

# Oriya Morphological Analyzer Using Lttoolbox

Itisree Jena, Himani Chaudhry<sup>(✉)</sup>, and Dipti Misra Sharma

International Institute of Information Technology, Hyderabad, India  
{itisree,himani}@research.iiit.ac.in, dipti@iiit.ac.in

**Abstract.** A Morphological analyzer is an essential tool for many NLP applications. Developing a fully-fledged morphological analyzer (MA) tool for an agglutinative language like Oriya is a challenging task. This paper deals with development of a MA for Oriya, a resource poor language. The MA is being developed using the paradigm approach. It consists of various paradigms under which nouns, pronouns, adjectives, verbs and indeclinables are classified. The paradigms have been created for inflected forms using an XML based morphological dictionary from the Lttoolbox package. At present, a total of 10,840 words have been entered into the dictionary. In the course of the paper we talk about the design and implementation of the MA. We also talk about the issues and limitations experienced in developing it using Lttoolbox.

**Keywords:** Oriya morphological analyzer · Apertium · Lttoolbox · Compound verb · Paradigm approach · Morphological analyzer

## 1 Introduction

Efforts are on for past many years, to develop various of NLP tools such as Morphological Analyzers (MA), Part-of-Speech (POS) taggers, spell checkers and so on, for Indian languages (ILs), to assist tasks such as Machine Translation. These efforts to develop NLP tools for ILs have especially focused on computational morphology, since ILs are morphologically quite rich. Developing an Oriya MA becomes important, to help build these tools for the language. This work presents the design and development of a MA for Oriya.

The official language of the state of Odisha (Orissa), Oriya, now officially pronounced ‘Odia’ belongs to the eastern branch of the Indo Aryan sub family of the Indo-European language. It has the status of the sixth classical language in India. Around 31 million people are using this language. “Oriya is a syntactically head-final and morphologically agglutinative language” [15]. Thus, quite some information is contained in morphological structures in Oriya.

The nouns in Oriya are generally characterized by inflectional categories like number, gender, case and also take articles and number classifiers. The definite articles ‘-ti’ and ‘-taa’ occur only with singular nouns. Its plural markers include ‘-maane’, ‘-gudaa’, ‘-gudika’, ‘-gudaaka’. The plural marker ‘-maane’ is only added to animate nouns e.g. ‘pua+maane (son+s)’. It can not be added to either human proper nouns or in-animate nouns. Thus, we can not say ‘kaatha+maane’ (wood+s).

Oriya has natural gender that does not reflect in the agreement with grammatical categories like verbs. For example, *baagha* (tiger) and *baaghuNi* (tigress). We use roman transliteration scheme to represent examples in this paper.

In Oriya, “adjectives which precede the nouns in attributive position do not show any agreement with the nouns except in a few cases where the adjective agrees with the noun in gender” [12]. For example, *kaLaa baLada* (black bull), *kaaLi gaaii* (black cow).

Oriya finite verbs are marked for person, number, tense, aspect and mood. They agree with subject nouns and this is reflected by an agreement marker that manifests attached to the end of the main verb. For example:

```
aame khaa-il-u
we eat-pst-agr.
‘We ate’
```

## 1.1 Related Work

Various methods have been adopted for morphological analysis in natural language processing. Brute Force method, Root Driven approach, Affix Stripping method are some of the methods evolved typically for the analysis of ILs. MAs being developed using the paradigm approach include Hindi MA by Bharati et al. (1995) [3], and the Marathi MA by Bapat et al. (2010) [2], of which, [2] combine a paradigm based inflectional system with finite state machines for modeling the morphotactics. Marathi derivational MA by Vaidya et al. (2009) [16], Tamil MA by Parameshwari (2011) [13] and Benagli by Faridee et al. (2009) [7], all adopt paradigm approach using Lttoolbox to develop their MA, which is similar to our work, discussed later in this paper.

Further, Oriya MA have been developed by Shabadi (2003) [15], Sahoo (2003) [14] using deterministic Finite State Automata (FSA), where the FSA recognize if the input string of morphemes is an appropriate Oriya word or not. They do this by plugging each forms into the FSA, using two level morphology. The work propose a model which can provide lexical, morphological and syntactic information for each lexical unit in the analyzed word form. The second approach followed for Oriya is our work using Lttoolbox from the Apertium toolkit which we have reported in Jena et al. (2011) [8].

