Sentence Based Discourse Classification for Hindi Story Text-to-Speech (TTS) System

Kumud Tripathi, Parakrant Sarkar and **K. Sreenivasa Rao**
Department of Computer Science and Engineering
Indian Institute Technology Kharagpur, India
{kumudtripathi.cs, parakarantsarkar}@gmail.com, ksrao@iitkgp.ac.in

Abstract

In this work, we have proposed an automatic discourse prediction model. It predicts the discourse information for a sentence. In this study, three discourse modes considered are descriptive, narrative and dialogue. The proposed model is developed using story corpus. The story corpus comprises of audio and its corresponding text transcription of short children stories. The development of this model entails two phases: feature extraction and classification of the discourse. The feature extraction is carried out using 'Word2Vec' model. The classification of discourse at sentence-level is explored by using Support Vector Machines (SVM), Convolutional Neural Network (CNN) and a combination of CNN-SVM. The main focus of this study is on the usage of CNN for developing the model because it has not been explored much for the problems related to text classification. Experiments are carried out to find the best model parameters (such as the number of the filter, filter-height, cross-validation number, dropout rate, and batch-size) for the CNN. The proposed model achieves its best accuracy 72.6% when support vector machine (SVM) is used for classification and features are extracted from CNN (which is trained using the word2vec feature). This model can leverage the utilization of the discourse as a suprasegmental feature from the perspective of speech.

1 Introduction

In recent years, there has been a lot of works on analysis of storytelling style speech. In (Montao et al., 2013), TTS system was developed for synthesizing story style speech in Spanish. The main focus was on the analysis of prosodic patterns (such as pitch, intensity, and tempo) based on discourse modes. The three discourse modes considered for the study are narrative, descriptive and dialogue. Further, the authors introduced narrative situations such as neutral narrative, post-character, suspense and affective situations within the narrative mode. The discourse information was manually assigned to each sentence of the story by text experts. Based on the discourse modes, the sentence was grouped, and prosodic rules are derived. These prosodic rules implemented using Harmonic plus Noise Model (HNM) for synthesizing storytelling style speech. In (Delmonte and Tripodi, 2015), an analysis based on discourse mode was carried out to make TTS system more expressive for English. The analysis of text was carried out at phonetic, phonological, syntactic and semantic level. The prosodic manager is proposed which takes discourse structures as input and uses the information to modify the parameters of the TTS. The authors further carried out studies by proposing various discourse relations (Delmonte, 2008; Delmonte et al., 2007).

In storytelling style speech (Theune et al., 2006), a storyteller uses his/her skill by putting variation in the speech for the better understanding of the listeners (especially children). The variation in speech is produced by mimicking various character's voices present in the story, making various sound effects, and using prosody to convey emotions. It creates a pleasant listening experience for the listeners. In Indian Languages, development of TTS systems for Hindi, Bengali and Telugu are carried out in (Verma et al., 2015; Sarkar et al., 2014). In most of the earlier works, discourse information of sentence is obtained by manual annotation. This information is annotated by the text experts of the particular language. In this work, we are developing a method to auto-

D S Sharma, R Sangal and A K Singh. Proc. of the 13th Intl. Conference on Natural Language Processing, pages 46–54,
Varanasi, India. December 2016.

matically classify the sentence based on discourse modes. This information will be processed further to improve the prediction performance of the prosody models that are developed for Story TTS systems. In Hindi Story TTS system, discourse information plays a vital role in capturing the story semantic information. The prosody modeling for duration, intonation, intensity and pause using discourse information are carried out in (Sarkar and Rao, 2015). In view of this, we have explored various machine learning techniques to automatically predict the discourse of the sentence in a story.

In NLP, sentence classification has been carried out using machine learning techniques such as SVM, KNN and K-fold cross-validation. There have various types of deep learning architectures like recurrent neural network (RNN), deep neural network (DNN) etc. In this work we have used a convolutional neural network (CNN) which is motivated from (Yoon, 2006). For the first time (Yoon, 2006) proposed a framework for using convolutional neural network (CNN) for text classification (i.e. emotion and question classification) at sentence level. The fundamental property of CNN (shared weight and local connectivity) make it different and suitable for sentence classification. Rather than learning single global weight matrix between layers, they expect to find an arrangement of locally connected neurons. Similarly, In the case of sentence classification, we need to find out the relationship between words in a sentence. Experiments were carried out using CNN and DNN, among these CNN gave better performance.

