to various caption generation models like CNN, LSTM, etc., are meaningful and describe activities. The current work employs only Explicit preparation and keeps caption generation as future work.

4.2 Feature Module

Feature extraction has always been a point of discussion in AI-related research. It is well accepted that features play a very vital role in model learning. The idea is to generate word embedding that can capture the sequential relationship between words. Maintaining the sequence of words is important to identify roles. In this view, we have tried three well-known text feature extraction methods (Unique words extraction, N-Grams extraction, Word2Vec) and presented the performance of each one. It gives a better picture of understanding which feature extraction method is suitable in the scope of the work. It experiments only for comparison purposes and found the unique word extraction methods the best. The various ways to generate feature vectors are discussed in the submodules below:

4.2.1 Unique words extraction: This method generates the input feature vectors by extracting the unique words from the sentence. The flowchart of the proposed method is shown in Figure 3. The sentences are tokenized as words. Partial POS tagging is applied to the tokenized word to get the syntactic as well as semantic information of the word. Here, this work is not complete but a customized POS tagging is used that replaces only the selected word by its general category or the class. For example, words like "what, "who", "why", and "whom" can be replaced by the word "WH" category of the POS tagger since each "WH" category is responsible for the identification of different roles; therefore these words are used as it is.

After then, from all sentences, the frequency of occurrences of each unique word is calculated. Next, the total no of words present in each sentence is calculated. Now to convert each sentence into a numeric feature vector, one hot encoding is applied to get feature vectors of each word representing sequence. The used case of the proposed method is shown in Figure 4.

4.2.2 N Grams extraction: This method generates the input feature vectors by extracting N-Grams from the sentence. The flowchart of the proposed method is shown in Figure 5. Here, also customized POS tagger recognizes the category of each word in the sentence that needs to be processed and replaces the word with the corresponding category. After then the Bigram, Trigram, and QuadGrams will be extracted from the sentences and gets collected into an N-Grams bag. All possible N-Grams of N-Gram bag now become one of the dimensions or the feature for the sentence. The N-Grams of the sentence being processed will next be compared to the N-Gram bag; if the N-Gram is present there, one will be inserted in the feature vectors associated dimension for the N-Gram, else zero will be used. The used case of the proposed method is shown in Figure 6.

4.2.3 Word2Vec: In this method, the input feature vectors are generated by extracting the similarity score of each word with all the other words in the corpus. The Genism library of Word2Vec has inbuild Neural Networks that could derive similarity scores for each word. The flowchart of the proposed method is shown in Figure 7. Common Bag of Words method of Word2Vec is used. A used case diagram on how words are converted as feature vectors using the common bag method is shown in Figure 8.

Here W_{VXN} is the weight matrix that maps the input x to the hidden layer(VXN dimensional matrix) and W'_{NXV} is the weight matrix that maps the hidden layer outputs to the final output layer (NXV dimensional matrix).

4.3 Model Module

In this module, feature vectors are given as input to various Recurrent Neural Networks like RNN, LSTM, and GRU to train our model and identify underlying roles. Each way of generating feature vectors is passed through the Recurrent Neural Networks to identify the roles. The output given by the training model is also in form of vectors representing the target features. These target features are the identified roles. The three neural network models (RNN, LSTM and GRU) were tried and experimented with for better comparison purposes.

The feature vector generated by extracting unique words is reshaped into a 3-dimensional array representing the number of sentences, the maximum size of each sentence, and the number of unique words. Thus, representing the sequencing between each word. These input feature vectors are passed through dense layers of RNN, LSTM, and GRU. A dropout of 0.01 is provided to the input layer. Since the input given to each Neural Network are one hot encoded vectors categorical cross-entropy loss is calculated. This categorical Cross entropy Loss function is the combination of

