A Semi Supervised Dialog Act Tagging for Telugu

Suman Dowlagar
suman.d@research.iiit.ac.in

Radhika Mamidi
radhika.mamidi@iiit.ac.in

Abstract

In a task oriented domain, recognizing the intention of a speaker is important so that the conversation can proceed in the correct direction. This is possible only if there is a way of labeling the utterance with its proper intent. One such labeling techniques is Dialog Act (DA) tagging. This work focuses on discussing various n-gram DA tagging techniques. In this paper, a new method is proposed for DA tagging in Telugu using n-gram karakas with back-off as n-gram language modeling technique at n-gram level and Memory Based Learning at utterance level. The results show that the proposed method is on par with manual DA tagging.

Keywords: Dialog Acts, Intention Recognition, Dialog System, n-grams, Karaka Dependencies, Back-off, Memory Based Learning.

1 Introduction

The term 'dialog' origins from the Greek word *dialogos* which means conversation. A Dialog system which is a conversational agent which converses with human.

విద్యార్థి	:	నమస్కారం సర్.
student	:	Hello sir.
లైబ్రేరియన్	:	నమస్కారం.
librarian	:	Hello.
విద్యార్థి	:	నాకు ఈ పుస్తకము జారీ చేయండి.
student	:	I want to issue this book.

Table 1: Conversation between a student and a librarain from ASKLIB corpus in Telugu with English translation

The crucial use of a dialog system is to convert simple yet complicated tasks from manual to automated. The process of understanding and generating the dialogs is known as dialog modeling. In dialog modeling, to understand the dialogs, speaker's intent must be recognized. The recognition of the speaker's intent is done with the help of Dialog Acts. Dialog Acts is a tagset that classifies utterances based on pragmatic, semantic and syntactic features. Dialog Acts are similar to Austin's speech acts. According to Austin (1975), a speech act represents the meaning of an utterance at the level of illocutionary force. DA tagging is assigning a Dialog Act to an utterance from the given DA tagset.

Earlier, research in DA tagging was limited to linguistic domain, but now with the help of statistics, machine learning and pattern matching, automated DA tagging with various DA recognition approaches (Král and Cerisara, 2012) have come into existence. Some of the DA tagging methods include word based DA tagging (Garner et al., 1996), which shows that individual words are the potential source for tagging utterances in dialogs. On the other hand, (Webb et al., 2005) used n-grams with *predictivity criterion* for DA tagging which shows that instead of considering all n-grams, take only those which surpass the threshold. (Klüwer et al., 2010), proved that n-grams obtained from dependency parsing are powerfull enough for DA tagging. (Liu et al., 2013) and (Rotaru, 2002), proved that memory based learning techniques can be used for DA tagging. Other methods for DA tagging include Naive Bayesian interpretation (Reithinger and Klesen, 1997), Hidden Markov Models (Stolcke et al., 2000).

Telugu is a free word order language. Existing n-gram cue based methods are mainly

D S Sharma, R Sangal and E Sherly. Proc. of the 12th Intl. Conference on Natural Language Processing, pages 376–383,
Trivandrum, India. December 2015.

