Pronominal Resolution in Tamil using Machine Learning

K. Narayana Murthy, L. Sobha & B. Muthukumari

AU-KBC Research Centre, Chennai, India
knmuh@yahoo.com,sobha@au-kbc.org

Abstract

In this paper we present our on going work on pronominal resolution in Tamil using two different approaches, viz Salience Measure approach and the Machine Learning approach. Initially we validate the salience measure approach. The analysis depends on the salience weight of the candidate nominals (NP) for the antecedent-hood of the pronominal from the list of possible candidates for antecedent-hood. The salience weight of an NP is obtained from the salience factors, which are determined by the likelihood of an NP to be the antecedent on the basis of the grammatical features of the head of NP. The salience factors arrived in this approach are then used as features in the Machine learning approach. Both the methods have been tested on generic data and encouraging results have been obtained.

1 Introduction

Pronominal resolution refers to the problem of determining the noun phrase (NP) that refers to a pronominal in a document. The most common type of anaphor is the pronominal anaphora and it can be exhibited by personal, possessive or reflexive pronouns. The major classifications in pronominals are the first, second and third person pronouns. First and second person singular and plural are commonly used as deictic, though they are used in anaphoric form in discourse. There are many approaches to solve this problem such as rule based, statistical and machine learning based approaches.

In anaphora resolution research, pronominal resolution generally took priority over non-pronominal resolution. As far as the latter is concerned, resolution can be achieved using syntactic information alone, whereas in the case of the pronominals this is not possible. At the syntactic level, resolution means assignment of one or more candidate antecedents to pronominals and the ambiguity that remains can only be resolved with the help of world knowledge. The problem of pronominal resolution was first stated (Hobbs 1978) by Jesperson in 1954 who observed that ”an ambiguity may sometimes arise when there are two antecedents to which it may refer: *If the baby does not thrive on raw milk, boil it*”. Hobbs’ algorithm (Hobbs 1978) depends on a simple tree search procedure, which is called a naïve approach. The other approach, according to Hobbs (1978), is the semantic approach using predicate calculus and one arrives at the view that the second approach yields a better result in comparison with the naïve one.

Systems developed after 1986 can be grouped into two classes: those based on integrated (or knowledge-based) approaches and those based on on alternative approaches, some examples of which being (Carbonell and Brown 1988). Among the alternative approaches to anaphora resolution ones that use statistical methods such as probability and Bayesian conditional probability are exemplified by

(Lappin and McCord 1990, Mitkov 1997, Kennedy and Boguraev 1996, Sobha and Patnaik 1998, inter al.). Lappin and Leass (1994) give an algorithm, which works on the syntactic representations generated by Slot Grammar parser (Lappin and McCord 1990) and relies on salience measures derived from the syntactic structure and a simple dynamic model of attentional state to select the antecedent noun phrase of a pronoun from a list of competing candidates. The salient weights are assigned using certain linguistics-based ideas like "the primacy of the subject over the object" as a consequence of which the subject NP gets more weight than the object NP, etc.

Here we present pronominal resolution of Tamil using salience measures (hereafter referred to as salience weight) and machine learning approaches. Initially we validate the salience measure approach. The analysis depends on the salience weight of the candidate nominals (NP) for the antecedent-hood of the pronominal from the list of possible candidates for antecedent-hood. The salience weight of an NP is obtained from the salience factors, which are determined by the probability of an NP to be the antecedent on the basis of the grammatical features of the head of NP. The salience factors arrived at in this approach are then used as features in the machine learning approach. Both the methods have been tested on generic data and encouraging results have been obtained.

In arriving at the salience weights we use linguistic knowledge in the form of salience factors. The salience factors are given salience weights using constraints and preference. The salience factors decide the probability of a candidate NP becoming an antecedent. The system uses a shallow parser. This parser is generic and not closely tied down to any one linguistic formalism. This makes our approach more suitable for relatively free word order languages such as Tamil, an Indian language.

The paper has the following sections: In the second section we give a brief introduction to the language under consideration, Tamil, and examples of pronoun-antecedent relationship in the language. An overview of Lappin and Leass method and the machine learning approach used here are discussed in the third section. Fourth section gives a detailed description of the implementation and evaluation of the two approaches used in the present study. The last section deals with the results, followed by a discussion and conclusions.

