MORPHOLOGICAL ANALYSIS OF TELUGU LANGUAGE USING APERTIUM

Dr. Siddhartha Ghosh[*1], Kalyani U.R.S[2] and Sujata M.Thamke[3]

*[1]Head of the Department of Computer Science & Engineering, KMIT, Hyderabad, Telangana, India
siddhartha@kmit.in[1]
[2]R&D Staff of Information Technology, KMIT, Hyderabad, Telangana, India
upadhyayula.kalyani@gmail.com[2]
[3]R&D Staff of Computer Science & Engineering, KMIT, Hyderabad, Telangana, India
sujata.thamke@gmail.com[3]

Abstract: In Natural Language Processing a concept called Machine Translation which is used to perform translation between Natural Languages. As we know that Dravidian Languages are mostly spoken in India, hence our research contribution mainly focuses on one of the Dravidian Languages (i.e. Telugu language) using the concept built on a long term use and understanding of a Rule-based open source Machine Translator called Apertium. Our work is carried out till second module within the eight modules of Apertium Machine Translator i.e. De-formatter and Morphological Analysers module. The text input is given to first module and generated output which will be used as an input to the second module and so on till the last module to get the expected translation of the language, using Apertium in reference with Telugu language, which is well spoken in the two southern states of India i.e. Telangana and Andhra Pradesh.

INTRODUCTION

This paper describes the implementation of Dravidian Language in Apertium. Apertium is a free open source shallow transfer Machine Translation system.

Machine Translation

Machine Translation (MT) is a computational linguistic that uses the software to translate natural languages speech and text from one to another.

Dravidian languages

In India Dravidian Languages are native. Dravidian family contains approximately 85 languages. Dravidian languages are mainly spoken by 215 million peoples in Southern India. Most of the speakers are Telugu, Tamil, Malayalam and Kannada which are recognized by Indian constituency. Dravidian languages are further separated into South, South Central, and Central north groups, and further these groups are divided into 24 subgroups. Only 2 Dravidian languages are exclusively spoken outside India.

1) Telugu Language

In India, Telugu language is one of the Dravidian languages which are spoken by 79 million speakers in 2013. In India Telugu language stands at 3rd position which is been spoken by large number of native speakers.

Machine Translation for Dravidian Language

Machine Translation is a computational linguistic which uses the software to translate text or speech into human natural language. In this paper we are dealing with the Dravidian Language-Telugu. Machine Translation for Dravidian languages requires a Telugu dictionary along with its parts of speech. So that we can perform Machine Translation for Telugu language by using the Dictionary.

METHODOLOGY

To apply Apertium Machine Translation for translating Dravidian Languages, mainly for Telugu and to make a study report on the capacity of the Apertium Translator. Also to work with at least two modules of Apertium to see how it works.

Earlier Work

In the earlier work we have seen the improper translation in the Dravidian language in Babel fish translator and Google translator. To overcome these problems we have applied direct translation and also we have developed a Telugu to Marathi and vice versa language translator in our research lab Dr. Siddhartha Ghosh (2014).

History of Apertium

Apertium concept came into existence in April 2004. Mikel L.Forcada had sent e-mail message to most of the research groups in Spain on the possibility of influence were government agencies involved. So the government has given funds to build a free open source machine translation system for Spain. Soon after that in July 2004 Spanish Ministry of Industry, Tourism and Commerce had announced to fund consortia to develop linguistic technology for the languages of Spain. Few of groups that answered to the e-mail mentioned by Mikel L.Forcada got funding for the development of 2 different Machine Translation Systems they are Matxin and Apertium. Apertium MT engine and tools are not developed from scratch, where as the result and extension of existing

Machine Translation system has been rewritten and developed at Universitat d'Alacant named as interNOSTRUM by Transducens group. Apertium 1 was designed with the Roman languages of Spain, Apertium 2 added support for less-related languages like Catalan to English and Apertium 3 added Unicode support. Apertium is a rule based and shallow transfer machine translation platform. It is free open source software released under GNU General Public License. We should have following requirements

- Install Ubuntu 14.04 LT Operating System
- Install Apertium
- Install SVN
- Develop a Telugu Dictionary

Shallow Transfer Machine Translation

Shallow Transfer is a deep linguistic process which concerns with the use of computers for parsing and generating grammar. As the grammar is manually developed and maintained but it is expensive to run through computers. In recent years machine translation approaches have been basically altered in the case of natural language processing. Apertium is a Shallow transfer machine translation system which is initially designed for translation between associated language pairs, which follows structural transfer rules. Even though some of the components have been used in the deep-transfer architecture that has been developed. The Apertium Machine Translation engine consists of 8-module assembly line, which has been represented in the Figure 1.