## 2 Current Work

### 2.1 Approach

We have adopted the paradigm based approach to create a MA for Oriya. Paradigms are employed to represent the inflectional regularities of lexical units in a language [6]. A paradigm is a set of related word forms which follow the same set of spelling rules and take the same kind of affixes. “Paradigm approach is well suited for agglutinative language nature” [1]. Oriya being an agglutinative language, the paradigm approach seems to work well for it.

## 2.2 Resources Used

**Lexical Resources.** The foremost requisite for a MA is a root word dictionary. But, Oriya being a resource poor language, an online root word dictionary wasn't available for it. We, thus, manually created the dictionary using the following resources:

- 'Taruna Sabdakosha' [9] an Oriya dictionary.
- 'A synchronic grammar of Oriya' [12].
- A corpus of 2,720,400 words from Central Institute of Indian Languages, Mysore (CIIL) - Our major resource for the database of the dictionary and also for the training and testing data for our MA.

The root word dictionary was created using the lexical resources mentioned above. Initially, we used a frequency based list from the CIIL Oriya corpus and added root words to it from the 'Taruna Sabdakosha', to enhance it. Currently the dictionary contains 10,840 root words, details of which are:

- |                     |                        |
|---------------------|------------------------|
| 1. Nouns - 5,031    | 6. Postpositions - 40  |
| 2. Pronouns - 18    | 7. Conjunctions - 36   |
| 3. Adjectives - 930 | 8. Clitics - 15        |
| 4. Verbs - 4,537    | 9. Particles - 14      |
| 5. Adverbs - 179    | 10. Interjections - 40 |

**Tool.** We used Lttoolbox [11] package from the Apertium [6] toolkit to develop the Oriya MA. The Lttoolbox is a well known NLP tool used to build tools like morphological analyzer and morphological generator. It is a free software and released under the terms of the GNU General Public License. It uses an XML based format to represent linguistic data. Paradigms are created inside it using some of the elements in its morphological dictionary. Further, a morphological dictionary can be used for both, a morphological analyzer and a morphological generator, depending on the direction in which it is read by the system.

## 2.3 Data Development for Oriya Morph Analyzer

**Oriya Morphological Dictionary in Lttoolbox.** The Oriya morphological dictionary consists of declension or conjugation patterns of words in XML format used in Lttoolbox. The dictionary has four sections, of these the two main sections are paradigm definition section and dictionary section. Alphabet and symbol definition sections being the other two sections.

Declension or conjugation used, are based on parameters such as, gender, number, person, case, vibhakti (case marker) for nouns and pronouns. Gender, number, person, suffix string taken as TAM (tense, aspect and modality) for verbs. [3]

**Classification of Paradigms.** Paradigms have been created for the open class categories like nouns, verbs and adjectives and later on, closed class categories like postpositions and conjunctions etc. The words that have identical grammatical information make one paradigm class. However, all words with similar endings/suffixes may not follow the same paradigm. For instance, two verbs ‘khaa’ (eat) and ‘gaa’ (sing) fall in the same paradigm as they take similar inflections. But the verb ‘jaa’ (go) falls in a different paradigm though it has the same ending. This is because the verb ‘jaa’ (go) changes its root form when it takes past tense inflection e.g. ‘jaa’ (go) becomes ‘gali’ (go+past) but in case of verb ‘gaa’ (sing) becomes ‘gaaili’(sing+past). There are some parts of speech like adverbs, conjunctions, postpositions, clitics etc., that remain uninflected, so we have listed them directly in our dictionary. Table 1 shows the paradigm classification for different categories.

**Table 1.** Number of paradigm classes.

S.No	Category	Paradigm classes
1	Noun	14
2	Pronoun	9
3	Adjective	13
4	Verb	13

### 3 Evaluation and Result

We conducted three experiments to evaluate our MA. We discuss this shortly. Since a MA produces more than one answer, we found it more appropriate to carry out a more detailed evaluation of the MA than just evaluating the precision and recall values, since “Precision-Recall gives general overall impression about the performance of a system” [10]. A more detailed evaluation is necessary to know what kind of words are over analyzed, which are under analyzed, and so on. This is discussed in detail, in Subsects. 3.3 and 3.4.

#### 3.1 Evaluation I

Here we focus on the overall coverage of our MA (Table 2). A corpus of 11,368 words (non-unique) was taken (Sect. 2.2) in order to evaluate the overall coverage of the morph in a random test data environment.