Tag list	**%**	**Description of Tags**	**Example of utterances translated to English**
RETURN	2.93	Utterance intent is to return the book	*I am returning this book.*
TIME_ASSERT	0.55	Speaker makes a claim with respect to library timings	*Now the time is 6.30pm.*
ISSUE_INFO_REQUEST	1.92	Utterance is bound to provide answer related to issue of book	*Do you want to issue this book?*
ISSUE	3.88	Utterance intent is to issue the book	*Issue this book to me*
REISSUE_INFO_REQUEST	1.92	Utterance is bound to provide answer related to reissue of book	*Are you willing to reissue this book?*
ISSUE_ASSERT	3.68	Speaker makes a claim while issuing the book	*You have to return this book in a month, or else you will be charged.*
ACCEPT_ACKNOWLEDGE	6.46	Utterance indicates speaker has understood and accepted the stated fact.	*Ok sir, I will return this book in a month.*
ASSERT	0.25	Speaker states a general fact with out any relation to issue, reissue, return etc	*Days have passed since I saw you.*
COMMIT	9.99	Utterance states that speaker is committing to perform the action in future.	*Ok sir, I will come tomorrow.*
GREETINGS_REPLY	12.26	Utterance is replying to a greet so as to maintain the conversation	*I am fine.*
ANSWER	2.37	Utterance is answering to the question	*The author of this book is Dan Jurafsky.*
GREETINGS	7.67	Utterance states that the conversation is started	*Good Morning sir.*
ACCEPT	6.16	Utterance shows speakers agreement to the proposal or claim	*Ok sir.*
INFO_REQUEST	6.76	Utterance that is a generic question without relation to any domain specific task	*How are you sir?*
RETURN_ASSERT	3.08	Speaker makes a claim while issuing the book	*I am deleting this book from your account.*
GREETINGS_EOC	12.26	Utterance states that the conversation is completed	*Thank you.*
RETURN_INFO_REQUEST	2.32	Utterance is bound to provide answer related to return of book	*Do you want to return this book?*
REISSUE_ASSERT	2.77	Speaker makes a claim while returning the book	*I am extending the book's due date.*
REISSUE	3.13	Utterance intent is to reissue the book	*Please, reissue this book to me.*
ACTION_DIR	9.64	Utterance intent is to make hearer, perform an action	*Give me the id card and the book.*

Table 2: Showing the tagset, its percentage in corpus, description of each tag with an example (written originally in Telugu) translated to English.

Speaker		Dialog in Telugu with English Translation	DA tag
విద్యార్థి	:	నమస్కారం సర్.	GREETINGS
student	:	Hello sir	
లైబ్రేరియన్	:	నమస్కారం.	GREETINGS_REPLY
librarian	:	Hello	
విద్యార్థి	:	నాకు ఈ పుస్తకము జారీ చేయండి.	ISSUE
student	:	I want to issue this book	

Table 3: Conversation between a student and a librarian with its respective DA tags

developed for English, When these are applied to DA tag Telugu dialogs, the baseline accuracy is not reached, because n-gram methods are position dependent. In this paper, a new method is proposed for DA tagging in Telugu using n-gram karakas with back-off and Memory Based Learning such as kNN. The novel method is compared with n-gram and a combination of unigram methods. The results show that the proposed method does better DA tagging for Telugu.

This paper is organized as follows. Section 2 gives an overview of the corpus and tagset used. In section 3 we present our work, section 4, results are tabulated, section 5 draws some important conclusions and finally section 6 gives the scope for future work.

2 Corpus

At present, there is no available corpus related to task oriented Telugu dialogs. Our work started with the construction and acquisition of the dialogs in Telugu.

The focus was on task oriented, domain dependent dialogs with *'Library'* as the domain. We named the corpus as ASKLIB. ASKLIB consists of nearly 225 dialogs that took place between students and the librarian. This corpus is also collected by frequently visiting different libraries and observing how people interact with the librarian. The data acquisition was also done through the *Wizard of Oz* technique. 27 active participants were told to assume the scenario of a library and were asked to write a few generic 2 party conversations. After the corpus acquisition, we observed that the dialogs pertaining to the library domain could be broadly classified into 4 types, viz. ISSUE, REISSUE, RETURN, ENQUIRY. In other words, a person's interaction with a librarian can result in either issue, reissue or return of a book or any enquiry related to a book/the library. The data that was collected has undergone various layers of automated spell checking using CALTSLAB Spell Checker (`http://caltslab.uohyd.ernet.in/spell_checker.php`) and manual spell checking to make the corpus reliable and correct.

Words	Dialogs	Utterances per Dialog
12826	225	7-9

Table 4: Corpus Statistics

Table 4 gives the information about the number of words, dialogs and utterances per dialog present in ASKLIB corpus.

The DA tagset is based on DAMSL (Core and Allen, 1997) and some domain dependent tags. DAMSL is one of the domain independent tagsets used for DA Tagging. The reason for choosing domain dependent tags is that, as the ASKLIB data is categorized to 4 types, the questions raised and assertions clamied in each category will be different. Hence, a modified DAMSL tagset is created to suit the library domain by adding a set of domain dependent tags along with some of the DAMSL tags. At present, our tagset consists of a total of 21 tags. The tagset and its related information is given in table 2. Table 3 gives a sample conversation taken from ASKLIB corpus with DA tags; the utterances are given in Telugu with their English translation.

