

An Integrated Framework for Pronominal Anaphora Resolution in Malayalam

Ajees Arimbassery Pareed*, Sumam Mary Idicula

Department of Computer Science, CUSAT, 682022, India

ARTICLE INFO

Article history:

Received: 13 July, 2019

Accepted: 31 August, 2019

Online: 18 October, 2019

Keywords:

Malayalam

NLP

Anaphora resolution

Deep level tagger

Multilayer perceptron

ABSTRACT

Anaphora resolution is one of the old problems in Natural Language Processing. It is the process of identifying the antecedent of an anaphoric expression in a natural language text. Most of the NLP applications such as text summarization, question answering, information extraction, machine translation etc. require the successful resolution of anaphors. In this paper, we propose a methodology for the resolution of pronominal anaphors present in Malayalam text document. The proposed methodology is a hybrid architecture employing machine learning and rule-based techniques. In our study, we have used a deep level tagger developed using a machine learning based algorithm. The deep level tagger provides detailed information regarding the number and gender of nouns in a text document. The morphological features of the language are effectively utilized for the computational analysis of Malayalam text. Despite using less amount of linguistic features, our system provides better results which can be utilized for higher level NLP tasks such as question answering, text summarization, machine translation, etc.

1 Introduction

Anaphora is a Greek word that originated in 16th-century [1]. It is comprised of 'ana' and 'phora', where 'ana' means back, and 'phora' means carrying. Anaphors can successfully avoid the repetition of a word or phrase in a text document. They frequently occur in both written and spoken forms of the text. Resolving the correct antecedent of an anaphor from the set of possible antecedents is called anaphora resolution. It is one of the complicated problems in Natural Language Processing that has got considerable attention among NLP researchers. Halliday and Hassan defined anaphora as "the cohesion which points back to some previous item" [2]. According to Hirst, "anaphora is a device for making an abbreviated reference to some entity in the expectation that the receiver of the discourse will be able to dis-abbreviate the reference and, thereby, determine the identity of the entity" [3]. Hence, anaphora resolution is the process of disambiguating the antecedent of a referring expression from the set of entities in a discourse. A simple example illustrating the use of an anaphoric expression in English is given below.

Example: **John** went to school. **He** forgot to take his umbrella.

Here the word '**He**' is the anaphoric expression and '**John**' is the antecedent. Expressions referring to the actual entities in the preceding text are called anaphoric expressions. While the targets of anaphoric expressions are called antecedents.

It is a dream of humankind to communicate with the computers, since the beginning of the digital era. Communication with computers is only possible through a proper understanding of natural language text. The proper understanding of natural language text is possible only after anaphora resolution. There is a large amount of text data available online in Malayalam. Malayalam Wikipedia itself contains more than 30,000 articles. This warrants us to develop tools that can be used to explore digital information presented in Malayalam and other native languages. Anaphora resolution is one such tool that helps in semantic analysis of natural language text. It is also necessary for areas like text summarization, question answering, information retrieval, machine translation, etc.

Anaphora can be broadly classified into three categories [4]. They are pronominal anaphora, Definite noun phrase anaphora, and one anaphora. Among the various types of anaphora, pronominal anaphora is the most widely used one, which is realized through anaphoric pronouns. Pronouns are short words that can be substituted for a noun or noun phrase. They always refer to an entity that is already mentioned in the discourse. Pronominal anaphors are always stronger than full definite noun phrases [5]. They are again classified into five namely personal pronouns, possessive pronouns, reflexive pronouns, Demonstrative pronouns and relative pronouns. All the above-mentioned pronouns need not be anaphoric. A simple example illustrating the use of an anaphoric expression in Malayalam is given below.

*Department of Computer Science, CUSAT, 9061859697 & ajeesap87@gmail.com

രജു വീട്ടിൽ പോയി. അവൻ പേനയെടുക്കാൻ മറന്നു.