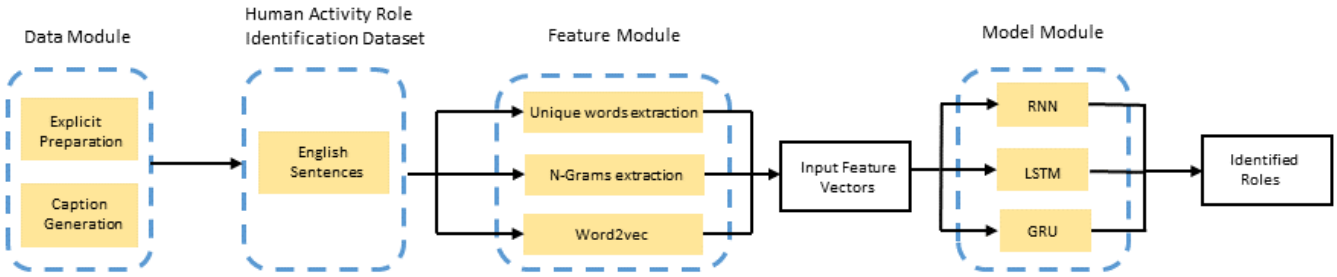

Figure 2: A proposed framework for understanding and comparison. There are two different options to generate data in the data module, three different options to generate input feature vectors in Feature Module, and three different Neural Networks to train our generated feature vectors in Model Module.

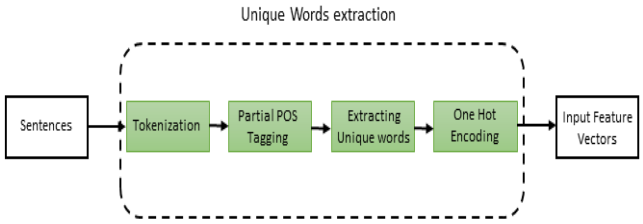

Figure 3: The flowchart of Unique words extraction method

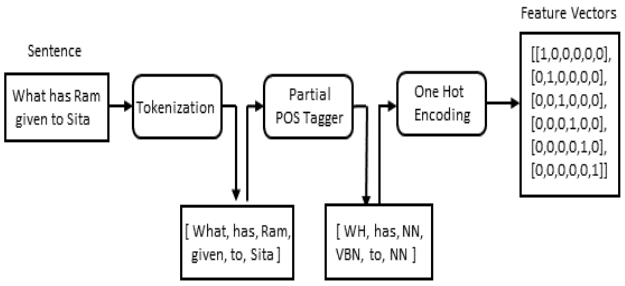

Figure 4: Use Case diagram to generate input feature vectors by extracting unique words

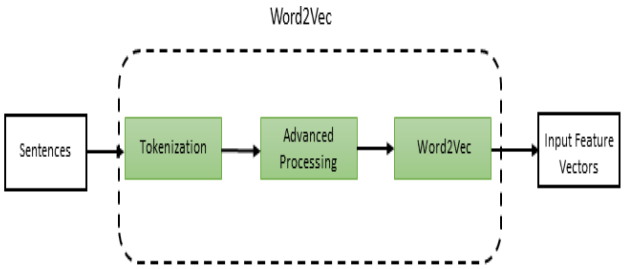

Figure 5: The flowchart of N-Grams extraction method

softmax loss and entropy loss defined in Equation 1.

$$L_{CE} = -\sum_{t=1}^{n} t_i * log(p_i) \tag{1}$$

where t_i is the truth label (target column).
p_i is the softmax probability of the ith class.
n is the total number of classes.

The feature vectors generated by extracting N-Grams are reshaped into 2-dimensional arrays representing a number of sentences and the number of n-grams. Thus, N-Grams are explicit features created and fed to the three Recurrent Neural Networks. Using the same equation 1 from above, we calculate the cross entropy loss in this case as well.

The feature vector generated from Word2Vec is a 2-dimensional array representing the similarity score of each word with all the other words present in the corpus. The mean value of the similarity score for each word is calculated. The feature vector is reshaped and is given as input to RNN, LSTM, and GRU. The previously mentioned categorical cross-entropy loss in Equation 1 is calculated once more for each of the three Recurrent Neural Networks. The best selected Neural Network Model on the basis of comparison can be further used in future work.