3 Classification Algorithms

Of all the methods developed for DA tagging, the easiest way to automatically tag the test utterance is by matching the test utterance to any of the utterances present in the training data. The problem with this method is the unlikely occurence of the same utterance in both the training data and the test data. It will also occupy a lot of memory (for huge corpus) as each and every unique utterance with its corresponding tag is stored in the training data. As the huge corpus was difficult to handle, the focus was made on words. The problem of considering only words is that words do not contain any local contextual information. Later, the methods were extended to n-grams. The n-gram methods consist of splitting the utterances into a sequence of n words. These n-grams are called cues. In cue based n-gram DA tagging techniques, the n-grams obtained from the training data act as cues. By matching n-grams of the training data with n-grams of test data, an appropriate tag to the test data was given. Also, the size of unique n-grams obtained from training data is very less when compared to the size of unique utterances in training data.

In this paper, we talk about three methods. They are:

1. DA tagging using n-grams,

2. Combinations of unigrams with Naive Bayesian plus k Nearest Neighbors(kNN) and

3. Our method, n-grams with karaka dependency relations between them using back-off plus Memory Based Learning such as kNN.

3.1 N-gram Method

Of all the approaches in DA tagging, cue based n-gram tagging methods are proven to be the easiest and the most powerful DA tagging scheme. In n-gram methods, the tag of the test utterance is obtained by converting the test utterance into a set of contiguous sequence of n words. Thus, allowing the obtained sequence to be compared with other n-gram training sequences using efficient algorithms. These DA tagging schemes are mostly developed for corpus related to English language because English has a fixed syntactic structure. For English, in both training and test data, the position of the words in an utterance will remain mostly the same. By comparing the n-grams obtained from training data with the n-grams obtained from test data, the best tag for the test utterance is obtained. As this method gives high accuracy for English, the same method is applied to the Telugu ASKLIB corpus.

For choosing the best tag, Naive Bayesian interpretation is used,

$$\hat{T} = \underset{T}{\operatorname{argmax}} \frac{P(U,T)}{P(U)} \tag{1}$$

where $\hat{T}$ is the correct tag, from the tagset T for the utterance U.

For n-grams, the above equation is modified to

$$\hat{T} = \underset{T}{\operatorname{argmax}} \prod_{i=1}^{N} \frac{P(w_i,T)}{P(w_i)} \tag{2}$$

where w_i represents the n-gram sequence and N is equal to the list of n-grams obtained for the utterance U.

3.2 Combinations of Unigrams and kNN

The n-gram method will have a low accuracy for free word order languages like Telugu. In free word order languages like Telugu, even though the speaker's intent might be the same, the position of the words (the syntactic structure) might change. Hence the n-grams method will not work. So word position independent methods must be considered, One such method is to extract n-grams by considering combinations of unigrams. For example: In the combinations of unigrams for n=2 we get bigrams. Here, each word will appear with all the other words in the given utterance and with itself. The problem with this approach is that the time complexity increases when n value increases. As low order n-grams will capture less context, for further processing Memory Based Learning(MBL) method such as k Nearest Neighbors(kNN) (Cover and Hart, 1967) is applied. In MBL the pattern of the training data will be tested against the pattern

of the test data with the word sequence independence as one of the criteria. This method gives the tag of the nearest training utterance to the test utterance without considering the position of words in an utterance in both the training data and the test data.

In combinations of unigrams and kNN method, for choosing the best tag, Naive Bayesian interpretaion in combination with kNN is used.