2 Brief Description of Tamil Language

Before we get into the details of our approach, we would like to provide the necessary information regarding the language under consideration, Tamil. Tamil belongs to the South Dravidian family of languages. It is a verb final language and allows scrambling. It has post-positions, the genitive precedes the head noun in the genitive phrase and the complementizer follows the embedded clause. Adjective, participial adjectives and free relatives precede the head noun. It is a nominative-accusative language like the other Dravidian languages. The subject of a Tamil sentence is mostly nominative, although there are constructions with certain verbs that require dative subjects. Tamil has PNG (person, number, and gender) agreement.

Let us now consider the pronouns in Tamil. They are as follows: First Person, Singular: na:n "I", Plural : na:nnal "we"; Second Person, Singular: ni: "you", Plural: ni:nnal "you" and Third Person, Masculine Singular: avan "he", Feminine: aval "she", Singular Neuter: atu "it" and Plural Neuter: avar "they". Thus the third person pronouns show singular-plural distinction and also masculine-feminine distinction. They also take the entire range of case inflections. In Tamil *atu* "it" can be neuter gender third person pronominal as well as a deictic marker.

Now let us consider the relationship between the pronominals and their antecedents in Tamil. For all pronouns, noun phrase cannot co-refer if they have incompatible number, gender and person agreement. Consider the Tamil sentence

1. si:ta avalai$_i$ atitta:l enRu kavita$_i$ conna:r
   sita she(acc) beat(pst) compl kavita say(pst)
   (Kavita said that Sita hit her)

Here the pronominal is avalai "she" and is the third person accusative pronominal with the agreement feature "third person, feminine, singular". The antecedent NP to this pronominal should have the same features as the pronominal. The pronoun is in the accusative form in the above sentence. The antecedent of the accusative pronoun avalai is *"kavita"* which is in the immediate clause. From the above we conclude that the antecedent of an accusative pronoun is the subject of the immediate clause before/after the clause in which the pronoun occurs. The antecedent of a non-possessive pronoun could be also in the non-immediate clauses as well. A pronoun must agree in number, gender and person with the antecedent as is clear from the above example. The antecedent to the pronoun "*avalai*" is outside the clause or sentence in which the pronominal occurs. Hence kavita is the antecedent of *avalai.*

2.1 Possible Antecedents

In English and other fixed word order languages we consider all the NPs that precede the pronominal as the probable candidates for the antecedent-hood. But in the case of relatively free word order languages this could not be the criterion. There are cases when the antecedent follows the pronoun and is still not used in the cataphoric form. This happens because the language allows main and subordinate clause inversion. That is, the main clause could be moved to the left of the subordinate clause and thus the antecedent could follow the pronoun. This inversion is possible only with certain type of clause constructions such as the Complimentizer clause and the Relative participial clause. Consider the following examples:

2. netru vanta avan$_i$raman$_i$
   Yesterday come+RP he Raman
   (He who came yesterday is Raman)

3. ravi vantaal aval$_i$ vittukku centrum entru sita$_i$ connaar
   ravi came+cont she house+dat go+Fut COMPL Sita say+pst
   (Sita said that if Ravi comes she would go home.)

In the above cases you could see that the antecedent is after the pronoun. If we analyse the sentences under the contemporary GB grammar formalism, the sentences adhere to the Binding principle. Hence the NPs following the pronoun are also to be considered as possible antecedents.

3 The Two Approaches

The two approaches used for developing the system are dealt in detail in the following sections. Though we say that one of the approaches we are using is similar to Lappin and Leass method, it can be seen that we have substantially moved from their approach.

3.1 Lappin and Leass Algorithm

The Lappin and Leass (1994) anaphora resolution algorithm uses salience weights in determining the antecedent to pronominals. It requires as input a fully parsed sentence structure and uses hierarchy in identifying the subject, object etc. The salience weights are as given below (Table 2). This algorithm uses syntactic criteria to rule out noun phrases that cannot possibly co-refer with it. The antecedent is then chosen according to a ranking based on salience weights.