Figure 1: Sample text illustrating the use of an anaphoric expression in Malayalam

Natural languages across the world are quite diverse in their structure. One language does not follow the syntax and semantics of the other. Moreover, anaphora resolution is a challenging task since it requires expertise in different domains of language processing such as lexical analysis, syntactic analysis, semantic analysis, discourse analysis, etc. When it comes to Malayalam, the major challenges associated with anaphora resolution are

- Lack of preprocessing tools
- Lack of standard datasets
- Agglutinative nature of the language
- Free word ordering behaviour
- Influence of cases in resolving the antecedent
- Use of different fonts to encode the text, etc

In this paper, we will focus only on resolution of pronominal anaphors present in a Malayalam text document using deep level tagging algorithms. The structure of this paper is as follows. Section 2 briefly reviews the related works. Section 3 explains the system architecture. Section 4 includes experiments and results. And section 5 concludes the article along with some directions for future work.

2 Related works

Most of the earlier works in anaphora resolution were reported in English and other European languages. 'STUDENT' is one of the earliest approaches to resolve anaphors through some automated mechanism. It was a high school algebra problem answering system attempted by Bobrow in 1964 [6]. He used a set of heuristics to resolve anaphors and other questions in a text. The second work in English was reported in 1972 by Winograd [7]. He was the first one to develop an algorithm to resolve pronominal anaphors. In his study, he considered previous noun phrases as the candidate antecedents for the pronominal anaphors. Syntactic positions examine all the possibilities from the preceding text. Wilks introduced a multilevel anaphora resolution system in 1973 [8]. Each level can resolve a set of anaphors based on the complexity to find the antecedent. The first level uses the knowledge of individual lexeme meaning to resolve anaphors. Similarly, the last level uses the "focus of attention" rules to find the topic of the sentence and hence the antecedent.

The modern era on anaphora resolution in English starts with the works of Hobbs in 1977 [9]. He developed an algorithm based on the syntactic constraints of the English language. Parse trees are used to search the antecedent of an anaphor in an optimal order such that the noun phrase upon which the search terminates is regarded as the probable antecedent of the pronoun. Carter reports a shallow processing approach in 1986 [10]. His work exploits the knowledge of syntax, semantics and local focusing. But he didn't

give much attention to a large amount of domain knowledge, which he considered as a burden to process accurately. Grosz presented centring theory, a discourse-oriented study for anaphora resolution in 1977 [11]. Lappin and Leass reported another work for anaphora resolution in 1994 [12]. Their algorithm focuses on salience measures derived from the syntactic structure rather than the semantic factors present in the text. McCallum and Wellner presented a CRF based anaphora resolution system for English [13]. Similarly, Charniak and Elsner demonstrated an anaphora resolution system using Expectation Maximisation (EM) algorithm [14].

When it comes to Indian languages, works in anaphora resolution are reported only from few languages such as Hindi, Tamil, Bengali and Malayalam. This situation is contributed by the lack of standard parsers and sophisticated preprocessing tools for Indian languages. The first work in anaphora resolution for Malayalam was reported by Sopha et al. in 2000 as part of her doctoral thesis [15]. Their system (VASISTH) was the first multilingual system for anaphora resolution in Indian languages. Initially, the system was developed for Malayalam and then modified for Hindi. It explores the morphological richness of the language and makes limited use of syntax and other traditional linguistic features. Specific rules are designed for each type of anaphora separately. Athira et al. proposed an algorithm to resolve pronominal anaphors in Malayalam text [16]. They explored a hybrid system for the resolution of pronominal anaphors by incorporating rule-based and statistical approaches. The morphological richness of the language is also utilized in their study. Later, a pronominal anaphora resolution system was proposed by Sobha in 2007 [17]. In her work, she calculated salience weight for each candidate noun phrase from the set of possible candidates. Grammatical features determined the salience factors. The system could achieve comparable performance. High accuracy on a limited data set was the major drawback of her paper. Linguistic issues involved in Co-reference resolution in Malayalam was discussed in detail by Rajendran S in 2018.