The above equation 2 will undergo a small modification by considering combinations of unigrams.

$$\hat{T} = \underset{T}{\operatorname{argmax}} \prod_{i=1}^{N_{all}} \frac{P(w_i, T)}{P(w_i)} \tag{3}$$

where N_{all} is equal to the list of combinations of unigrams

The kNN method is

$$DA(U_{test}) = DA(U_{train})$$
$$if : |U_{test} - U_{train_i}| = min\ |U_{test} - U_{train_N}|$$
for N = 1,2,...(no of unique train utterances) (4)

where U_{test}, U_{train} represents train and test utterances respectively

3.3 N-grams related with Karaka Dependency relations with Language Modeling and kNN

In Paninian framework (Bharati et al., 1995), for free word order Indian languages like Telugu, it is proven that the karaka based dependency relations will remain the same even though the syntactic structure of the sentence changes. By using these karaka dependencies, the syntactico-semantic relationships between the words is captured in the *modifier-karaka-modified* format. On careful inspection, this format seems similar to the n-grams with karaka relationships between them. After extracting all the n-grams with karaka dependencies, language modeling technique i.e. Katz's back-off model (at n-gram level) and memory based learning technique i.e. kNN (at utterance level) will be applied. The combination of the above two will give the best tag to test utterances when compared to the above models.

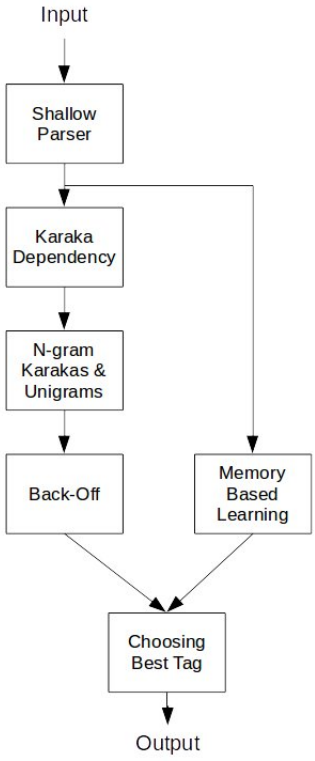

Figure 1: Figure showing the Karaka Dependency method with Back-off and Memory Based Learning for DA tagging.

Why Katz's back-off model? why not just check for karaka dependencies with smoothing algorithms and tag it?

Linguistically speaking, for an utterance to be given a specific tag, each and every word in the utterance must contribute to the tag. So in the test utterance, after extracting all the words with karaka relations between them, there will be certain words whose dependencies are missed. It might be due to any of the following reasons given below.

1. Some karaka dependencies are currently not annotated in the Telugu tree bank.

2. The n-grams have a different karaka dependency between them.

3. Telugu is a morphologically rich language. In n-gram karakas, the root word and the karaka dependency might be the same but the suffix might change.

4. The words themselves are not present in the training data.

Telugu utterance WX-format	VixyArWi : nenu I puswakaM wIsukuMtAnu
Gloss Translation	Student : I this book take+future Student : I will take this book
Karaka Dependencies extracted in modifier-karaka-modified format	[WisukuMtAnu]-k1-[nenu],[WisukuMtAnu]-k2-[I puswakaM]

Table 5: Example of n-gram karaka format extraction for an utterance

Hence, for those n-gram with karaka relationships the karaka dependencies form the test data are dropped. It is verified that they are mostly unigrams. Hence back-off to unigrams is considered. One of the basic back-off techniques is Katz's back-off model, which is presently proven as an effective LM algorithm for the given training and test data.

The algorithm is:

1. Extracting n-grams with karaka dependencies:
   (a) The training data is run through the shallow parser tool (`http://ltrc.iiit.ac.in/analyzer/telugu/`) for clustering the words and morph related information (PVS and Karthik, 2007).
   (b) Telugu tree bank (which consists of huge data containing karaka dependencies) is used as annotated data for karaka dependencies.
   (c) The karaka dependencies are extracted for the words present in each utterance (for both training and test data). An example of karaka dependencies is shown in Figure 2
   (d) The karaka dependencies between the words or word clusters will be converted to *modifier-karaka-modified* format as shown in table 5.
   (e) By observation, the format will be similar to n-grams with just karakas present between them. They are abbreviated as n-gram karakas.

2. Language Modeling technique:
   (a) After extracting n-grams with karaka dependencies for all the training and test utterances, Katz's back-off model (Katz's back-off model, 2015) is applied.