Factors	Weights
Sentence recency	100
Subject emphasis	80
Existential emphasis	70
Accusative emphasis	50
Indirect object/oblique	40
Head noun emphasis	80
Non-adverbial emphasis	50

Table 1: Lappin and Leass Salience Weights

The candidates that agree in PNG are ranked according to their salience. A salience value is then calculated. Consider the sentence *Mary saw Bill*. The salience of Mary is

Mary = W_{sent} +W_{subj} +W_{head} + $W_{non=adv}$
= 100 + 80 + 80 + 50
= 210

The candidate with the highest salience is considered as the antecedent for the pronominal.

3.2 Multiple Linear Regression as a Classification Technique

Regression analysis is a statistical technique for investigating and modeling the relationship between variables in a system (Montgomery, Peck and Vining 2001, Glantz and Slinker 2000, Allison 1999). When there are more than two variables in the system, the term multiple regression is employed. Regression is often used

as a modeling technique where the value of one of the selected variables, called the response variable, is determined by the values of the other independent variables, also called the regressors. The modeling process basically involves determining parameters of the model, i.e. the weights of the regressor variables. The model itself could be linear or non-linear in the parameters. Regression distinguishes the response veritable from the regressors and is thus generally considered to be a non-symmetric technique.

Here we show how Multiple Linear Regression can be used as a two-class classification tool (Murthy and Bharadwaja Kumar 2006). The regressor variables are the feature vectors extracted from the training data. Since we are using regression for classification rather than for modeling, no particular feature is selected as a response variable or expressed in terms of the other features. We posit a separate decision variable, whose value is determined in terms of the actual features. The method is thus symmetric in the features. We give below a brief formulation of the Multiple Linear Regression as a classification technique.

Suppose there are k features. Let x_{ij} denote the i^{th} observation of feature x_j where $i = 1, 2...n$ and $j = 1, 2...k$. Let y_i be the i^{th} observed value of the decision variable. Then

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + ... + \beta_k x_{ik} + \epsilon_i \tag{1}$$

where the parameters $\beta j,\ j = 0, 1, 2...k$ are called regression coefficients and ε_j are called error terms or residuals. The regression coefficients are the parameters in the model. Note that the equation is linear in the parameters. The aim is to estimate the values of these parameters from training data. In matrix notation, we have

$$y = \mathbf{X}\beta + \epsilon \tag{2}$$

where y is an $n \times 1$ vector of observations, $\mathbf{X}$ is an $n \times p$ matrix of feature values, where p is $k + 1$, β is $p \times 1$ vector of the regression coefficients, and ε is an $n \times 1$ vector of error terms. We may estimate the values of the parameters $\hat{\beta}$ using the least square method. The equation for estimating the parameters can be written as

$$\hat{\beta} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'y \tag{3}$$

where $(\mathbf{X}'\mathbf{X})^{-1}$ exists provided the features are linearly independent.

In order to determine the parameters, we need to know the value of the decision variable on the left hand side of the regression equation. Since the decision variable is not a feature in the system but an additional variable introduced for the purposes of using regression as a classification technique, the value of the decision variable can be chosen arbitrarily subject to the following constraints. In order to ensure adequate separation between the two classes, the values for the two classes must be clearly separated. Also, the choice of the values for the decision variable influences the range of values for the parameters - the value chosen must result

in reasonable ranges of values for the parameters, avoiding overflows and underflows in the extreme. Finally, choice of symmetric values for the two classes in the two-class case makes the decision rule and thresholding for rejection simpler. In practice the values are decided after a bit of experimentation with the actual data on hand.

For the two-class classification problem, we use differential features -actual value of each feature is calculated as the difference between the values of the feature for the two classes. This will trivially extend to cases where the features are binary valued. Values of the decision variable for the two classes are chosen symmetrically around zero and the parameters are estimated from the training data. A test sample can then be classified as belonging to class Cl or C2 depending upon whether the value of the decision variable is positive or negative. It is possible to reject a point if the value of the decision variable is too close to zero, say, closer than a specified threshold. The method has been successfully applied to two-class classification problems earlier (Murthy and Bharadwaja Kumar 2006).