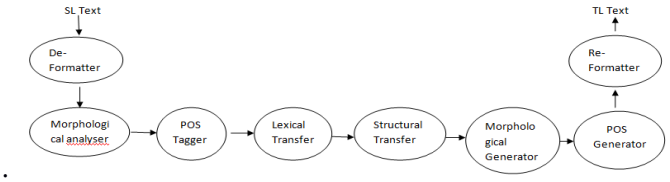


Figure 1. Modules of Apertium.

For diagnosis each and every module can communicate with each other using a text. In this way input and output of the modules can be checked out at any moment due to which the error while translation under process can be easily recognized. As Apertium is an open source so it provides the facility to modification with the existing language pair which are comparatively same and include the language pair which we want to develop.

1) De-formatter

De-formatter module is the first module in Apertium Machine Translator. De-formatter is used to separate the text to be translated from the format information. Format information is encapsulated so that the rest of the modules treat it as a blank between words. In this the input is passed through de-formatter module and the produced output by the de-formatter module is taken as input for the morphological analyser module to produce the output.

```
kmit@kmit:~/Desktop/te-madictionary$ echo "విరాట్ కోహ్లి నేతృత్వంలోని భారత క్రికెట్
బృందం ఆసీస్ పర్యటనకు బయల్దేరింది. నాలుగు టెస్టు మ్యాచ్ ల సిరీస్ లో భాగంగా 18 సభ్యుల
టీమిండియా శనివారం ఉదయం ఆసీస్ పర్యటనకు పయనమైంది. డిసెంబర్ 4 వ తేదీన బ్రిస్బేన్ లో తొలి
టెస్టు జరుగనుంది. మహేంద్ర సింగ్ ధోనికి కుడి చేతి బొటన వ్రేలికి గాయం కావడంతో అతను తొలి టెస్టుకు
అందుబాటులో ఉండటం లేదు. దీంతో మొదటి టెస్టు బాధ్యతలను విరాట్ కోహ్లి తీసుకోనున్నాడు.
తొలిటెస్టు ధోని దూరం కావడంతో నమాన్ ఓజాకు స్థానం కల్పించారు.కారు" | apertium-destxt
విరాట్ కోహ్లి నేతృత్వంలోని భారత క్రికెట్ బృందం ఆసీస్ పర్యటనకు బయల్దేరింది. నాలుగు టెస్టు మ్యాచ్
ల సిరీస్ లో భాగంగా 18 సభ్యుల టీమిండియా శనివారం ఉదయం ఆసీస్ పర్యటనకు పయనమైంది.
డిసెంబర్ 4 వ తేదీన బ్రిస్బేన్ లో తొలి టెస్టు జరుగనుంది. మహేంద్ర సింగ్ ధోనికి కుడి చేతి బొటన వ్రేలికి
గాయం కావడంతో అతను తొలి టెస్టుకు అందుబాటులో ఉండటం లేదు. దీంతో మొదటి టెస్టు బాధ్యతలను
విరాట్ కోహ్లి తీసుకోనున్నాడు. తొలిటెస్టు ధోని దూరం కావడంతో నమాన్ ఓజాకు స్థానం కల్పించారు.కారు.[]
[
]kmit@kmit:~/Desktop/te-madictionary$
```

Figure 2. De-formatter for Telugu text.

2) Morphological Analyser

Morphological analyser is generated using monolingual dictionary.