5 EXPERIMENTAL RESULTS

The performance of the proposed model has been demonstrated with the newly created dataset "Human Activity Role Identification Dataset". This dataset is created specifically to perform role identification from textual data describing mutual activities. This dataset is used to evaluate each method (discussed earlier) for producing the feature vectors. Various state-of-the-art Recurrent Neural Networks like RNN, LSTM, and GRU were employed to train the dataset, and their performance is compared using various evaluation metrics. A detailed description of the dataset as well as the evaluation metrics is discussed below:

5.1 Human Activity Role Identification Dataset

The human Activity Role Identification dataset consists of English sentences describing mutual activity created exclusively for mutual role identification. The dataset consists of English sentences

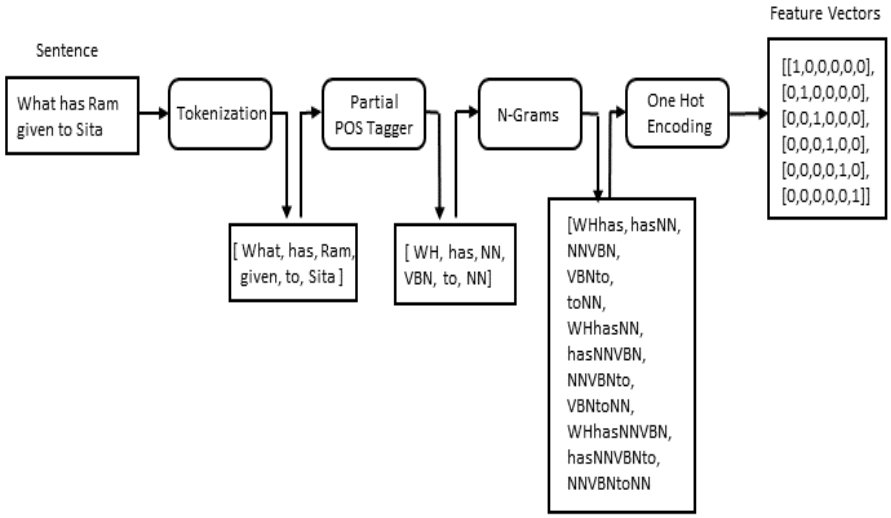

Figure 6: Use Case diagram to generate input feature vectors by extracting N Grams

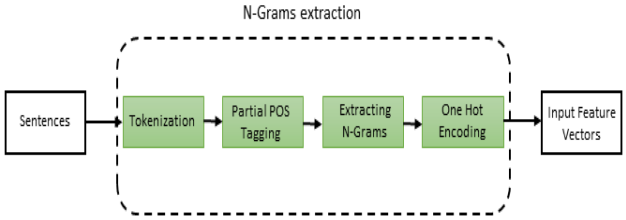

Figure 7: The flowchart of Word2Vec method

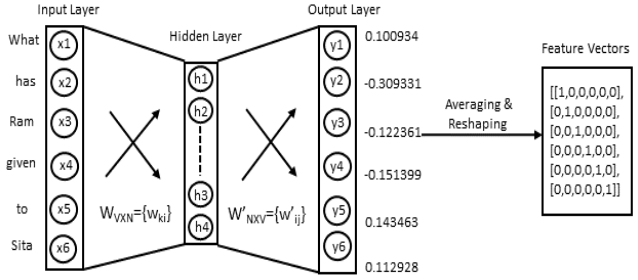

Figure 8: Use Case diagram to generate input feature vectors by using Common Bag of words method of Word2Vec

describing the activity like giving, hitting shouting, killing, playing, standing, and sitting are used to generate the sentences of the dataset. The description of the target role features is given in Table 1. The dataset is small in size and consists of 200 sentences, along with 13 target role features. These target features represent the roles that will be extracted from the sentence. The dataset is developed specifically for role identification by the explicit preparation method of the data module in the proposed architecture.

5.2 Evaluation and Comparision

The experiments conducted for Human Activity Role Identification Dataset are processed by taking the same number of samples as present in the dataset. The dataset consists of 200 sentence samples, where 160 were used for training and the remaining for testing. As discussed earlier our proposed model can be trained by using any of the three Recurrent Neural Networks. Thus, experiments were performed on RNN, LSTM, and GRU, and their results were compared. The three ways to generate feature vectors were also tested on these three Recurrent Neural Networks and combined comparative analysis is shown in Table 2. Furthermore, multiclass weighted average Precision, Recall, and F1 scores were used as the evaluation metrics. In addition to, a loss vs epochs graph for each experimental result was discussed to understand the performance of our model. A detailed description of various feature generation methods trained on various Recurrent Neural Networks using the evaluation metrics is discussed below:

5.2.1 Precision: means that the model will produce more relevant results as compared to irrelevant ones. It is defined in Equation 2 as:

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive} \tag{2}$$

Since our dataset is imbalanced with multiple classes Weighted Average Precision is calculated in Equation 3 as:

$$WeightedAveragePrecision = \sum_{i=1}^{n} x_i * p_i / n \tag{3}$$

where n is the total number of classes.
p_i is the Precision of each class.
x_i is the weight assigned to each class.

