Developing A Heterogeneous Contextual Corpus Rater For Odia Language

Rudranarayan Mohapatra
P.G. Department of Odia, Utkal University, Vanivihar, Bhubaneswar, Odisha, India

Abstract: It is a common knowledge that having access to an error free corpus is of great benefit for developers, learners, instructors and users. Each corpus provided a unique perspective. When developing the Corpus Rater for Odia language content having heterogeneous in nature, certain linguistic issues like current theories of language, language learning and good practice in assessment are taken into consideration. The paper describes self-made classification of Odia language errors types using Odia language corpus data. The paper also presents and elaborates the developing a corpus rater in principle of zero quality control using ‘Process writing’ behavior in order to reach its perfection. The knowledge can be a great use in assessing soundness of our error annotation.

Keywords: Heterogeneous Corpus Rater (HCR), Intention Counter, Error Mode and Impact Analysis Model, Error Tree Builder, Bootstrapping Algorithm

I. INTRODUCTION

The cost of ignoring Official Communication has become substantial as the proliferation of electronic information amidst an increasingly regulatory environment has made it prohibitively expensive to take a reactive approach to the issue very specific to Indian Languages. The dramatic growth in the use of electronic information within Official Communication and expectation of standards in business practice requires new technology solutions that can fully address governance in an automated fashion in Indian Languages.

Looking this aim to build a surveillance tool for Official languages domain (Odia language) for automatically editing and rating the official communication documents in Indian languages and would help to try to fix the improvised suggestion not only from morphological or syntax direction but also from the direction of contextual semantic nature looking the whole text subjective mood and expectation in a multilingual or cross lingual environment to cater the client interest in desired languages.

II. CORPUS RATER & ITS FUNCTION

The Corpus Rater system has two major features. Firstly, the texts known as warrant are to be analyzed by our pre-machine learning library to identify the features that make them distinctive. Secondly, machine-learning strategies will be used to analyze the texts to derive other contextual features that may be useful in classification and would try to automatically fixing the probable improvements with an expected rating.

This task will comprise of major technologies like Internet document classification system query based analyzer and Machine Translation system.

The Corpus Rater can:
- Detect the text languages, domains & domain names
- Make Automatic Formatting with the given setting parameters.
- Find those pesky mistakes like (spelling, punctuation and Conjunction markers) and correct them before turning the text.
- Automatic handle the Syntax, morphology the text and makes possible grammatical agreements.
- Automatic alerts to opportunities to improve the writing...
Automatic try to replace the domain specific & contextual vocabulary looking assessing the mode and mood of the written text.

The user can personalize the proofreading process by giving probable real-time committed mistakes if necessary.

Automatically Grade the Corpus, its style and Word Choice Analysis

The system would combine the power of Natural Language Processing (NLP), Artificial Intelligence (AI), machine learning, Information Retrieval (IR), Computational Linguistics, Data Mining, and Advanced Pattern Matching (APM) process in Indian Languages looking to the future huge data management and reducing the task of large human intervention.

So we wanted to create tools that do not muddy the waters in the process of assisting with the craft of writing in Indian languages but to behave as a surveillance servicing tool for large data management in improving the editing quality of Indian language corpus very specific to here is for Odia language.

III. LITERATURE SURVEY

Generally a Corpus rater behaves as an eco-system for Automatic Language production. In the early 1980s, NLP began to facilitate the development of new writing tools as well: Writer's Workbench was designed to help students edit their writing. This software provided automotive feedback mostly related to writing mechanics and grammar (MacDonald, Frase, Gingrich, & Keenan, 1982). Some suggested resources in this direction are MLA Guide, APA Guide etc. And some suggested products are: Grammarian Pro X, After the Deadline, style-check.rb etc. However, looking to the Odia language prospective, a standard Corpus rater tool is a day dream to language market.

IV. APPROACH TO BE FOLLOWED

The Corpus Rater would follow the ‘Process Writing’ behavior for proofing the heterogeneous corpus from the very beginning of language recognition to review, edit and the final proof.

V. ARCHITECTURE OF CORPUS RATER SYSTEM

The Sub-components of the system can be described through (1) LMS Data base Creation, (2) “Error Mode and Impact Analysis Model: EMIA”, (3) Intention Counter Module, (4) Error Tree Builder. The details of system Architecture is illustrated at the end of the paper at Fig. No. 2.