In this morphological analyser takes the De-formatter module output as input to its module. Tokenization of text into surface forms is defined as morphological analyser. Every surface form will have exactly one or more than one lexical forms which contains lemma. Tokenization of surface forms is not in straight forward due to existence. The morphological analysers will analysis the complex surface forms and will treat them, so that the complex surface forms will be processed in the next modules. In the case of shrinking, the system reads a single surface form and gives output as a sequence of two or more lexical forms. Single Lexical forms are defined as one or more word made by lexical unit.

```
kmit@kmit:~/Desktop/te-madictionary$ echo "కారులు" | apertium-destxt | lt-proc
te-ma.automorf.bin
^కారులు/కారు<vblex><pl>$.[][
]kmit@kmit:~/Desktop/te-madictionary$
```

Figure 3. Morphological Analyser

In the above mentioned example the word కారులు is a surface form which is having the word lemma కారు. To perform

morphological analyser module we need to generate automorf.bin file. Generating files through terminal in Ubuntu

Lt-comp lr telugu.dix automorf.bin

The above mentioned command is used to create automorf.bin file. In this command lt-comp is used to compile the dictionary, lr means from left to right where telugu.dix is the dictionary which we have created in xml format.

Apertium for Telugu

In Apertium there is no machine translation available for Telugu language. So our research is about implementation of Telugu language in Apertium Machine Translator, which discuss about De-formatter and morphological analyser which is first and second module out of eight modules to perform language pair translations in Apertium.

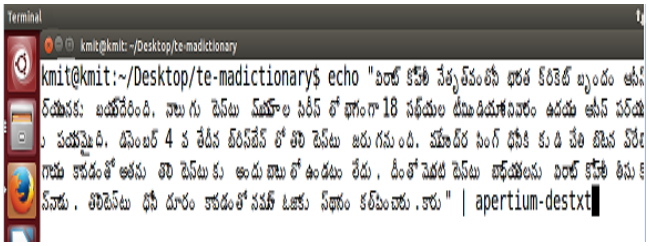
1) Dictionary Creation

Dictionary is defined as set of words listed in the alphabetical order with the one or more meaning and parts of speech of the words and also in dictionary they will describe the usage of the word by giving an example. Dictionary creation plays an important role to produce output for morphological analyzer module. The format of the dictionary should follow xml format. Here is the sample format of the Dictionary. Here is the sample format of the Dictionary.

Screenshot 1. Dictionary

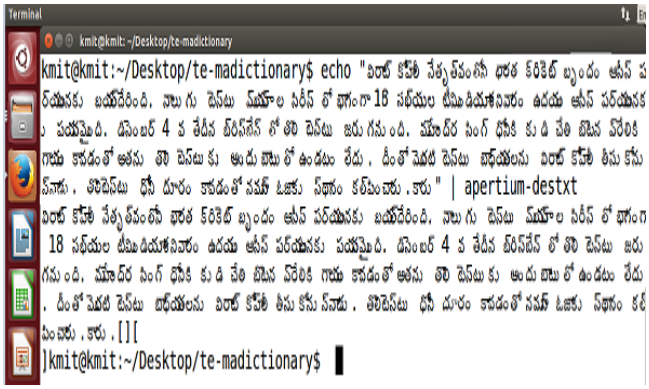
RESULTS AND DISCUSSION

It is preferred to discuss all the results in detail in case of original research paper. To explain observed data you can use figures, graphs and tables. After creating the above dictionary we have given the Telugu language text input command in our terminal where for particular de-formatter we Apertium accept the text followed by the pipe and apertium-destxt keyword for generating the de-formatter output. Below Screenshot.2. Show the how to give input for Telugu Language.



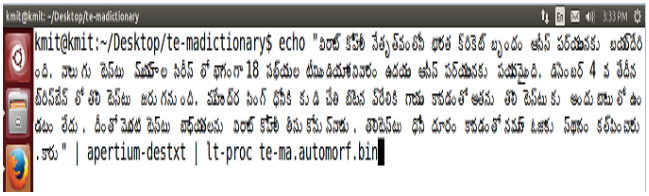
Screenshot 2. De-formatter input

Once input is passed the command will help to generate the de-formatter output which is clearly shown in the Screenshot.3.