In Table 2, LSTM offers the highest precision of 0.36 when feature vectors are generated by extracting unique words. The precision

Table 1: Description of Target columns in the Dataset

Target Columns	Description
WHAT_HAS_VERB3	If sentence has what +Third form of the verb.
WH0_HAS_VERB3	If sentence has who +Third form of the verb.
WHOM_HAS_VERB3	If sentence has whom +Third form of the verb.
WHAT_VERB	If sentence has what +First form of the verb.
WHO_VERB	If sentence has what + First form of the verb.
WHOM_VERB	If sentence has what +First form of the verb.
HOW_VERB	If sentence has what +First form of the verb.
Time	If Time is described in the sentence.
REASON_OF_VERB	If sentence gives a reason for the verb.
Event	If Event is described in the sentence.
PLACE_OF_EVENT	If Place of the event is described in the sentence.
FROM_WHERE_HAS_BEEN_VERB3	If Sentence has from where has been + Third form of the verb.
WHERE_HAS_BEEN_VERB3	If Sentence has where has been + Third form of the verb.

values for all three Neural Networks when features are extracted using Word2Vec are very poor. Thus, the Word2Vec method of feature extraction is not providing satisfactory results in terms of precision.

5.2.2 Recall: means that the method will produce most of the relevant results. It is defined in Equation 4 as:

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative} \tag{4}$$

Here also Weighted Average Recall is calculated in Equation 5 as:

$$WeightedAverageRecall = \sum_{i=1}^{n} x_i * r_i/n \tag{5}$$

where n is the total number of classes.
r_i is the Recall of each class.
x_i is the weight assigned to each class.

Feature vectors generated by extracting N-Grams and extracting unique words are having almost the same recall value. However, the performance of RNN is better with N-Grams as compared to unique words. Furthermore, the performance of Word2Vec is extremely poor in terms of recall.

5.2.3 F1 Score: is the harmonic mean of precision and recall.It is defined in Equation 6 as:

$$F1Score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{6}$$

The Weighted Average F1 Score is calculated in Equation 7 as:

$$WeightedAverageF1Score = \sum_{i=1}^{n} x_i * f_i/n \tag{7}$$

where n is the total number of classes.
f_i is the F1 Score of each class.
x_i is the weight assigned to each class.

The higher the value of precision and recall, the higher the F1 score. Refer to Table 2, F1 Score is highest in that case where feature vectors are generated by unique words. The LSTM model is performing better with an F1 Score of 0.44. However, the performance of Word2Vec is poor in terms of F1 Score also.

5.2.4 Loss: In order to have a graphical view of the performance of the proposed model, the categorical cross-entropy loss as discussed in Equation 1 is calculated against each epoch for all the explored methods. The graphs of Training loss and Testing loss for all the feature generation methods on the three Recurrent Neural Networks are shown in Figure 9.

From the graphs, we observed that the performance of LSTM in all three feature generation methods is better compared to RNN and GRU. The LSTM model is the least overfitted as compared to RNN and GRU. The performance of the Word2Vec feature generation method is poor as it is the most overfitted model in all three Recurrent Neural Networks.

6 CONCLUSION AND FUTURE SCOPE

The primary objective of the study is to identify "roles" while presuming that the activities have been previously acknowledged and described in the text. Primarily, the model stores sentences from which words are extracted, and input feature vectors are generated. These feature vectors are passed through the Recurrent Neural Networks to extract roles. Experiments of role identification on the newly created Human Activity Role Identification Dataset were performed using various ways to generate feature vectors on various types of Recurrent Neural Networks. The performance of LSTM is better than GRU and RNN when feature vectors are generated by extracting unique words because the unique word extraction method extracts word embedding that represents large sequences. Similarly, the performance of RNN is good when feature vectors are generated by extracting N-grams because the N-grams extraction method extracts word embedding that represents small sequences. Word2Vec has a strong estimation of words based on occurrences thus, their performance is poor in terms of capturing the relationship between sequences of words. Finally, the conclusion is that the

Table 2: Table depicts a comparison between all the three ways of input feature vector generation on the three Recurrent Neural Networks.