In this work, our aim is to create a model which can recognize the discourse of sentences. We have implemented Convolutional neural network (CNN) for automated sentence (from Hindi story corpus) level discourse classification that has not been addressed yet. We have considered three discourse modes- a Descriptive mode which enables the audience to develop a mental picture of what is being discussed, Narrative mode, it relies on stories and Dialogue mode, which includes the exchange of conversation in a group or between two persons directed towards a particular subject. The performance of these models is evaluated by using confusion matrix.

The work flow of this paper is as follows; the story-speech corpus is discussed in section II. Section III describes the proposed architecture for discourse classification. This section also explains about the vector representation of words, convolutional neural network (CNN), multiclass SVM and combined CNN-SVM model. The Section IV, discuss the experiments and results of the systems. The conclusion and future work of this paper have been included in section V.

2 Story Speech Corpus

The story speech corpus in Hindi consists of both story text and its corresponding story wave files. The children story texts are collected from story books like *Panchatantra*[1] and *Akbar Birbal*[2]. The speech corpus comprises of 105 stories with a total of 1960 sentences and 25340 words. These stories were recorded by a professional female storyteller in a noise free studio environment. For maintaining the high recording quality of the stories, continuous feedback is given to the narrator for improving the quality of the recordings. The speech signal was sampled at 16 kHz and represented as 16-bit numbers. The total duration of the story corpus is about 3 hours.

In this study, we considered only three different kinds of discourse modes (i.e. narrative, descriptive and dialogue). In literature, there are discourse modes such as narrative, descriptive, argumentative, explanatory and dialogue (Adell et al., 2005). It is been observed in the children stories that the different parts of story are narrated in different styles based on the semantics present at that part of story. In general, most of the children stories in Hindi, begins with introducing the characters present in story, followed by various events related to the story and finally story will conclude with a moral. In the narration of the story, as it progresses one event after another, narrative mode is used to depict the listener/reader about the actions taking place in story. The descriptive mode shows the various activities that the main character is experiencing. Dialogue mode is used for any type of conversation taking place between any two characters. Generally, a greater amount of the text comprises of narrative mode. A storyteller uses his/her skills to add various expressive registers at sentence-level while narrating a story.

For Hindi children stories text classifications are shown in (Harikrishna and Rao, 2015) and (Harikrishna et al., 2015). Similar approached is followed for manually annotating the story-corpus

[1] https://en.wikipedia.org/wiki/Panchatantra
[2] https://en.wikipedia.org/wiki/Akbar_Birbal

Data	C	N	V	L	Test
Hindi Storyteller Speech	3	1960	3512	44	CV

Table 1: Dataset information. *C*: Number of Output classes. *N*: Number of sentences in storyteller speech corpus. $|V|$: Size of vocabulary. *L*: Maximum length of a sentence. *Test*: Size of test data (CV: train/test data partition is done by using 6-fold crossvalidation (CV)).

based on the three discourse modes. At sentence-level, text of the story was entrusted by four native Hindi speakers on text classification. They have been trained separately and work independently in order to avoid any labeling bias. In order to make the task of the annotation more focused, various discourse modes are annotated from the point of view of the text. Each annotator's task is to label the sentence with one of the modes of discourse (i.e. descriptive, dialogue and narrative). In the story corpus, there are 1960 sentences in which narrative, descriptive, and dialogue mode have 1127, 549, and 294 sentences, respectively. The inter-annotator agreement is given by Fleiss Kappa (κ). The κ values above 0.65 or so can be considered to be substantial. The κ value is 0.73 for the annotating the discourse mode to each sentence. Following are the example sentences of given discourse mode:

Descriptive mode

- ek taalaab men do machchha rahate the
- yah kah kar mendhak vahaan se chala gaya
- tabhi dono ne hi samajha ki ab to jaan bachi

Narrative mode

- tab tak birbal bhi darbar men aa pahunche
- baadshah ne vahi prashna unse bhi puchha
- rakhvala sevak ghabra gaya

Dialogue mode

- aisa mat karo isase to ham donon hi maare jaenge
- soch lo na dikha sakhe to saja milegi
- tumne yah chamatkar kaise kiya

3 Proposed Model

In the work, we have used three model for discourse classification. In the first and second model, word2vec is used for feature extraction, and CNN and SVM (Joachims, 1998) respectively are used for classification. The third model is the combination of CNN and SVM (Cao et al., 2015), where CNN is used for feature extraction and SVM is used for classification. All these models are described in details, further.

3.1 Word to Vector

In text processing problems, words act as an important (Turian et al., 2010) feature. The words are considered as a distinct atomic (Collobert et al., 2011) attribute, for example, a word 'car' might be expressed as '*id123*'. This representation of a word is not sufficient enough to highlight the relation that may exist between the words in a story. In order to train the models successfully, there is a need for better representation of words. The vector representation of the word allows capturing the relevant information of the word for a task at hand. The values for the vector of words either could be generated randomly or by using some learning model like word2vec. Traditionally TTS system used uniform distribution for vector representation of words. Uniform distribution provides random values to word vector. Training using these vectors require large training data. For less training data, word vector cannot be learned properly, and they may be overfitted.

Therefore, instead of randomly initialization of word, we used word2vec model for vector representation of words. In general, word2vec model uses two types of algorithms (i)Continuous Bag-of-word (CBOW) model and (ii) Skip-Gram model (Mikolov and Dean, 2013). In this work, CBOW model is used for obtaining the vector representation of the word.

The architecture of the CBOW model can be seen as a fully connected neural network with a single hidden layer. The bag-of-word represents the relationship between a word and its surrounding word. In this work, two successor and two predecessor words are taken as input to recognize the current word. We have evaluated the accuracy of the CBOW by varying the dimension of the word vector such as 10, 15, 20, 25, 30, 40, 50 and 100. The 20-dimensional feature vector gave the optimal performance for the training data. Table 2

shows that similar words (Mikolov et al., 2013) in the vocabulary is extracted by measuring the cosine distance between the word vectors.

Query Word	Simillar Word
birbal	manoranjak
	akabar
	baadshaah
	taariph
	samjhaya
pyaara	sajidhaji
	aanandit
	chaahti
	bhaabhi
	ijjat

Table 2: Top five similar words (calculated using cosine distance measure technique) from the vocabulary size of 3512 words.

3.2 Convolution Neural Network (CNN)

In this work, CNN has been explored for sentence based discourse classification problem. The reason for using CNN is to have a model that can easily manage the word sequence in a sentence and finds the relationship between the surrounding words. The key concept in CNN is the convolution operation. The convolution is between the input and filter matrices to find the relation in a sequence. In this work, input matrix of the CNN corresponds to a sentence. Figure 1 represents the complete process of training and testing the CNN model with an example of training and testing sentence. A sentence in the CNN model is represented by $S_1 \in \Re^{n \times v}$, where S denotes an input matrix to the CNN, n is the number of words in a sentence and v is the word vector dimension which is extracted from word2vec model[3]. Zero-padding is done for the varying sentence length. Let $F \in \Re^{h \times v}$ corresponds to a filter for performing mathematical convolution operation which convolve with each possible pair of h words in a sentence $[S_{1:h}, S_{2:h+1}, ..., S_{n-h+1:n}]$ to generate a feature map

$$m = [m_1, m_2, ..., m_{n-h+1}]$$

The next task is to perform max-pooling operation on the feature maps generated using a convolution filter and calculates the maximum value $M = max\{m\}$. In this way CNN extract dominant features for each feature map. CNN learns