3 Proposed method

Our goal is to build an anaphora resolution system for Malayalam which resolves all pronominal anaphors present in a text document. The architecture of the proposed system is shown in figure 2. It shows the general block diagram of pronominal anaphora resolution system for Malayalam. The first module of the architecture is the preprocessing module, where the sentence segmentation and word tokenization operations are carried out. After preprocessing, the preprocessed text is fed to the POS tagging module, where the words from the preprocessed input text are tagged with POS information. The tags used for this study belong to the BIS tagset. The third module of the architecture is the animate noun identification module which identifies the animate nouns from the set of nouns given by the POS tagging module. Animate nouns are the semantic category of nouns referring to persons and animals. They usually correlate with animate pronouns present in a natural language text. Hence identifying animate nouns from the set of nouns in a text document is necessary to resolve the pronominal anaphors present in the document.

After identifying the animate nouns in a POS-tagged text, they

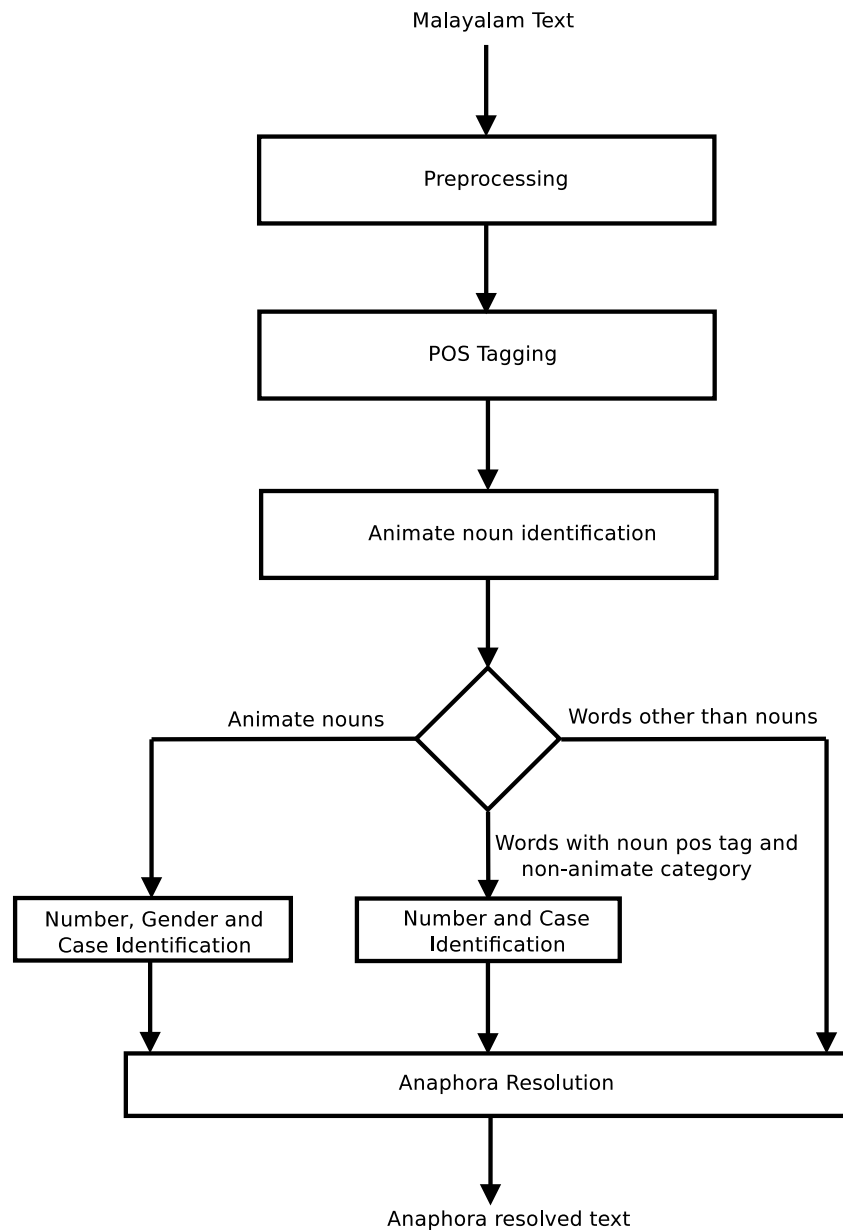


Figure 2: General architecture of the proposed system.