Model	N-Grams			Unique Words			Word2Vec		
	Precision	Recall	F1-score	Precision	Recall	F1-score	Precision	Recall	F1-score
RNN	0.32	0.50	0.39	0.31	0.50	0.38	0.26	0.17	0.10
LSTM	0.34	0.58	0.42	0.36	0.58	0.44	0.10	0.28	0.14
GRU	0.33	0.54	0.40	0.30	0.40	0.40	0.10	0.28	0.10

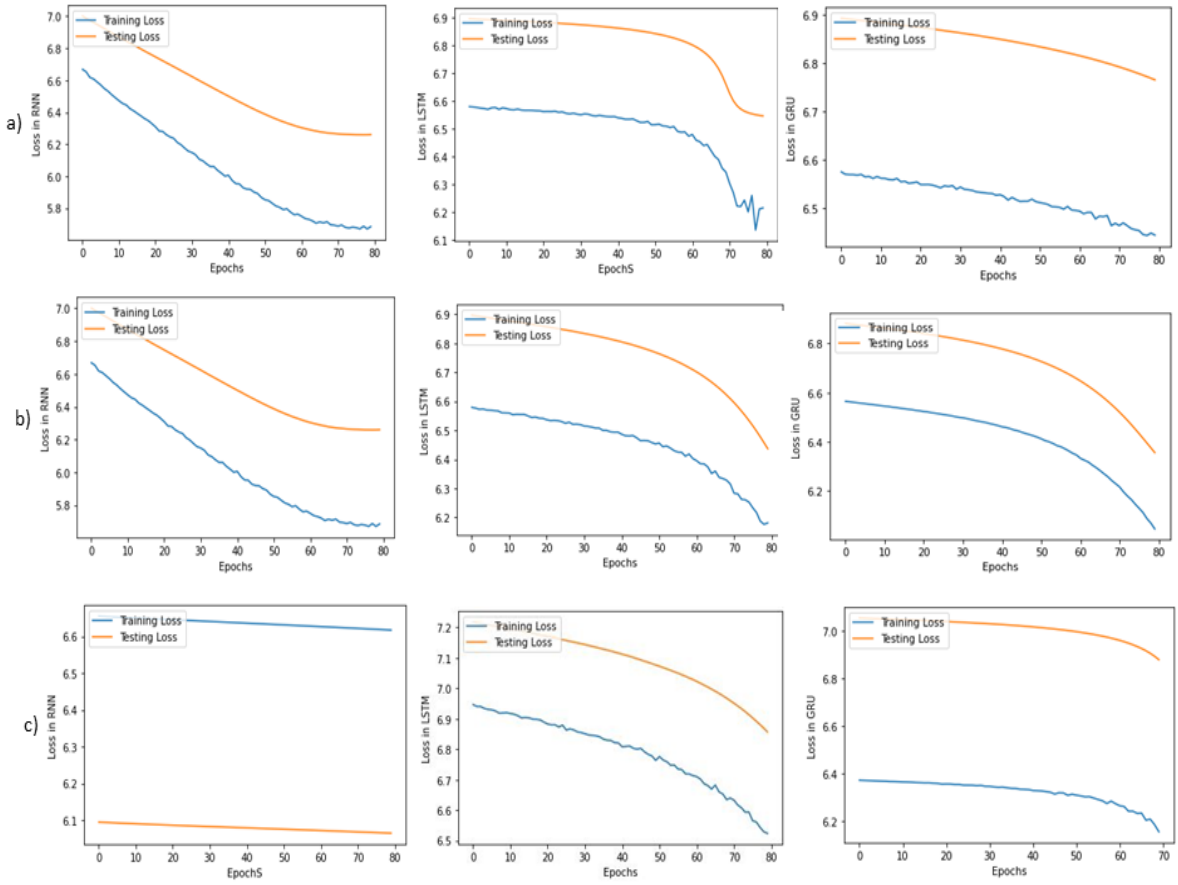

Figure 9: Training and Testing loss vs epochs for RNN, LSTM, and GRU on extracting feature vectors by a) extracting N-Grams,b) extracting unique words, c) Word2Vec.

combination of 'unique word extraction' and 'LSTM' gives promising results. There is no need to consider all three feature extraction methods and three deep learning models in real-life deployment. The experiments reveal that unique word extraction' and LSTM' should be used only.