Classification performance can be measured in terms of Accuracy, or Precision and Recall, or using some combined measure such as the F-Measure.

We have outlined a general method for supervised two-class classification using Multiple Linear Regression. The method is conceptually simple and based on sound theoretical foundations. Techniques exist for validating the adequacy of the model for a given problem and for evaluating the relative significance of the various features (which can also be used for feature selection). The method is symmetric in the features. Although matrix inversion is required for estimating the values of the parameters, once the model is built, classifying objects is very efficient - only computation of the linear regression equation and checking the sign of the decision variable are required. The technique is thus highly suitable for two-class classification problems with a reasonably small number of features. In this paper we show the application of the MLR technique for the task of anaphora resolution.

4 About the Present System

In this section we discuss how the above two approaches are implemented and we also give evaluation of the two approaches.

4.1 Salience Factors and Weights

The method adopted here uses salience weights in identifying the antecedent of a pronominal. The salience weights are arbitrary but the salience factors are arrived at using linguistic analysis. The method envisaged here has correlation with Lappin and Leass method but has deviated substantially from it. In their approach there is a hierarchical structure whereas the approach outlined here does not exploit the hierarchy, but exploits the nominal morphology. The salience measures in our work are different and are arrived at more from the discourse analysis point of view. As in Lappin and Leass method, fully parsed output is not required for

the analysis in our work, only a very shallow parsed output is sufficient.

Initially, values of salience factors were manually assigned based on linguistic considerations and fine tuned through experimentation. The following weights (See Table 3) gave best performance after experimental fine-tuning. We give below a detailed description of salience factors and the weights given for each factor.

Our analysis showed that the subject of the sentence or clause could be the most probable antecedent for the pronoun. The subject of a sentence could be identified by the case markings the nouns take. According to that, in Indian languages, a Nominative noun, a Possessive NP with a nominative head and a Dative noun could become a subject of a sentence. These languages also have a peculiarity of having Dative nouns as subject. Certain class of verbs called cognitive verbs such as "like", "understood" etc take only Dative noun as subject. So we have three types of subject nouns and in that, the most common is Nominative noun. Hence Nominative noun is given a very high score of 80 and the other two are give a score of 50 each. An NP with accusative case gets the next highest score of 40. NPs with other case markings (N. other) get a score of 30.

The current sentence in which the pronoun occurs gets a score of 50 and this gets reduced by 10 for each preceding sentence. We consider up to four sentences preceding the sentence containing the pronoun. Thus the fifth sentence gets a score of "10". The salience factors and weights used are given in the tables below.

Salience Factors	Definitions
Current sentence	Sentence under consideration
Current clause	Clause in which the anaphor occurs
Immediate clause	Clause next to the current clause (Immediately preceding or following)
Non-immediate clause	Neither an immediate nor a current clause
N-Nom	Any nominative noun
N-Poss	Any possessive noun
N-Dat	Any dative noun
Others	Any noun with case suffices other than above mentioned

Table 2: Definitions of the factors

4.2 System Architecture

The working of our system is as follows: The input documents are pre-processed through steps such as sentence splitting, morphological analysis, POS tagging, NP chunking and clause boundary identification. At first the text is analysed by the morphological analyzer (Viswanathan, Ramesh Kumar, Kumara Shanmugam, Arulmozi and Vijay Shanker 2003) where each word is analysed for the suffixes that are attached to it. This system is developed using Finite State Automata and has a dictionary of 33,000 root words. The system has been tested on a 3 million-word corpus with 98% accuracy.

Salience Factors	Weights
Current sentence	50
Subsequent sentences upto four sentences from the current sentence	Reduce by 10
Current clause	75
Immediate clause	70
Non-immediate clause	65
N.Nom	80
N.Poss	50
N.Dat	50
N.Acc	40
N.others	30