VI. ERROR MODE AND IMPACT ANALYSIS MODEL (EMIA)

The GUI of EMIA would be would be a fourfold divisional model. It starts from input and followed by improvement, annotation and then evaluation. The life cycle process of EMIA takes its continuation process of initiation to conclusion till it receives perfection.

A. TYPES OF AUTOMOTIVE EMIA

For a Corpus Rater tool, the EMIA starting from Structural level to Semantic and then conceptual level i.e. pragmatic EMIA should be taken into consideration. Where lexical, morphological and syntactic behavior are falls under Structural EMIA, the Pragmatic part is the part of Conceptual EMIA. The semantic part of corpus is the bridge between Structural and conceptual EMIA both.
B. PRINCIPLES OF AUTOMOTIVE EMIA

The principles of Automotive EMIA are Zero Quality Control. The cause of Errors in a text are Forgetfulness, Errors due to misunderstanding, Errors in identification, Errors made by amateurs, Willful errors, Inadvertent errors. Errors due to slowness, Errors due to lack of standards, Surprise errors, Intentional error.

C. RELATIONSHIP OF AUTOMOTIVE EMIA

There is a great cohesive relationship among the modules of Structural EMIA, Semantic EMIA and Pragmatic EMIA in the foundation of both linguistic theories and technical practice. However we shouldn’t Mix up the Semantic Errors and causes with Pragmatic Errors.

VII. CREATING STRUCTURAL (GRAMMATICAL) EMIA FLOW CHART

A. LEXICAL EMIA

After standard tokenization, each token is passed through the lexical verification process and then inserted into the Lexicon Management System (LMS) which supports automated and manual resolution of unknown tokens. The LMS is a system developed to store the accumulated lexical knowledge and contains categorizations of spelling errors, abbreviations, acronyms and a variety of non-word tokens.

B. MORPHOLOGICAL EMIA AND ITS FUNCTIONALITY

In the field of language, morphology means the structure of words, how words are formed, and how the parts fit together. If you get the wrong morpheme (i.e., word part) in the wrong place at the wrong time, you’ve committed a morphological error. It is the branch of linguistics that deals with the study of the internal structures of words and how new words are created from the existing ones through the use of various morphological processes namely affixation, compounding, conversion, blending, chipping, reduplication etc. (cf. O’Grady and Guzman 1996; Quirk and Greenbaum, 1973).

In this section we present the actual morphological errors in our subjects’ scripts and spoken Odia. The errors are subdivided into: (a) Affixation-related errors; (b) Compound-related errors; and, (c) Conversion-related errors

<table>
<thead>
<tr>
<th>Sl. No.</th>
<th>Error Types</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>(a) Affixation-related errors</td>
<td>Errors arising from the wrong use of prefixes</td>
</tr>
<tr>
<td></td>
<td></td>
<td><em>Aapana jane besadhu</em> (asaadhu) byakti.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Errors arising from making uncountable nouns countable</td>
</tr>
<tr>
<td></td>
<td></td>
<td><em>Samudrara jalagudi luniua.</em> (jala)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Errors arising from omission of suffixes</td>
</tr>
<tr>
<td></td>
<td></td>
<td><em>bikaasa odia byaakarana</em> (bikasita odia byaakarana)</td>
</tr>
<tr>
<td>2.</td>
<td>(b) Compound-related errors</td>
<td>Compounding is a morphological process which consists in the combination of at least</td>
</tr>
</tbody>
</table>
two free morphemes such as book + shop (bookshop).

<table>
<thead>
<tr>
<th>Error Type</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compound errors in the rewriting</td>
<td>&quot;Post Office&quot; instead of &quot;Office&quot; (Post-Office)</td>
</tr>
<tr>
<td>exercise</td>
<td></td>
</tr>
<tr>
<td>3. (d) Conversion-related Errors</td>
<td>Few conversion-related errors in the written responses of the subjects.</td>
</tr>
<tr>
<td></td>
<td>&quot;post office&quot; (instead of &quot;Post Office&quot;)</td>
</tr>
<tr>
<td></td>
<td>mu jibaathun prastuta (mu jibaakur prastuta)</td>
</tr>
</tbody>
</table>

C. SYNTACTIC EMIA AND ITS FUNCTIONALITY

A syntax error is a violation of the syntax, or grammatical rules, of a natural language. However the Syntactic EMIA module will handle both the Syntactic & Non- Syntactic error types.