are sent to a number and gender identification neural network. Morphological features of the language are utilized to identify the number and gender information associated with animate nouns. The morphological features are captured with the help of a suffix stripper which can strip the suffixes of different length. Suffixes of different length act as the feature set for number and gender identification module. MLP classifier is used to build the in-depth analyzer of animate nouns. MLP is a class of feedforward artificial neural networks [18]. It is characterized by multiple hidden layers and non-linear activation. The case information associated with the animate nouns is also identified in this module. A rule-based approach is employed for this purpose.

The final module of the architecture is the anaphora resolution module, which is shown in figure 4. The deep level tagged text from the previous modules is the input towards the anaphora reso-

lution module. For each pronoun present in the tagged text, a set of possible candidates are shortlisted. The possible candidates are animate nouns from the previous four sentences of the anaphor, that agree with the number and gender of the pronoun. After shortlisting the possible candidates, a set of salience values are assigned to each candidate. The salience factors are identified through proper analysis of Malayalam text. The set of salience factors we have considered in our study are given in table 2.

The clause information and subject-object information of shortlisted candidates are identified using a rule-based technique. For that, all the sentences containing the shortlisted candidates are divided into clauses. The words with a verb or adjective POS tag act as the clause boundary in a sentence [19]. In the case of adjectives, the nouns following them are also included in the same clause. For each clause identified in the last step, the subject, object and predicate

```
[['നരേന്ദ്രമോദി\N_NNP', 'സർക്കാരിനെ\N_NN', 'പിടിച്ചുലയ്ക്കുന്ന\VM_VNF', 'വിവാദബോംബാണ്\VAUX', 'വിജയ്\N_NNP', 'മധ്യ\N_NNP', 'ഇന്നലെ\N_NN', 'ലണ്ടനിൽ\N_NNP', 'പൊട്ടിച്ചത്\VM_VNF', '.\RDPUNC'], ['ഇന്ത്യയിൽ\N_NNP', 'സാമ്പത്തിക\JJ', 'തട്ടിപ്പുകൾ\N_NN', 'നടത്തിയവർക്ക്\N_NN', 'ബിജെപിയുമായും\N_NN', 'കേന്ദ്ര\JJ', 'സർക്കാരിലെ\N_NN', 'ഉന്നതതലമായും\N_NN', 'അടുത്ത\N_NST', 'ബന്ധമുണ്ടെന്ന\VM_VNF', 'ആരോപണം\N_NN', 'പ്രതിപക്ഷ\N_NN', 'കക്ഷികൾ\N_NN', 'നിരന്തരം\RB', 'ഉയർത്തുന്ന\VM_VNF', 'ഘട്ടത്തിലാണ്\VAUX', 'അവർക്ക്\PR_PRP', 'വെടിക്കോപ്പു\N_NN', 'പകരുന്ന\VM_VNF', 'മധ്യയുടെ\N_NNP', 'വെളിപ്പെടുത്തൽ\N_NN', '.\RDPUNC'], ['ഇന്ത്യ\N_NNP', 'വിടു\N_NN', 'മുൻപ്\N_NN', 'ധനമന്ത്രി\N_NN', 'അരുൺ\N_NNP', 'ജയ്ഘരിയെ\N_NNP', 'കണ്ടിരുന്നു\VM_VF', '.\RDPUNC'], ['ചർച്ച\VM_VNF', 'നടത്തിയിരുന്നു\VM_VF', '.\RDPUNC']]
```



```
[['നരേന്ദ്രമോദി\N_NNP\ANI', 'സർക്കാരിനെ\N_NN\NANI', 'പിടിച്ചുലയ്ക്കുന്ന\VM_VNF', 'വിവാദബോംബാണ്\VAUX', 'വിജയ്\N_NNP\ANI', 'മധ്യ\N_NNP\ANI', 'ഇന്നലെ\N_NN\NANI', 'ലണ്ടനിൽ\N_NNP\NANI', 'പൊട്ടിച്ചത്\VM_VNF', '.\RDPUNC'], ['ഇന്ത്യയിൽ\N_NNP\NANI', 'സാമ്പത്തിക\JJ', 'തട്ടിപ്പുകൾ\N_NN\NANI', 'നടത്തിയവർക്ക്\N_NN\NANI', 'ബിജെപിയുമായും\N_NN\NANI', 'കേന്ദ്ര\JJ', 'സർക്കാരിലെ\N_NN\NANI', 'ഉന്നതതലമായും\N_NN\NANI', 'അടുത്ത\N_NST', 'ബന്ധമുണ്ടെന്ന\VM_VNF', 'ആരോപണം\N_NN\NANI', 'പ്രതിപക്ഷ\N_NN\NANI', 'കക്ഷികൾ\N_NN\NANI', 'നിരന്തരം\RB', 'ഉയർത്തുന്ന\VM_VNF', 'ഘട്ടത്തിലാണ്\VAUX', 'അവർക്ക്\PR_PRP', 'വെടിക്കോപ്പു\N_NN\NANI', 'പകരുന്ന\VM_VNF', 'മധ്യയുടെ\N_NNP\ANI', 'വെളിപ്പെടുത്തൽ\N_NN\NANI', '.\RDPUNC'], ['ഇന്ത്യ\N_NNP\NANI', 'വിടു\N_NN\NANI', 'മുൻപ്\N_NN\NANI', 'ധനമന്ത്രി\N_NN\NANI', 'അരുൺ\N_NNP\ANI', 'ജയ്ഘരിയെ\N_NNP\ANI', 'കണ്ടിരുന്നു\VM_VF', '.\RDPUNC'], ['ചർച്ച\VM_VNF', 'നടത്തിയിരുന്നു\VM_VF', '.\RDPUNC']]
```