Table 3: Salience factor weights for Tamil

The morphologically analysed text gets POS tagged by a rule-based tagger (Arulmozhi, Sobha and B. 2004). The rule-based system uses lexical rules and context sensitive rules. There are 53 major tags and 75 sub tags that include case markers, number gender person markers, clitics etc. The POS tagged output is NP Chunked using a rule based chunker. The clause boundaries are identified using the clause splitter, which is a rule-based one. It uses clause marker on the verb as the point of identification of the presences of a clause. Then the subject and object of the clause is identified using selectional restrictions. For selectional restriction, we use the sub-categorization of the verb. This system performs with 65% accuracy. The output is then manually checked. The POS tagger or the morphological analyzer is not able to give the information regarding honorific Proper nouns. Since this type of nouns have occurred in large numbers in the corpus we chose we had to manually add this tag. For example the pronoun that refers to "Mother Teressa" will be in plural form. Hence we tag "Mother Teressa" with honorific marker. We also correct the tags if there is an error from the POS, NP chunker and the clause splitter. The schematic representation of the system is given below.

$$Input \rightarrow pre\text{-}processor \rightarrow Salience - Factor - Assigner \rightarrow Output$$

We consider four sentences before the sentence containing the pronoun for finding the antecedent. The pre-processed output goes into the salience factor assigner component. All the NPs, which precede an anaphor (in certain cases the NPs which follows the pronoun is also considered as explained in section 2), are taken as possible candidate. Then the candidate NPs that are identified and checked for PNG and the salience weights are given. The salience weight of a noun is the weighted sum of all salience factor values. The salience factors get the value from the grammatical features of head of NP. The noun with the maximum salience weight is considered as the antecedent of the pronoun. Consider the output from the system for the sentence (4) given below.

4. si:ta avalai atitta:l enRu kavita conna:r
   sita she+acc beat+pst compl kavita say+pst
   (Kavita said that Sita hit her)

In this example *avalai* is the pronominal and the candidate NPs are *sita* and *kavita*. The salience factor weights are computed as follows:

sita = $w_{current} + w_{N.nom}$ = $50 + 80 = 130$

kavita = $w_{current} + w_{current-clause} + w_{N.nom}$ = $50 + 75 + 80 = 205$

Here *kavita* with the highest salience weight is considered as the antecedent of *avalai* after the PNG check.

5 Results and Discussions

The system has been evaluated on a generic text taken from the CIIL (Central Institute of Indian Languages, Mysore, India) corpus, which talks about Mother Teressa and her work. The number of pronouns found in a text of 500 sentences is 226. Other than these pronouns there were deictic (*atu* "that") pronouns, which we had to manually remove. We have checked the type of NP that is taken as the antecedent by each pronoun and calculated the precision and recall for each. We have checked each case manually. Out of the 226 pronouns we have considered, 212 got extracted and 183 were correct extractions. Thus we get an overall precision of 86.32% and Recall of 80.9% for the text we have used. The table given below shows the precision and recall for each type of antecedent and the overall performance of the system.

Case	Total	Extracted	Correct	Precision (%)	Recall (%)
Nom	167	160	143	89.37	85.62
Acc	20	20	11	55.00	55.00
Gen	20	17	15	88.23	75.00
Dat	9	7	6	85.71	66.66
Loc	6	4	4	100.00	66.66
Inst	4	4	4	100.00	100.00
Total	226	212	183	86.32	80.90

Table 4: Precision and Recall of the system

There are cases where we got multiple NPs with the same salience score for a pronoun. In some cases the NPs had the same case markings and when checked manually they were all definite descriptions. Though they are theoretically correct we did not take them as correct extraction. In some cases the NPs had different case markings. We found 5 cases, where there are 3 NPs identified as the antecedent which had different case markings. We are trying to fine tune the salience weights and add some more linguistic rules to overcome this defect.

5.1 Anaphora Resolution using Regression

Motivated by the encouraging results obtained from the above method using salience weights, experiments were conducted on the same data using the same set of salience factors as features using Multiple Linear Regression as a classifier. The salience factors were modeled as binary features (for example, whether the candidate NP is within the current clause or not). Training data included 175 pronouns and 503 candidate NPs, all satisfying the agreement requirements, spread over 442 sentences. During the training phase, the values of the regression coefficients were computed. In the testing phase, the response variable was computed for each candidate NP using the beta values already obtained in the training phase. The candidate NPs were ranked on the values of the response variable and the candidate NPs classified accordingly. Since there can be only one antecedent for a given pronoun, the candidate NP whose response variable is closest to the positive class was taken as the computed antecedent. Performance obtained is given below.