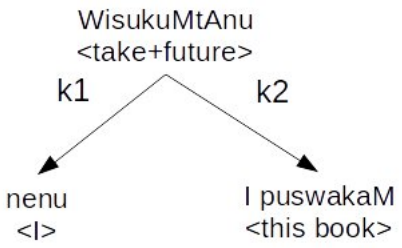

Figure 2: Showing karaka dependencies for an example utterance given in table 5

   (b) In Katz's back-off model, it is verified whether n-gram karakas are present or not in the training data, if present then tag probabilities are updated
   (c) If not, as Telugu is a morphologically rich language, we back-off to morphed n-gram karakas.
   (d) If the morphed n-gram karakas are not present in the training data then, it is known that particular dependencies are not annotated in the training data
   (e) Then the n-gram karakas will be decomposed to non karaka dependency words i.e. we back-off to unigrams and further to unigram morphs.
   (f) When all the above steps fail, smoothing method is considered.

3. k Nearest Neighbor:
   (a) To capture the utterance level information and to provide a strong ground for the respective tag, kNN is used.
   (b) kNN technique applied is same as given in equation 4

Katz's back-off model for n-gram karakas with back-off to morphed n-gram karakas is given in equation 5.

Katz's back-off model for non dependencies i.e. unigrams with back-off to morphed unigrams is given in equation 6.

$$p_{bo}(n\_gram\_karaka_i, T) = \begin{cases} discount_1 \ \frac{C(n\_gram\_karaka_i, T)}{C(n\_gram\_karaka_i)} \\ if \ C(n\_gram\_karaka_i, T) > 0 \\ \\ \alpha_1 \ \frac{C(morph\_n\_gram\_karaka_i, T)}{C(morph\_n\_gram\_karaka_i)} \\ if \ C(morph\_n\_gram\_karaka_i, T) > 0 \\ (otherwise) \end{cases} \quad (5)$$

$$p_{bo}(unigram_i, T) = \begin{cases} discount_2 \ \frac{C(unigram_i, T)}{C(unigram_i)} \\ if \ C(unigram_i, T) > 0 \\ \\ \alpha_2 \ \frac{C(morph\_unigram_i, T)}{C(morph\_unigram_i)} \\ if \ C(morph\_unigram_i, T) > 0 \\ (otherwise) \end{cases} \quad (6)$$

where $discount_1, discount_2$ are discounts obtained by Good Turing estimation as C^*/C and α_1, α_2 are back-off weights

4 Experiments and Results

The testing is done on our ASKLIB corpus using all the three algorithms. The corpus consisting of 225 dialogs is divided into four parts, each time one part is used for the testing and the remaining are combined for the training.

Firstly, the experiment is run on an n-gram method with Bayesian interpretation where n-grams of length 1-3 are considered. The results are shown in table 6

n-grams	**Accuracy**
n = 1	58.17%
n = 2	54.71%
n = 3	47.79%

Table 6: Accuracy obtained for n-grams.

From the results in table 6, it is clear that the n-grams are not sufficient for DA tagging for free word order languages like Telugu. When the n-gram length increases, there is a decrease in accuracy. The explanation is that, as Telugu is a free word order language, even though the intent is the same, the position of words will not be the same in both the training data and the test data. Hence this method will not work for free word order languages like Telugu.

Next, n-grams with position independence are considered by taking combinations of unigrams from the training data and the test data combined with the kNN algorithm. The results are shown in table 7

Combinations of unigrams with kNN	**Accuracy**
n = 1 + kNN	66.67%
n = 2 + kNN	71.80%

Table 7: Accuracy obtained for combinations of unigrams.

From the results in table 7, it can be seen that, combinations of unigrams combined with kNN does show a good response in accuracy. The problem with this method is that, for an utterance of n words, each word is repeated n times which results in an increase in time complexity.

Now consider n-grams with karaka dependencies using back-off and kNN. This method is applied on ASKLIB corpus. The results are as shown in table 8

n-gram karakas using back-off and kNN	**Accuracy**
n-gram karakas + back-off + kNN	73.34%

Table 8: Accuracy obtained for n-gram karakas using back-off and kNN.