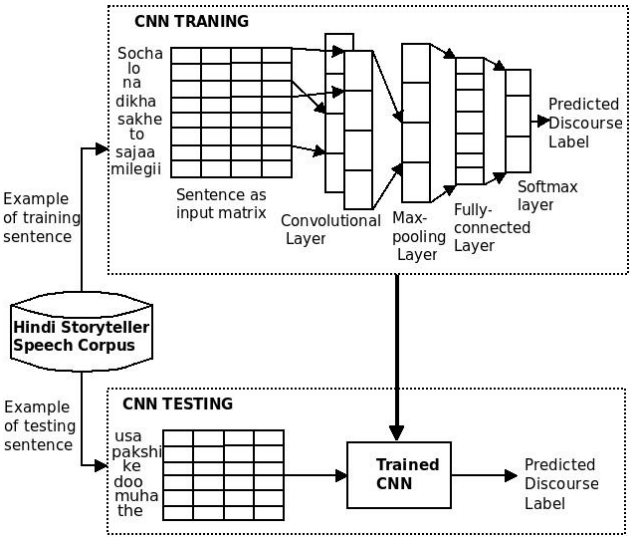

Figure 1: CNN Achitecture for discourse classification.

to convert a given sentence into a discourse label by calculating discriminant features (Dosovitskiy et al., 2014).

Model Hyperparameters Finding a set of hyperparameters (Hoos et al., 2014) is a neccessary task for optimising convolutional networks. There are various hyperparameters to tune in CNN such as the number and the height of filters, learning rate, dropout rate (Krizhevsky et al., 2012), L2 constraint, etc. Value of these hyperparameters depends (Zhang and Wallace, 2015) on the task at hand.

We observed that model performance is varying by using various hyperparameters. In the case of discourse classification, the proposed model achieves the best performance at the following values of the hyperparameters. It includes three filters for convolution operation, and their corresponding sizes are (3×100, 4×100 and 5×100) where 3,4,5 is the height of the filter with 100 feature map. AdaGrad (Duchi et al., 2011) technique is used for training the model where the parameter for learning rate is 0.05. The value for L2-constraint is 3 which is used for regularizing the model parameters. The penultimate layer of the model is regularized using dropout rate of 0.7. We observe that model performance is varying as per the various dropout rate. In the case of discourse classification model performance is reduced if dropout rate is more or less than 0.7.

[3]https://code.google.com/p/word2vec/

Model Regularization CNN is more prone to overfitting because of a large number of hyper-parameter tuning while training. Overfitting is a situation when the model is overtrained for training data, and it will be unable to predict the new data correctly. This problem is resolved by using regularization. For Regularization, dropout technique is implemented at the second last layer of CNN. Dropout is a method to deactivate (Miao and Metze, 2013) randomly selected hidden nodes. The dropped out nodes does not contribute to the training of the model. With the help of the dropout, technique model will learn more generalize feature. Also, the performance of the model increases because the active nodes are now insensitive to dropped-out nodes. In this work, we have seen dropout rate of 0.7 gives good result compare to values greater and less than this.

3.3 Multiclass Support Vector Machine

In this section, we discuss SVM (Rennie and Rifkin, 2001) for the multiclass problem by utilizing one-vs-rest method. The process involves L binary SVM for L classes and data from that class is taken as positive and remaining data is taken as negative. In the Hindi story corpus, at sentence level features are extracted to train the SVM. The words $w \in \Re^v$ in the sentence is represented by v dimensional vector extracted using word2vec model. The input sentence is represented as:

$$S_2 = [w_1 + w_2 + ... + w_n]$$

Here, the plus $+$ symbol denotes the concatenation of the word vector. The Figure 2 shows the framework of the procedure followed for training and testing of SVM model.

3.4 CNN-SVM Model

In this section, we explored the combination of CNN-SVM model for developing the automatic discourse predictor. The CNN generates discriminant features, and SVM gives better generalization on the new data. During learning SVM tries to find out global hyperplane and CNN tries to minimize cross-entropy value. SVM provides better classification results than the softmax function at the fully-connected layer of the CNN. Here, the architecture of the CNN model is same as we have used before in this work 3.2. In this model, CNN is used for extraction of the feature for the sentence and then these features are used for training the SVM. The softmax function used to generate the probabilistic value for the input data. This value is treated as a feature produced by CNN.