It must be noted that the coverage here is based on a small dictionary size of 10,840 root words. The class of recognized words includes the cases where the tool gave an analysis (irrespective whether the analysis was correct, partially correct or wrong). While the class of unrecognized words comprises those cases where the morph analyzer didn’t give an output or analysis.

**Table 2.** Results: the overall coverage.

Total no. of words	11,368
Recognized words	8,303
Unrecognized words	3,065
Coverage	73.03 %

### 3.2 Results and Error Analysis

In Table 2 we see that 3,065 words remained unrecognized by our MA, which forms 26.97 %. These words can be easily accounted for (Table 3 shows the break up of the unanalyzed words). Out of this 26.97 %, out of vocabulary (OOV) words (which include foreign words, proper nouns and numerals) form 29.81 % and noise (meaningless characters/words occurring in the corpus) takes up 6.62 %. The remaining words fall into causative verbs (2.34 %) and ‘others’ (61.20 %). Since causative verbs are currently not being handled, these remain unanalyzed. ‘Others’ in Table 3, are Oriya words that remain unanalyzed because they have yet to be entered in the morphological dictionary. These form a major part of the unrecognized words.

Therefore, the two major categories that affect the coverage of the MA, are OOV & noise (36.43 %) and ‘Others’ (61.20 %) of 3,065 unrecognized words. With a small dictionary size of 10,840 words, the MA’s coverage is 73.03 % and increasing the dictionary size can further improve the coverage.

**Table 3.** Error analysis.

S.No	Unanalyzed words	Occurrences	%
1	Numerals	164	5.35
2	Foreign words	338	11.02
3	Proper nouns	412	13.44
4	Causative verbs	72	2.34
5	Others	1876	61.20
6	Noise	203	6.62
	Total	3065	100 %

### 3.3 Evaluation II

When an MA produces output, it may have 6 possible cases:

1. Type1: correct output, e.g. ABCD/ABCD.
2. Type2: added some wrong output to correct output, e.g. ABCD/ABCDE.
3. Type3: missed some correct output, e.g. ABCD/ABC.
4. Type4: missed some correct output and add some wrong output, e.g. ABCD/ABCE.

5. Type5: all incorrect output, e.g. ABCD/EFG.
6. Type6: no output, ABCD/No Output.

These six cases help us to decide which aspect of morphology needs further attention for improvement. To evaluate an MA, some data manually tagged with morph features (gold-standard data) is needed. It contains all possible analysis of the words. In the above examples ‘ABCD’ is gold standard data and others are machine’s output. To create the gold standard data to evaluate our MA we randomly took 1066 words from the CIIL Oriya corpus. The data was tagged using Sanchay (an open source platform for working on languages, with components like a text editor with customizable support for languages and encodings, annotation interfaces, etc.) annotation interface, in Shakti Standard Format (SSF) (This format is a highly readable representation for storing language analysis [4]). The Apertium produced morphological analysis was compared with the gold standard data.

We compared the machine produced morphological analysis using our gold standard data as the reference data. After we ran our MA on the randomly taken corpus we compared it with the gold standard data. Table 4 shows the results for type wise evaluation of the accuracy against a gold-standard corpus.

**Table 4.** Results: type wise evaluation of the accuracy against a gold-standard corpus.

S.No	Types	Gold/Output	Count	% Count
1	Type:1	ABCD/ABCD	754	70.73
2	Type:2	ABCD/ABCDE	41	3.84
3	Type:3	ABCD/ABC	25	2.34
4	Type:4	ABCD/ABCE	100	9.38
5	Type:5	ABCD/EFG	0	0
6	Type:6	ABCD/No Output	146	13.69
		Total tokens	1066	100 %

In Table 4, **Type:1** gives fully correct output (comprises 70.73 % of total count), whereas **Type:2**, **Type:3** and **Type:4** give partially correct output (comprises 15.56 % of the cases). Further, the coverage of the tool is 86.30 %. As mentioned earlier, **Type:4** consists of some correct output and some wrong output (partially correct output), we notice that **Type:4**–9.38 % has the highest contribution in cases with partially correct output, as compared to the other types with partially correct output (**Type:2**–3.84 % and **Type:3**–2.34 %). **Type:6** includes cases where MA fails to give the output.