Figure 3: Sample text showing the input and output of the Animate Noun Identification module

Table 1: Distinct classes considered for deep level tagging

Label	Class
M/S	Singular male
M/P	Plural male
F/S	Singular female
F/P	Plural female

information are extracted using the following set of rules [20].

4 Experiments and Results

- Predicate —The verb or adjective present in the clause will form the predicate.
- Object —If the predicate is a verb, the noun preceding it will be the object and if the predicate is an adjective, the noun following it will be the object.
- Subject —If the predicate is a verb, the noun preceding the object will be the subject. And if there is no noun preceding the object in the same clause, follow the given rules.
 1. If the predicate of the preceding clause is an adjective, its object will form the subject.
 2. Else the subject of the preceding clause will be the subject.

Conversely, if the predicate is an adjective, the noun preceding it will be the subject.

Finally, the possible candidates are sorted based on the total salience score and the candidate having the highest salience score is selected as the antecedent of the anaphor.

In this section, we will discuss the experiments on each phase of the architecture one by one. The first phase is the preprocessing of raw Malayalam text. We have used NLTK implementation sentence tokenizer and word tokenizer for this purpose [21]. Sentence tokenizer converts raw Malayalam text into a list of sentences, and word tokenizer converts each sentence into a list of words. The preprocessed text is provided to the POS tagging module, where the tagged text is generated. A CRF based tagger developed in our department is employed for this purpose [22]. The third module of the architecture deals with animate noun identification. MLP classifier is used to build the animate noun identifier. A set of nouns belonging to both animate and non-animate category is used as the training data. Each noun in the training data is labelled with corresponding class information (animate/non-animate). A set of 109430 nouns were prepared in this way and used for building the machine learning model. ‘Fasttext’ is used to convert words into vectors. ‘Fasttext’ is an efficient way to convert words to vectors developed by Facebook’s AI research lab [23].