<table>
<thead>
<tr>
<th>Sl. No.</th>
<th>Error Type</th>
<th>Incorrect Examples</th>
<th>Correct from</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Quotation marks</td>
<td>Incorrect values: &quot;Tanakara kichhi bahigudika achhi.&quot;</td>
<td>Correct values:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;Tanakara kichhi bahi achhi.&quot;</td>
<td></td>
</tr>
<tr>
<td>2.</td>
<td>Date expressions</td>
<td>Incorrect values: &quot;1, January 2017&quot;</td>
<td>Correct values:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;1, 2016&quot;</td>
<td></td>
</tr>
<tr>
<td>3.</td>
<td>Several spaces in a row</td>
<td>Incorrect values: &quot;1, January 2017&quot;</td>
<td>Correct values:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;1, 2016&quot;</td>
<td></td>
</tr>
<tr>
<td>4.</td>
<td>Abbreviation &amp; Acronyms</td>
<td>Incorrect values: &quot;1.3. (ରାଜଧାନୀ କିଛି)&quot;</td>
<td>Correct values:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;ରାଜଧାନୀ କିଛି&quot;</td>
<td></td>
</tr>
<tr>
<td>9.</td>
<td>Parentheses</td>
<td>Incorrect values: &quot;(college (in parenthesis))&quot;</td>
<td>Correct values:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;(college in parenthesis)&quot;</td>
<td></td>
</tr>
<tr>
<td>10.</td>
<td>Measurement Units</td>
<td>Incorrect values: &quot;(college (in parenthesis))&quot;</td>
<td>Correct values:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;(college in parenthesis)&quot;</td>
<td></td>
</tr>
<tr>
<td>11.</td>
<td>Punctuation marks</td>
<td>Incorrect values: &quot;(college (in parenthesis))&quot;</td>
<td>Correct values:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;(college in parenthesis)&quot;</td>
<td></td>
</tr>
</tbody>
</table>

Table 2: Some non-syntactic error type

Table 1: Some frequent Syntactic error types

VIII. SEMANTIC EMIA

To make the semantic EMIA, there is a Semantic lexicon would be prepared by using ‘Bootstrapping Algorithm’ which discovers words with semantic properties similar to a small set of labeled seed examples. The semantic EMIA would be primarily responsible for:

- Structural Ambiguity: Would calculate based upon previous and post text sentences about best and appropriate Syntactic structure out of more than one Structural output.
- Domain Context Ambiguity: Would calculate the contextual stylistic errors and would suggest probable replacements.
IX. CREATING PRAGMATIC EMIA FLOW CHART

Creating a Pragmatic EMIA flowchart is still in our conceptual level and needs more study for its better form. However some major steps of the same are mentioned here for future study and analysis.

- Identify the Pragmatic Type
- Look the Manual Feedback Model
- Agree on Starting and Ending point of the Context
- Agree on level of Details
- Look for Text Areas that to be Improved
- Construct Pragmatic Flowchart
- Analyze the Results

Diagram of Pragmatic EMIA flowchart

- Fully define each branch of the tree before beginning another branch
- During the construction of the tree, it is advisable to use a block diagram of the system to simplify determining the main branches

XI. INTENTION COUNTER MODULE

It would analyze the features like: Forgetfulness, Errors due to misunderstanding, Errors in identification, Errors made by amateurs, Willful errors, Inadvertent errors, Errors due to slowness, Errors due to lack of standards, Surprise errors, Intentional error.

- They are the mental causes of actions, that is, they are what together with some bodily movements constitute an action, as distinct from a mere event.
- They have conditions of consistency. You can desire p and desire not-p at the same time, but you cannot intend p and intend not-p at the same time.
- Their object is presupposed to be attainable by the agent. You can desire to go to the moon this afternoon, but you cannot intend to go to the moon this afternoon (unless you are a multimillionaire who has made an arrangement with some spatial agency).
- Their object represents their conditions of satisfaction.

XII. CONCLUSION

In this paper, we have attempted to identify and classify some of the frequent occurring errors in the Odia language and their probable solution for to develop well managed Odia corpora in an automation process. This paper among other things reveals that the causes of these errors are numerous, and they range from the inconsistency inherent in Odia language itself, overgeneralization of rules, and misapplication of rules interference. Looking to the constraint of paper and our analysis report, we take semantic and pragmatics error rectification part for further study.

REFERENCES