### 3.4 Evaluation III

In the third evaluation we focused on the accuracy of only two features–‘root’ and ‘category’ instead of all of the features. This is so because for some applications

only these two features are taken into consideration. Other feature structure values may not be important for them. Thus, through evaluation II the accuracy of the MA for such applications is also reported. Additionally, for evaluation III we took the same data sets that were used in evaluation II. Table 5 shows the results for type wise evaluation of the accuracy for two features.

**Table 5.** Results: Type wise evaluation of the accuracy for ‘root’ and ‘category’.

S.No	Types	Gold/Output	Count	% Count
1	Type:1	ABCD/ABCD	853	80.01
2	Type:2	ABCD/ABCDE	21	1.96
3	Type:3	ABCD/ABC	27	2.53
4	Type:4	ABCD/ABCE	19	1.78
5	Type:5	ABCD/EFG	0	0
6	Type:6	ABCD/No Output	146	13.69
		Total tokens	1066	100 %

We see that the percentage count of **Type:1** increased to 80.01 % in evaluation II, whereas the percentage count of **Type:4** decreased to 1.78 % (dropped by 7.60 %). Thus considering only root and category features shows an overall higher accuracy of the MA. The coverage remains the same for both the evaluations.

## 4 Challenges and Limitations

### 4.1 Foreign Words

As seen in Sect. 3.2, foreign words remain unrecognized, and thus unanalyzed in our MA since they are not part of the data base. Presence of foreign words in ILs is a widely occurring phenomenon, given a high degree of code switching in ILs. They cause the coverage of the MA to go down. They are not a part of the Oriya morph dictionary since they are foreign language words and can not be included in the ‘Oriya’ dictionary.

A possible solution to handle these would be creating a separate dictionary for them. To add a separate tag ‘foreign word’ in the MA, a dictionary of foreign words would have to be included and manually created. However, since foreign words have widely occurring in the language, creating an exhaustive list would be required for the MA to tag them as ‘foreign word’. This would be expensive in terms of time and resources. Further, though a work around for the problem, this is not a very good option either, as this would call for capturing too many irregularities by way of the inflections they take (or do not take). Capturing these irregularities falls out of the purview of our MA, as this would entail entering all these types of inflections in the dictionary. Since their taking of inflections is a productive process, this may make the task more complex, and may also fail generalization.

## 4.2 Analyzing Oriya Compound Verbs

Though simple verbs could be handled by creating paradigms for them, in Lttoolbox with relative ease, handling Oriya compound verbs (CV) proved quite a challenge for us. Before we go on to discuss the issues we came across in this, we would like to discuss briefly about CV in general, and about Oriya CV in particular:

A Compound verb consists of two verbs (v1, v2), yet acts as a single verb. One of its components is a ‘secondary’ verb which carries inflections like gender, number, person, tense, aspects and modality and the other, the ‘main’ verb which carries most of the semantics of the compound, and determines its arguments. The ‘secondary’ verbs “cannot be said to be predicating fully, though they are clearly not entirely devoid of semantic predicative power” [5].

Forming compounds is a highly productive process in IL. In languages like Hindi and Oriya, secondary verbs are generally, a small set that form compounds with the ‘main’ verbs.

**Structure and Behaviour of Oriya Compound Verbs.** We have identified 13 ‘secondary’ verbs in Oriya, as seen below in Table 6.

**Table 6.** List of ‘secondary’ verbs in Oriya.

S.No	Secondary verbs	Meaning
1	jaa	go
2	de	give
3	ne	take
4	pakaa	throw
5	bas	sit
6	paar	can
7	aas	come
8	pad	fall
9	uth	awake
10	chaal	walk
11	saar	finish
12	aaN	bring
13	he	happen

In Oriya CV the stem vowel ‘-i’ attaches to the ‘main’ verb, which in turn is followed by a ‘secondary verb’ from a limited number of verb roots that occur as ‘secondary’ verbs. For example:

```
se so-i-pad-il-aa
he sleep-stemvowel-fall-pst-agr.
‘He fell asleep’
```



The stem vowel ‘-i’ is different from an aspectual marker, though both have/ take the same form. The difference between them is that the aspectual marker is followed only by an auxiliary verb while the non-aspectual marker which is a ‘stem vowel’ is followed by a secondary verb [12]. An example for this is:

- a. se lekh-i-ch-i  
 he write-prs-aux-agr.  
 ‘He has written’
- b. aame khaa-i-de-l-u  
 we eat-stemvowel-give-pst-agr.  
 ‘we have finished eating’

Also, while in simple verbs inflectional suffixes attach to the main verb root, in CV the inflectional suffixes attach to the secondary verb, since Oriya is an agglutinative language, and these two verbs occur together. Thus the two verbs together arrive at a derived root. And so, we get two roots, a ‘main’ root and a ‘derived’ root.