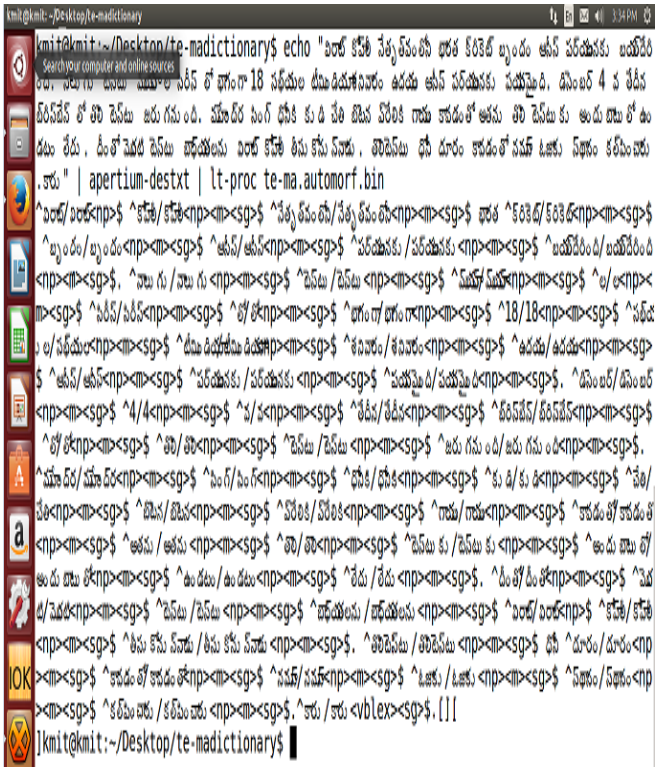
Screenshot 3. De-formatter output

In the below Screenshot. 4. We have passed the input for the Morphological Analyser where the de-formatter output in form of apertium-destxt and language.automorf.bin file is passed with the input command for this module to generate the output for Morphological Analyser. Without producing automorf.bin file and de-formatter module output we will be not able to get the required output for Morphological module.



Screenshot 4. Morphological Analyser input

Screenshot.5 is the output generated for the paragraph which we have passed as an input for Morphological Analyser module where the paragraph is divided into words along with its parts of speech which have been defined in the dictionary that show whether the word is



Screenshot 5. Morphological Analyser output

In our survey we have gone through different Machine Translator which have some or the other drawbacks where in our previous paper i.e. “Translation of Telugu-Marathi and vice-versa using Rule based Machine Translation” we have tried to overcome the problem regarding proverbs etc. Studying further we have realized that Apertium a Machine Translator is one of the open sources where we need to work with modules. It shows clear and fine ways of working with any machine translator, and emphasis how each and every module gives the output and how step by step we are able to generate the proper translation of the language. So we have tried to work with one of the module that is Morphological Analyser which is the second module.

CONCLUSION

The overall conclusion of this paper is working with the Dravidian language on the Apertium machine translator which don’t have translator for Telugu language, so here we have developed a Telugu language dictionary i.e. morphological dictionary which help to perform operation for first two module of Apertium Machine Translator i.e. De-formatter and Morphological Analyser module where we have to pass the Telugu language text as input to de-formatter module and the output which is generated has given that as an input to the next module i.e. Morphological Analyser. Therefore we have successfully run the two modules. Also the work can be extended by working on rest of the modules.

ACKNOWLEDGMENT

We are thankful to the IRC Apertium channel Committee for guiding us for our work and also the below mentioned references which helped us a lot to carry out of research successfully.

REFERENCES

[1] Uma Maheshwar Rao, G. “Morphological Analyzer for Telugu”.(electronic form) Hyderabad: University of HyderabaD, 1999.

[2] Uma Maheshwar Rao, G. Amba Kulkarni, P “Computer Applications in Indian Languages”, Hyderabad: The centre for distance education, University of Hyderabad, 2006.

[3] Uma Maheshwar Rao, G. Amba Kulkarni, P. and Christopher M. “Morphological Analyzer and Its Functional Specifications for IL-ILMT System”. CALTS, Hyderabad: University of Hyderabad, 2007.