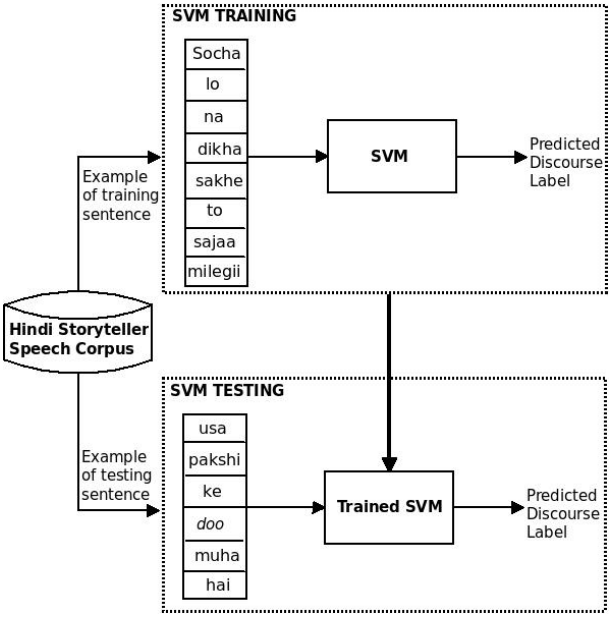

Figure 2: SVM Achitecture for discourse classification.

The Figure 3 shows that CNN takes a sentence as an input matrix S_1 which is generated using word2vec model. For feature extraction task CNN uses its original model in which softmax function is used at the output layer. After training of CNN, the features are obtained. These features are used for training the SVM model. The testing is carried out by extracting the features from CNN and using SVM classification.

4 Experimental Results

In this section, we discuss the experiments carried out on Hindi story corpus to analyze the accuracy of the discourse prediction model. The evaluation is performed using a various parameter of CNN (number of the filter (*N*), filter-height (*H*), cross-validation (*CV*) number, dropout rate (*D*), and batch-size (*B*)). Effect of each value of the parameters significantly alters the performance of the model. During experiment value of CV varies from 8 to 10, the value of N in between 1 to 3, the value of H ranges in between 3 to 5, the value of B varies from 50, 100 and 150 and D lies in the range of 0.2 to 0.8. After several experiments, we got optimal results at the number of filter 3, filter-height [3,4,5], Cross-validation (CV) 8, batch-size 50 and dropout rate 0.7.

At the time of training and testing, input to the

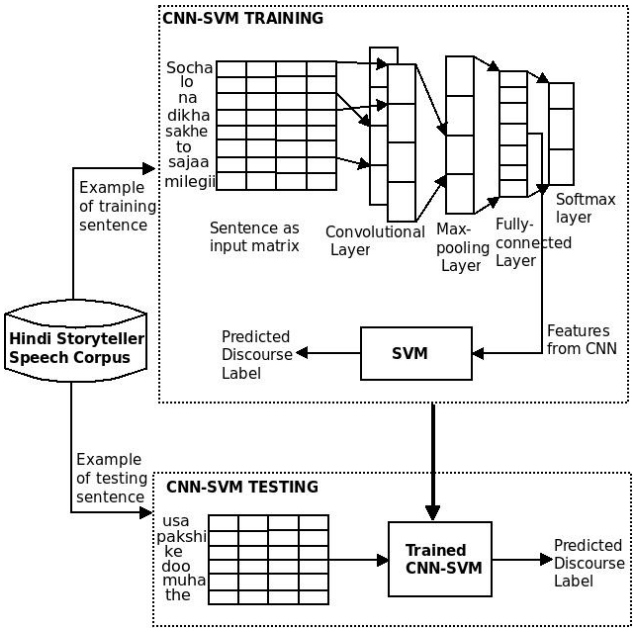

Figure 3: CNN-SVM Achitecture for discourse classification.

model is a sentence. In training, each story is divided into a set of sentences. Each of the sentences are labeled with one of the discourse mode (descriptive (*DS*), dialogue (*DL*), narrative (*NR*)). 1613 sentences are used in training and remaining, 347 sentences are used for testing the performance of the model.