The different configuration of the neural network with various word vector sizes was attempted to build the animate noun identification model. A four-layered architecture with two hidden layers seemed to be the optimum one. The first layer is the input layer which accepts the input. Second and third layers are hidden lay-

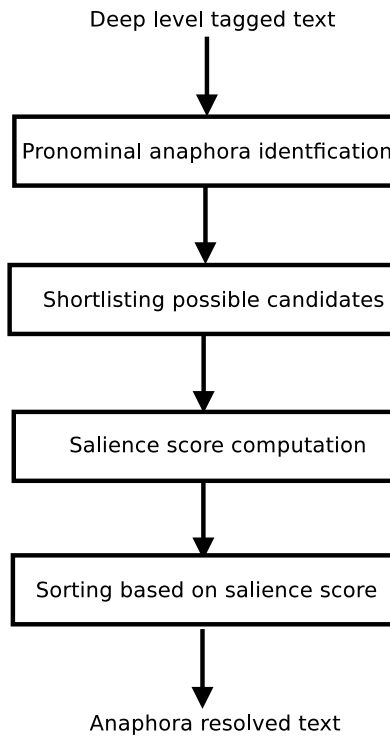


Figure 4: Architecture of anaphora resolution module

Table 2: Salience factors and their weights

Salience factors	Weights
Current sentence	100
Preceding sentences	Reduce sentence score by 10
Subject	80
Object	50
Case	10 to 40
Current clause	100
Immediate clause	50

ers with 'tanh' activation. The last layer is the output layer which generates the output. Softmax is used as the activation function in the output layer. A network with more hidden layers could not improve the performance and only delayed the convergence time. Hence we finalized our hidden layer size as (2,200), where 2 is the number of hidden layers, and 200 is the size of the hidden layers. The maximum accuracy obtained by the classifier on test data is 90.01%. Figure 3 shows the input and output of the animate noun identification module on sample text.

The fourth module of the architecture deals with deep level tagging of animate nouns. MLP classifier is used to build the deep level tagging module. Training data required for the study is a set of names with different number and gender features. Distinct classes considered for the investigation are shown in table 1. 12600 nouns belonging to different categories were prepared for this purpose. Suffixes of various length are used as the feature set for the classifier. Since words and suffixes are symbolic units, it can't be directly fed into neural networks. Hence, we converted our feature set numeric values using Dictvectorizer, a python functionality [24]. Various configurations of the MLP classifier are attempted in our study. A

smaller network was not able to represent the data efficiently and increasing the network size did not improve accuracy. Hence, we experimentally finalized our hidden layer size as (2,200), where 2 is the number of hidden layers, and 200 is the size of each hidden layer. 'Relu' is used as the activation function and 'Adam' as the optimizer. The maximum accuracy was obtained by the classifier when the number of features was eight. The performance of the number and gender classifier with a different number of features is shown in figure 5.

In order to evaluate the performance of the proposed algorithm, experiments were conducted on two different types of datasets. These experiments were aimed to find the contribution of animate noun identifier and deep level tagger of nouns to the overall accuracy of the system.

4.1 Dataset1

This dataset contains short stories from children story domain. We have taken short stories in the Malayalam language from [25], a popular site for Malayalam short stories. Each story contains around 20 to 30 sentences of moderate length. Since children stories follow a