For free word order languages like Telugu, it is known that the karaka dependencies remain the same even though the word order changes. Hence there is no need to consider the methods such as combinations of unigrams. From this, there will be no problem of time complexity. There will be an advantage of morphs during back-off. Also, utterance level contextual information is captured using kNN. Due to the above reasons, from the results in table 8 we can see that the accuracy has risen to 73.34%.

When karaka based dependency method is

compared with the position specific n-grams methods and also combinations of unigrams, we can surely see the increase in accuracy, which proves that our method performs better.

5 Conclusion

Various classification algorithms are considered to tag the utterance using DA tagging scheme for ASKLIB corpus. Out of them, the proposed novel method obtained by considering the n-grams karaks with the Language Modeling technique (back-off) at intra utterance level in combination with Memory Based Learning such as k Nearest Neighbor method at the utterance level is applied. This method for DA tagging provided an accuracy of 73.34% when compared to the other methods. The results given in table 8 prove that this method performs better when compared to the other methodologies and is best suited for the DA tagging in Telugu task oriented dialogs.

6 Future Work

The given method will be tested on several Dravidian Languages like Kannada, Malayalam and Tamil etc. ASKLIB corpus is currently being developed in Kannada, Malayalam and Tamil. Further, new algorithms will also be applied taking the concept of karaka dependencies with contextual handling of previous utterances in dialogs as well.

References

Austin, J. L. 1975. *How to do things with words* volume 367. Oxford university press.

Bharati, A., Chaitanya, V., Sangal, R., & Ramakrishnamacharyulu, K. V. 1995. *Natural language processing: a Paninian perspective*, 67–71. New Delhi: Prentice-Hall of India.

Core, M. G., & Allen, J. 1997. Coding Dialogs with the DAMSL Annotation Scheme. *AAAI Fall Symposium on Communicative Action in Humans and Machines* 28–35

Cover, T. M., & Hart, P. E. 1967. Nearest neighbor pattern classification. *Information Theory, IEEE Transactions on*, 13(1):21–27.

Garner, P. N., Browning, S. R., Moore, R. K., & Russell, M. J. 1996. A theory of word frequencies and its application to dialogue move recognition. *In Spoken Language, 1996. ICSLP 96. Proceedings., Fourth International Conference on*, volume 3, 1880–1883. IEEE.

Katz's back-off model. 2015. *In Wikipedia, The Free Encyclopedia.* `https://en.wikipedia.org/w/index.php?title=Katz\%27s_back-off_model&oldid=647273356`

Klüwer, T., Uszkoreit, H., & Xu, F 2010. Using syntactic and semantic based relations for dialogue act recognition. *In Proceedings of the 23rd International Conference on Computational Linguistics : Posters*, pp. 570-578 Association for Computational Linguistics.

Král, P., & Cerisara, C. 2012. Dialogue act recognition approaches. *Computing and Informatics*, 29(2):227–250.

Liu, P., Hu, Q., Dang, J., Jin, D., & Cao, J. 2013. Dialog Act classification in Chinese spoken language. *In Machine Learning and Cybernetics (ICMLC), 2013 International Conference on*, volume 2, 516–521 IEEE.

PVS, A., & Karthik, G. 2007. Part-of-speech tagging and chunking using conditional random fields and transformation based learning. *Shallow Parsing for South Asian Languages, 21.*

Reithinger, N. & Klesen, M. 1997. Dialogue act classification using language models. G. Kokkinakis, N. Fakotakis & E. Dermatas (eds.), *EUROSPEECH*,:ISCA

Rotaru, M. 2002. Dialog act tagging using memory-based learning. *Term project, University of Pittsburgh*, 255–276

Stolcke, A., Ries, K., Coccaro, N., Shriberg, E., Bates, R. A., Jurafsky, D., Taylor, P., Martin, R., Ess-Dykema, C. V. & Meteer, M. 2000.perform Dialog Act Modeling for Automatic Tagging and Recognition of Conversational Speech. *Computational Linguistics*, 26, 339–373

Webb, N., Hepple, M., & Wilks, Y. 2005. Dialogue act classification based on intra-utterance features. In *Proceedings of the AAAI Workshop on Spoken Language Understanding*