The performance of the proposed methods is evaluated using confusion matrix, ROC curve, and F-Score. A graphical plot of the performance is shown by ROC curve. This curve considers only true positive rate and false positive rate of the testing data. F-Score tells about the sensitivity, specificity, precision, recall, f-measure, and g-mean. Here sensitivity and recall show the true positive rate, specificity shows the true negative rate, precision gives the positive predicted value, f-measure (Espíndola and Ebecken, 2005) is the harmonic mean of precision and recall and g-mean is the geometric mean of precision and recall (Powers, 2014).

Table 3 represents that each discourse is classified (using SVM which is trained on features extracted from CNN) correctly by almost 72.6%. Narrative mode classification is 76.3% because of more training data for this mode, and dialogue mode classification is 65.5%, and descriptive mode classification is 65% because for this class we have fewer data to train our model.

Figure 4 represents the receiver operating characteristic (ROC) for CNN-SVM model where class 1, class 2 and class 3 accounts for the descriptive, narrative, and dialogue mode respectively. Class 2 (narrative mode) has larger true positive rate than other two classes.

	DS (in %)	NR (in %)	DL (in %)
DS	**70 .1**	22.4	7.5
NR	19.4	**78.3**	2.3
DL	10.6	20	**69.4**

Table 3: Discourse classification results for the Hindi story corpus using CNN-SVM.

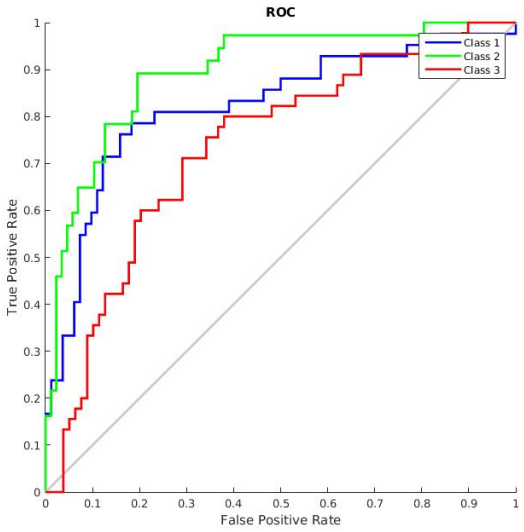

Figure 4: ROC curve for CNN-SVM model.

Table 4 shows that each discourse is classified (using CNN with softmax function which is trained on features extracted from word2vec model) correctly by almost 62.66%. Dialogue mode classification is 39.58% which is lesser than descriptive and narrative mode classification because for dialogue mode we have fewer data to train our model. CNN learns better feature if a particular mode has sufficiently large amount of training data.

Figure 5 represents the ROC curve for CNN model. Class 1 (Descriptive mode) has larger true positive rate than other two classes.

Table 5 shows that each discourse is classified (using SVM which is trained on features extracted from word2vec model) correctly by almost 54.3%. Dialogue mode classification is 18.75% which is

	DS (in %)	NR (in %)	DL (in %)
DS	**77.08**	8.33	14.58
NR	10.86	**71.73**	17.39
DL	25	35.41	**39.58**

Table 4: Discourse classification results for the Hindi story corpus using CNN.

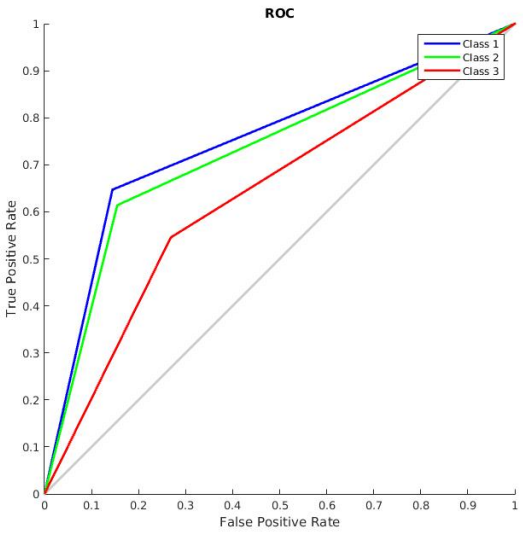

Figure 5: ROC curve for CNN model.

lesser than descriptive and narrative mode classification because for dialogue mode we have fewer data to train our model. SVM give worse results if a class does not have sufficient amount of training data.