The proposed method is not a typical NLP+NN modelling experiment, but a way of performing role Identification for level-2 activity by using NLP and Deep Learning. Role identification was the utmost objective of the presented study. The idea of the role binding with entities and objects to identify level-2 activity is kept for future work. The study attempted to find the best feature generation method and best neural network to train the dataset by performing a comparative analysis between various methods. This setup may be utilized for role binding to recognize the complex activity. Furthermore, the caption generation method for sentence creation is still open and needs to validate through experiments.

REFERENCES

[1] Anam Arshad, Vivek Tiwari, Mayank Lovanshi, and Rahul Shrivastava. 2023. Role Identification from Human Activity Videos using Recurrent Neural Networks. In *proceedings of the 8th IEEE International Women in Engineering (WIE) Conference on Electrical and Computer Engineering (WIECON-ECE).*

[2] Djamila Romaissa Beddiar, Brahim Nini, Mohammad Sabokrou, and Abdenour Hadid. 2020. Vision-based human activity recognition: a survey. *Multimedia Tools and Applications* 79, 41 (2020), 30509–30555.

[3] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. 2017. Realtime multi-person 2d pose estimation using part affinity fields. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 7291–7299.

[4] Alebachew Chiche and Betselot Yitagesu. 2022. Part of speech tagging: a systematic review of deep learning and machine learning approaches. *Journal of Big Data* 9, 1 (2022), 1–25.

[5] Jason PC Chiu and Eric Nichols. 2016. Named entity recognition with bidirectional LSTM-CNNs. *Transactions of the association for computational linguistics* 4 (2016), 357–370.

[6] Wongun Choi and Silvio Savarese. 2013. Understanding collective activitiesof people from videos. *IEEE transactions on pattern analysis and machine intelligence* 36, 6 (2013), 1242–1257.

[7] Meenakshi Choudhary, Vivek Tiwari, and U Venkanna. 2020. Enhancing human iris recognition performance in unconstrained environment using ensemble of convolutional and residual deep neural network models. *Soft Computing* 24, 15 (2020), 11477–11491.

[8] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555* (2014).

[9] Rahul Dey and Fathi M Salem. 2017. Gate-variants of gated recurrent unit (GRU) neural networks. In *2017 IEEE 60th international midwest symposium on circuits and systems (MWSCAS)*. IEEE, 1597–1600.

[10] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. 2015. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2625–2634.

[11] John T Hancock and Taghi M Khoshgoftaar. 2020. Survey on categorical data for neural networks. *Journal of Big Data* 7, 1 (2020), 1–41.

[12] Maria M Hedblom, Oliver Kutz, Rafael Peñaloza, and Giancarlo Guizzardi. 2019. Image schema combinations and complex events. *KI-Künstliche Intelligenz* 33, 3 (2019), 279–291.

[13] Jiaxin Huang, Chunyuan Li, Krishan Subudhi, Damien Jose, Shobana Balakrishnan, Weizhu Chen, Baolin Peng, Jianfeng Gao, and Jiawei Han. 2021. Few-Shot Named Entity Recognition: An Empirical Baseline Study. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. 10408–10423.

[14] Yanli Ji, Guo Ye, and Hong Cheng. 2014. Interactive body part contrast mining for human interaction recognition. In *2014 IEEE international conference on multimedia and expo workshops (ICMEW)*. IEEE, 1–6.

[15] Yaozong Jia and Xiaobin Xu. 2018. Chinese named entity recognition based on cnn-bilstm-crf. In *2018 IEEE 9th international conference on software engineering and service science (ICSESS)*. IEEE, 1–4.

[16] Rafal Jozefowicz, Oriol Vinyals, Mike Schuster, Noam Shazeer, and Yonghui Wu. 2016. Exploring the limits of language modeling. *arXiv preprint arXiv:1602.02410* (2016).

[17] Maosen Li, Siheng Chen, Xu Chen, Ya Zhang, Yanfeng Wang, and Qi Tian. 2019. Actional-structural graph convolutional networks for skeleton-based action recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 3595–3603.

[18] Ivan Lillo, Juan Carlos Niebles, and Alvaro Soto. 2017. Sparse composition of body poses and atomic actions for human activity recognition in RGB-D videos. *Image and Vision Computing* 59 (2017), 63–75.