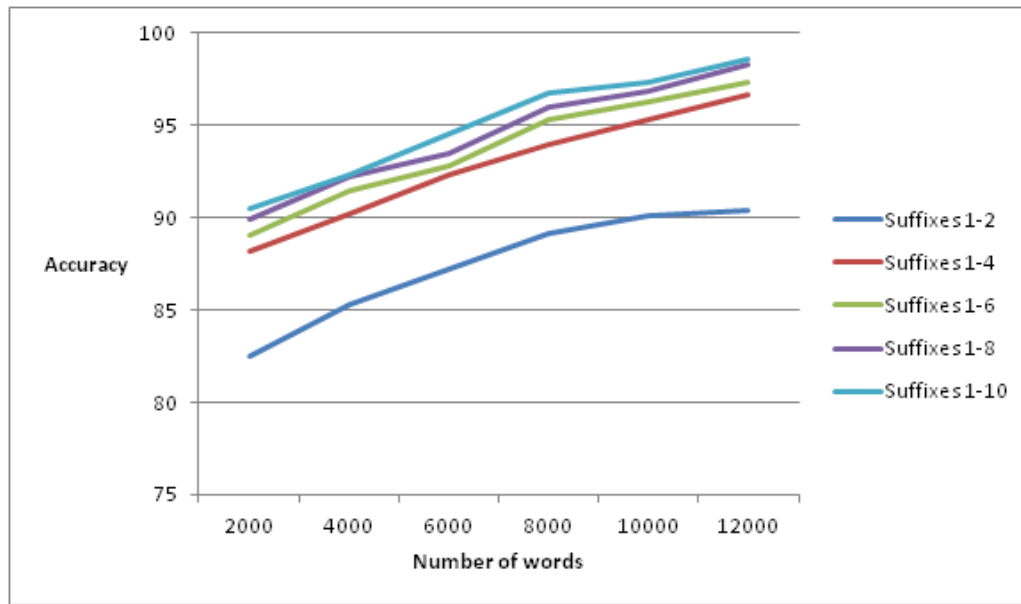


Figure 5: Effect of different features on the performance of the number and gender classifier

Table 3: Results of our experiments on dataset 1

Data set	Total sentences	Total pronouns	Correctly resolved	Accuracy
1	353	19	16	84%
2	466	21	17	80%

straight forward narrative style, the structural complexity of the sentences will be limited. The performance of the anaphora resolution algorithm on this dataset can represent the baseline performance.

4.2 Dataset2

This dataset contains text from news article domain. We have taken news articles from [26], a popular site for Malayalam news. Unlike the children short stories, here the structural complexity of the sentences is not limited. The length of the sentences may go over 25 words. Each article contains around 30 to 40 sentences. Results of our algorithm on both the datasets are summarized in table 3.

From the above experiments, it is observed that the proposed system achieves an overall accuracy of 82.5%. The correctness of the results obtained is verified with the help of language experts. Malayalam is a free word order language, which affects the accuracy of the system. It is observed that the case information associated with the nouns can also decide the antecedent of the pronouns. Some pronouns refer to both animate and inanimate things, which also affects the performance of the system.

5 Conclusion

This paper presents the experimental results of anaphora resolution system in Malayalam language using a hybrid approach. The exclusive feature of the proposed system is the use of machine learning as well as the rule-based technique for the resolution of pronominal anaphors. Malayalam is an agglutinative free word order language, and hence it has many complications in resolving pronouns. The

deeper level analysis of nouns helps in the semantic understanding of natural language text. The power of word embedding is also exploited in the study, which helped in recognizing the animate nouns from non-animate nouns. Experimental results show that the deep level analysis of nouns is essential for resolving the pronominal anaphors present in a text document. The morphological richness of Malayalam language is also utilized in this study with the help of suffix stripping algorithms. In our experiments, we have observed that the increase in morphological features results in the increase in accuracy of tagging systems. Hence, incorporating morphological features in the analysis of natural language text appears to be promising for languages such as Malayalam. In the current work, we have focussed only on the resolution of pronominal anaphora. In future, we aim at the resolution of non-pronominal anaphora such as event anaphora, one-anaphora, etc.