	DS (in %)	NR (in %)	DL (in %)
DS	**67.39**	17.39	15.29
NR	10.41	**77.08**	12.5
DL	37.5	43.75	**18.75**

Table 5: Discourse classification results for the Hindi story corpus using SVM.

Figure 6 represents the ROC curve for SVM model. Class 2 (Narrative mode) has larger true positive rate than other two classes. In class 3 (Dialogue mode) false positive rate is greater than true positive rate.

Table 6 represents f-score for all the proposed model. F-score gives almost complete knowledge about the methods. It can be observed that the best performance for discourse classification is 72.6%

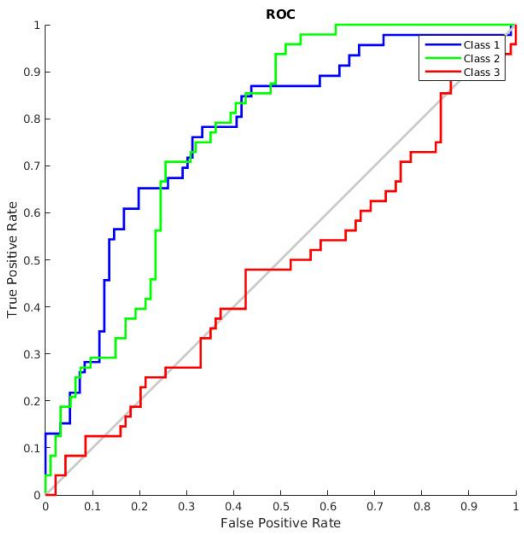

Figure 6: ROC curve for SVM model.

using CNN-SVM model. SVM model gives worst performance against all given model.

F-Score	CNN (in %)	SVM (in %)	CNN-SVM (in %)
Accuracy	62.66	54.3	**72.6**
Sensitivity	77.08	77.08	58.86
Specificity	55.32	42.55	71.97
Precision	46.84	40.66	52.06
Recall	77.08	77.08	58.86
F-measure	58.27	53.24	59.55
G-mean	65.30	57.27	70.88

Table 6: F-Score for discourse classification models.

5 Summary and Conclusion

In this work, we have developed automatic discourse classification model which determine the discourse information of the sentence. We explored SVM and CNN for developing the automatic discourse classification model. In view of this, we collected short children stories to develop story corpus. This corpus is used for developing the automatic discourse predictor. Three modes of discourse are considered, narrative, descriptive and dialogue. The features are used for training the SVM and CNN model are obtained using word2vec method. Our current model achieves its best accuracy (72.6%) when the feature is obtained

using CNN (which is trained on word2vec feature) and classification is done by using SVM.

Future scope of this work is to increase the corpus size to improve the accuracy of the model. Apart from word2vec, we can explore Latent Semantic Analysis (LSA) for obtaining the features. We can also compare the current work by using recurrent neural network (RNN).

Acknowledgments

The authors would like to thank the Department of Information Technology, Government of India, for funding the project, *Development of Text-to-Speech synthesis for Indian Languages Phase II*, Ref. no. 11(7)/2011HCC(TDIL).

References

Jordi Adell, Antonio Bonafonte, and David Escudero. 2005. Analysis of prosodic features towards modelling of emotional and pragmatic attributes of speech. *Procesamiento de Lenguaje Natural*, 35:277–284.

Yuhui Cao, Ruifeng Xu, and Tao Chen. 2015. Combining convolutional neural network and support vector machine for sentiment classification. In *Chinese National Conference on Social Media Processing*, pages 144–155. Springer.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12(Aug):2493–2537.