[19] Mayank Lovanshi and Vivek Tiwari. 2023. Human Pose Estimation: Benchmarking Deep Learning-based Methods. In *proceedings of the IEEE Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation*.

[20] Fang Luo, Han Xiao, and Weili Chang. 2011. Product named entity recognition using conditional random fields. In *2011 Fourth international conference on business intelligence and financial engineering*. IEEE, 86–89.

[21] Steven L Lytinen. 1992. Conceptual dependency and its descendants. *Computers & Mathematics with Applications* 23, 2-5 (1992), 51–73.

[22] Jamie C Macbeth and Dagmar Gromann. 2019. Towards Modeling Conceptual Dependency Primitives with Image Schema Logic. (2019).

[23] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).

[24] Kriti Pawar, Raj Srujan Jalem, and Vivek Tiwari. 2019. Stock market price prediction using LSTM RNN. In *Emerging Trends in Expert Applications and Security: Proceedings of ICETEAS 2018*. Springer, 493–503.

[25] Ronald Poppe. 2010. A survey on vision-based human action recognition. *Image and vision computing* 28, 6 (2010), 976–990.

[26] Michalis Raptis and Leonid Sigal. 2013. Poselet key-framing: A model for human activity recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2650–2657.

[27] Michael S Ryoo. 2011. Human activity prediction: Early recognition of ongoing activities from streaming videos. In *2011 International Conference on Computer Vision*. IEEE, 1036–1043.

[28] Mohammad Sadegh Aliakbarian, Fatemeh Sadat Saleh, Mathieu Salzmann, Basura Fernando, Lars Petersson, and Lars Andersson. 2017. Encouraging lstms to anticipate actions very early. In *Proceedings of the IEEE International Conference on Computer Vision*. 280–289.

[29] Roger C Schank. 1972. Conceptual dependency: A theory of natural language understanding. *Cognitive psychology* 3, 4 (1972), 552–631.

[30] Alex Sherstinsky. 2020. Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network. *Physica D: Nonlinear Phenomena* 404 (2020), 132306.

[31] Rahul Shrivastava, Vivek Tiwari, Swati Jain, Basant Tiwari, Alok Kumar Singh Kushwaha, and Vibhav Prakash Singh. 2022. A role-entity based human activity recognition using inter-body features and temporal sequence memory. *IET Image Processing* (2022).

[32] Kamilya Smagulova and Alex Pappachen James. 2019. A survey on LSTM memristive neural network architectures and applications. *The European Physical Journal Special Topics* 228, 10 (2019), 2313–2324.

[33] Daniel Soutner and Luděk Müller. 2013. Application of LSTM neural networks in language modelling. In *International Conference on Text, Speech and Dialogue*. Springer, 105–112.

[34] Vivek Tiwari, Aditi Agrahari, and Sriyuta Srivastava. 2021. Performance analysis of hand-crafted features and cnn toward real-time crop disease identification. In *Information and Communication Technology for Intelligent Systems: Proceedings of ICTIS 2020, Volume 1*. Springer, 497–505.

[35] Arash Vahdat, Bo Gao, Mani Ranjbar, and Greg Mori. 2011. A discriminative key pose sequence model for recognizing human interactions. In *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*. IEEE, 1729–1736.

[36] Daniel Weinland, Remi Ronfard, and Edmond Boyer. 2006. Free viewpoint action recognition using motion history volumes. *Computer vision and image understanding* 104, 2-3 (2006), 249–257.

[37] Yong Yu, Xiaosheng Si, Changhua Hu, and Jianxun Zhang. 2019. A review of recurrent neural networks: LSTM cells and network architectures. *Neural computation* 31, 7 (2019), 1235–1270.

[38] Kiwon Yun, Jean Honorio, Debaleena Chattopadhyay, Tamara L Berg, and Dimitris Samaras. 2012. Two-person interaction detection using body-pose features and multiple instance learning. In *2012 IEEE computer society conference on computer vision and pattern recognition workshops*. IEEE, 28–35.

[39] Yimeng Zhang, Xiaoming Liu, Ming-Ching Chang, Weina Ge, and Tsuhan Chen. 2012. Spatio-temporal phrases for activity recognition. In *European Conference on Computer Vision*. Springer, 707–721.

[40] Qiang Zhou and Gang Wang. 2012. Atomic action features: A new feature for action recognition. In *European Conference on Computer Vision*. Springer, 291–300.