References

- [1] Kath. Word of the day – anaphora. 2018 (accessed December 7, 2018).
- [2] MAK Halliday and R Hasan. 1976: Cohesion in english. london: Longman. 1976.
- [3] Grahame Hirst. Anaphora in natural language understanding. 1981.
- [4] M Sadanandam and D Chandra Mohan. Telugu pronominal anaphora resolution.
- [5] Dan Jurafsky and James H Martin. *Speech and language processing*, volume 3. Pearson London, 2014.
- [6] Daniel G Bobrow. A question-answering system for high school algebra word problems. In *Proceedings of the October 27-29, 1964, fall joint computer conference, part I*, pages 591–614. ACM, 1964.

- [7] Terry Winograd. Understanding natural language. *Cognitive psychology*, 3(1):1–191, 1972.
- [8] Yorick Wilks. Preference semantics. Technical report, STANFORD UNIV CA DEPT OF COMPUTER SCIENCE, 1973.
- [9] Jerry R Hobbs. Pronoun resolution. *ACM SIGART Bulletin*, (61):28–28, 1977.
- [10] David Maclean Carter. *A shallow processing approach to anaphor resolution*. PhD thesis, University of Cambridge, 1986.
- [11] Barbara J Grosz. The representation and use of focus in dialogue understanding. Technical report, SRI INTERNATIONAL MENLO PARK CA MENLO PARK United States, 1977.
- [12] Shalom Lappin and Herbert J Leass. An algorithm for pronominal anaphora resolution. *Computational linguistics*, 20(4):535–561, 1994.
- [13] Ben Wellner, Andrew McCallum, Fuchun Peng, and Michael Hay. An integrated, conditional model of information extraction and coreference with application to citation matching. In *Proceedings of the 20th conference on Uncertainty in artificial intelligence*, pages 593–601. AUAI Press, 2004.
- [14] Eugene Charniak and Micha Elsner. Em works for pronoun anaphora resolution. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 148–156. Association for Computational Linguistics, 2009.
- [15] L Sobha and BN Patnaik. Vasisth: An anaphora resolution system for indian languages. In *Proceedings of International Conference on Artificial and Computational Intelligence for Decision, Control and Automation in Engineering and Industrial Applications*, 2000.
- [16] S Athira, TS Lekshmi, RR Rajeev, Elizabeth Sherly, and PC Reghuraj. Pronominal anaphora resolution using saliency score for malayalam. In *2014 First International Conference on Computational Systems and Communications (ICCSC)*, pages 47–51. IEEE, 2014.
- [17] L Sobha. Resolution of pronominals in tamil. In *2007 International Conference on Computing: Theory and Applications (ICCTA'07)*, pages 475–479. IEEE, 2007.
- [18] Pedregosa Fabian, Varoquaux Gaël, Gramfort Alexandre, Michel Vincent, Thirion Bertrand, Grisel Olivier, Blondel Mathieu, Prettenhofer Peter, Dubourg Vincent, Vanderplas Jake, et al. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
- [19] MS Bindu and Sumam Mary Idicula. A hybrid model for phrase chunking employing artificial immunity system and rule based methods. *International Journal of Artificial Intelligence & Applications*, 2(4):95, 2011.
- [20] Rajina Kabeer and Sumam Mary Idicula. Text summarization for malayalam documents—an experience. In *2014 International Conference on Data Science & Engineering (ICDSE)*, pages 145–150. IEEE, 2014.
- [21] Edward Loper and Steven Bird. Nltk: The natural language toolkit. In *Proceedings of the ACL-02 Workshop on Effective tools and methodologies for teaching natural language processing and computational linguistics-Volume 1*, pages 63–70. Association for Computational Linguistics, 2002.
- [22] AP Ajees and Sumam Mary Idicula. A pos tagger for malayalam using conditional random fields.
- [23] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146, 2017.
- [24] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct):2825–2830, 2011.
- [25] Bed time stories - wonderful stories in malayalam for kids. <https://kutykadhakal.blogspot.com/>. Accessed: 2018-09-30.
- [26] Madhyamam. <https://www.madhyamam.com/>. Accessed: 2018-08-28.