Rodolfo Delmonte and Rocco Tripodi. 2015. Semantics and discourse processing for expressive tts. In *Workshop on Linking Models of Lexical, Sentential and Discourse-level Semantics (LSDSem)*, page 32.

Rodolfo Delmonte, Gabriel Nicolae, Sanda Harabagiu, Cristina Nicolae, Stefan Trausan-Matu, Cristina Grigore, and Liviu Dragomirescu. 2007. A linguistically-based approach to discourse relations recognition. *Natural Language Processing and Cognitive Science: Proc. of 4th NLPCS (Funchal, Portugal)*, pages 81–91.

Rodolfo Delmonte. 2008. A computational approach to implicit entities and events in text and discourse. *International Journal of Speech Technology*, 11(3-4):195–208.

Alexey Dosovitskiy, Jost Tobias Springenberg, Martin Riedmiller, and Thomas Brox. 2014. Discriminative unsupervised feature learning with convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 766–774.

John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159.

RP Espíndola and NFF Ebecken. 2005. On extending f-measure and g-mean metrics to multi-class problems. *WIT Transactions on Information and Communication Technologies*, 35.

DM Harikrishna and K Sreenivasa Rao. 2015. Children story classification based on structure of the story. In *Advances in Computing, Communications and Informatics (ICACCI), 2015 International Conference on*, pages 1485–1490. IEEE.

DM Harikrishna, Gurunath Reddy, and K Sreenivasa Rao. 2015. Multi-stage children story speech synthesis for hindi. In *Contemporary Computing (IC3), 2015 Eighth International Conference on*, pages 220–224. IEEE.

Holger Hoos, UBC CA, and Kevin Leyton-Brown. 2014. An efficient approach for assessing hyperparameter importance.

Thorsten Joachims. 1998. Text categorization with support vector machines: Learning with many relevant features. In *European conference on machine learning*, pages 137–142. Springer.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105.

Yajie Miao and Florian Metze. 2013. Improving low-resource cd-dnn-hmm using dropout and multilingual dnn training. In *INTERSPEECH*, pages 2237–2241. ISCA.

T Mikolov and J Dean. 2013. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013. Linguistic regularities in continuous space word representations. In *HLT-NAACL*, volume 13, pages 746–751.

Ral Montao, Francesc Alas, and Josep Ferrer. 2013. Prosodic analysis of storytelling discourse modes and narrative situations oriented to text-to-speech synthesis. In *8th ISCA Workshop on Speech Synthesis*, pages 191–196, Barcelona, Spain, August.

David MW Powers. 2014. What the f–measure doesnt measure.

Jason DM Rennie and Ryan Rifkin. 2001. Improving multiclass text classification with the support vector machine.

Parakrant Sarkar and K Sreenivasa Rao. 2015. Analysis and modeling pauses for synthesis of storytelling speech based on discourse modes. In *Contemporary Computing (IC3), 2015 Eighth International Conference on*, pages 225–230. IEEE.

Parakrant Sarkar, Arijul Haque, Arup Kumar Dutta, Gurunath Reddy, DM Harikrishna, Prasenjit Dhara, Rashmi Verma, NP Narendra, Sunil Kr SB, Jainath Yadav, et al. 2014. Designing prosody rule-set for converting neutral tts speech to storytelling style speech for indian languages: Bengali, hindi and telugu. In *Contemporary Computing (IC3), 2014 Seventh International Conference on*, pages 473–477. IEEE.

Mariët Theune, Koen Meijs, Dirk Heylen, and Roeland Ordelman. 2006. Generating expressive speech for storytelling applications. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(4):1137–1144.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. 2010. Word representations: a simple and general method for semi-supervised learning. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pages 384–394. Association for Computational Linguistics.

Rashmi Verma, Parakrant Sarkar, and K Sreenivasa Rao. 2015. Conversion of neutral speech to storytelling style speech. In *Advances in Pattern Recognition (ICAPR), 2015 Eighth International Conference on*, pages 1–6. IEEE.

Ye Zhang and Byron C. Wallace. 2015. A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification. *CoRR*, abs/1510.03820.