[4] Uma Maheshwar Rao, G. and Parameshwari, K. “On the Description of Morphological Data for Morphological Analysers and Generators: A case of Telugu, Tamil and Kannada”. Mona Parekh (ed.) in Morphological Analysers and Generators, pp73-81.Mysore:LDCIL,CIIL.2010.

[5] Uma Maheshwar Rao, G. and Christopher, M. “Word Synthesizer Engine. Mona Parekh (ed.) in Morphological Analysers and Generators”, pp73-81. Mysore: LDCIL,CIIL 2010.

A., Sarasola K., M. Forcada, S. Ortiz, L. Padró. “An Open Architecture for Transfer-based Machine Translation between Spanish and Basque”. IXA Taldea, Informatika Fakultatea, Euskal Herriko Unibertsitatea, E-20071 Donostia 2Transducens group, Departament de Llenguatges i Sistemes Informàtics Universitat d'Alacant, E-03071 Alacant 3TALP group, Departament de Llenguatges Sistemes Informàtics Universitat Politècnica de Catalunya2005.

[6] Armentano-Oller, C., Carrasco, R. C., Corb´ı-Bellot, A. M., Forcada, M. L., Ginest´ı-Rosell, M., Ortiz-Rojas, S., Perez-Ortiz, J. A., ´Ram´ırez-Sanchez, G., S ´ anchez-Mart ´ ´ınez, F.,and Scalco, M. A. “Open-source Portuguese–Spanish machine translation”. In Computational Processing of the Portuguese Language, Proceedings of PROPOR 2006, volume 3960 of LNCS, pages 50–59.

[7] Canals-Marote.R., Esteve-Guillen, A., GarridoAlenda,A., Guardiola-Savall, M., IturraspeBellver, A., Montserrat-Buendia, S., OrtizRojas, S., Pastor-Pina, H., Perez-Anton, P., and Forcada, M. (2001). “The Spanish–Catalan machine translation system interNOSTRUM”. InProc. of MT Summit VIII. Santiago de Compostela, Spain, 18–22 July2001.

[8] Carme Armentano-Oller, C., Corbi-Bellot, A. M.,Forcada, M. L., Ginest´ı-Rosell, M., Bonev,B., Ortiz-Rojas, S., Perez-Ortiz, J. A., ´Ram´ırez-Sanchez, G., and S ´ anchez-Mart ´ ´ınez,F "An open-source shallow-transfermachine translation toolbox: consequences of its release and availability". In OSMaTran, Aworkshop at MT Summit X, pages 23–30. Carreras, X., Chao, I., Padro, L., and Padro, MT Summit X, pages 23–30, 2005.

[9] Corbi-Bellot, A.M., Mikel L. Forcada, Sergio Ortiz-Rojas, Juan Antonio Perez-Ortiz, Gema Ramirez-Sanchez, Felipe Sanchez-Martinez, Inaki Algeria, Aingeru Mayor, Kepa Sarasola (2005) "Open-Source Shallow-Transfer Machine Translation Engine for the Romance Languages of Spain", Proceedings of EAMT 2005 (Budapest, 30-31 May 2005).

[10] Dr.Siddharatha Ghosh, Sujata Thamke, Kalyani U.R.S, "Translation of Telugu-Marathi and vice-versa using rule based machine Translation", Fourth International Conference on Advances in Computing and Information Technology (ACITY 2014) Delhi, India - May 2014.

[11] "Francis Morton Tyers Apertium Tutorial ", 17th July 2013.

[12] Jimmy O'Regan, "Apertium: Open source machine translation", Published in Issue 152 of Linux Gazette, July 2008.

[13] Mikel L. Forcada, Boyan Ivanov Bonev, Sergio Ortiz Rojas, Juan Antonio Perez Ortiz, Gema Ramirez Sanchez, Felipe Sanchez Martınez, Carme Armentano-Oller, Marco A. Montava, Francis M. Tyers, "Documentation of the Open-Source Shallow-Transfer Machine Translation Platform Apertium, Department de Languages i Sistemes Informatics Universitat d'Alacant" March 10, 2010.