65% of the pronominals were correctly resolved when the top-most ranked candidate NP was taken as the obtained antecedent. The top two candidate NPs included 88.6% of the correct antecedents. The top three NPs included 94.9% of the correct antecedents. This was as expected because although the number of candidate NPs for a given pronoun varied from 1 to 18, the number of cases with more than four possible antecedents were relatively less. It was also observed that the Beta values obtained corresponded roughly in relative significance to the weights used for the salience factors in the earlier method, thereby providing mutual support to the two methods. The only significant difference that was observed was that the "current" sentence feature was somewhat subsumed by the "current clause" feature, indicating that in the training data, the antecedent was, more often than not, inside the current clause, whenever it was inside the current sentence.

6 Conclusions

In this paper we have described our experiments in resolving pronominals in Tamil within the inter-sentential level using two approaches viz, salience measure and machine learning approaches. The first approach substantially deviates from Lappin and Leass method. In their approach there is a hierarchical structure whereas the approach outlined here does not exploit the hierarchy, but exploits the nominal morphology. Here our salience factors are more granular and fine-tuned for the relatively free word order languages. The salience factors that we have arrived here are not language dependent and hence can be used across languages which are morphologically rich and relatively free ward order such as other Indian Languages.

Motivated by the encouraging results we have obtained by the above method, a machine learning approach based on regression was developed. Machine learning approaches have the advantage that feature weights are estimated automatically from training data. Results obtained are again encouraging.

Further work is in progress for handling other types of anaphora as well. Pronominal resolution will be of value for all natural language processing applications such as question answering, information extraction and machine translation.

References

Allison, P.(1999), *Multiple Regression: A Primer*, Pine Forge Press.

Arulmozhi, P., Sobha, L. and B., K. S.(2004), Part of speech tagger for tamil, *Symposium on Indian Morphology, Phonology and Language Engineering*, March 19-21, 2004 IIT Kharagpur, 55-57, India.

Carbonell, J. and Brown, R.(1988), Anaphora Resolution: A Multi-strategy Approach, *Proceedings of the 12th International Conference on Computational Linguistics*, pp. 96–101.

Glantz, S. and Slinker, B.(2000), *Primer of Applied Regression and Analysis of Variance (2nd ed.)*, McGraw-Hill, New York.

Hobbs, J.(1978), Resolving pronoun references, *Lingua* **44**, 311–338.

Kennedy, C. and Boguraev, B.(1996), Anaphora for Everyone: Pronominal Anaphora Resolution without a Parser, *Proceedings of the 16th International Conference on Computational Linguistics COLING'96*, Copenhagen, Denmark, pp. 113–118.

Lappin, S. and Leass, H.(1994), An Algorithm for Pronominal Anaphora Resolution, *Computational Linguistics* **20(4)**, 535–561.

Lappin, S. and McCord, M.(1990), Anaphora Resolution in Slot Grammar, *Computational Linguistics* **16(4)**, 197–210.

Mitkov, R.(1997), Factors in Anaphora Resolution: They are not the only things that matter. a case study based on two different approaches, *Proceedings of the ACL'97/EACL'97 Workshop on Operational Factors in Practical, Robust Anaphora Resolution*, Spain, pp. 14–21.

Montgomery, D., Peck, E. and Vining, G.(2001), *Introduction to Linear Regression Analysis*, John Wiley & Sons, Inc., New York.

Murthy, K. N. and Bharadwaja Kumar, G.(2006), Language Identification from Small Text Samples, *Journal of Quantitative Linguistics* **13(1)**, 57–80.

Sobha, L. and Patnaik, B. N.(1998), An Algorithm for Pronoun and Reflexive Resolution in Malayalam, *Proceedings of the International Conference on Computational Linguistics, Speech and Document processing*, pp. C63–66.

Viswanathan, S., Ramesh Kumar, S., Kumara Shanmugam, B., Arulmozi, S. and Vijay Shanker, K.(2003), A Tamil Morphological Analyser, *Proceedings of the International Conference On Natural language processing ICON 2003*, Central Institute of Indian Languages, Mysore, India, pp. 31–39.