Thus, since Oriya CV are different from simple verbs, in structure and behaviour, we can’t analyze them like simple verbs even though they occur as a single entity most of the time. Also, since Oriya CV are composed of two verbs (v1, v2) agglutinated together, that arrive at a derived root, the output of our MA should give this inflectional information for each derived root, in order to capture the information about their structure and the derivation happening in it.

For example, in ‘khaaidelaa’ (finished eating), the root khaa ‘eat’ is where we get the information of the action ‘eat’. When the secondary verb attaches to the main verb, another root ‘khaaide’ is derived. Our morph’s output for the derived verb ‘khaaidelaa’ should thus be:

```
~khaaidelaa/root:khaa<droot:khaaide>
<dsuffix:de><cat:v><gen:any>
<num:sg><per:a><tam:ilaa>$
```

However, this may not be a feasible solution for us, since information pertaining to such output will have to be incorporated in the dictionary for each CV, making this a cumbersome task. Besides, since there wouldn’t be any (scope of) generalization here, this would beat the purpose of using the paradigm approach.

Another solution for this would be the Apertium way—using nested paradigms to handle derivational forms, since Oriya CV are composed of combinations of verbs from the set of Oriya (main) verbs. “The use of nested paradigms is to facilitate the processes of derivation followed by inflection.” [16]

Here, the paradigms of secondary verbs would be ‘called’ upon, from within the main verbs’ paradigms to arrive at their compounds. For instance, the verb ‘khaaideichi’ is derived from the main verb ‘khaa’ (eat) and secondary verb ‘de’ (give) to form a compound. So the paradigm for ‘de’ is called from within the paradigm of the verb ‘khaa’. Likewise, other secondary verbs would be ‘called’ from within the paradigm ‘khaa’ to form compounds.

However, not all verbs of a paradigm class take the same secondary verbs to form compounds. There are verbs that fall under the same paradigms (since they share same types of inflections) that form compounds with different sets of

secondary verbs. For example, the verbs ‘khaa’ and ‘gaa’ are classified under the same paradigm class, but they take different secondary verbs. Thus, if we call all the secondary verbs that go with ‘khaa’, within the paradigm ‘khaa’, then while processing, the analyzer gives a similar output for ‘gaa’ also, though they don’t take same secondary verbs. We say ‘gaaithilaa’ ‘started singing (suddenly)’ but we don’t say ‘khaaithilaa’ ‘started eating suddenly’. It thus leads to some ungrammatical structures also.

It needs a mention here, that though the nested paradigm approach may work for a MA, from the perspective of generation it may lead to generation of ungrammatical structures. Since these two modules are obtained from a single morphological dictionary (depending on the direction they are read from—left to right for analyzer and right to left for generator as given by [6], a different resolution is needed to resolve this.

Therefore, based on discussion above we conclude that using nested paradigms doesn’t seem to be the best option for the analysis of Oriya CV.

```
<sdef n="droot:khaaide" c="khaaide"/>
```

The third, and a very simple approach to resolve this issue of handling Oriya CV in our MA, would be entering the derived roots of the CV in the morphological dictionary, in the dictionary section. The morphological dictionary contains the root or ‘lemma’, the part of the lemma which is common for all inflected forms, that is ‘lemma cut’ and the paradigm name. We simply add the derived root and the lemma cut of the derived root in the place where this information is entered in the dictionary. This would save us the task and the effort of preparing separate paradigms for the compound verbs.

For example, the dictionary entry for the CV ‘khaaidelaa’ (finished eating) with the derived root ‘khaaide’ would be:

```
<e lm="khaaide"><i>khaaid</i>
<par n="d/e_v"/></e>
```

< par > in the entry indicates which paradigm from among the ones defined in the < pardefs >, the derived root belongs to. Here, the derived root ‘kaaide’ falls under the paradigm for the root ‘de’, since the CV ‘khaaidelaa’ takes the inflections of the secondary verb ‘de’. Thus reference to the ‘de’ paradigm through the element <par> saves us the effort of listing all the inflected forms of the derived root/lemma in the morphological dictionary entry.

The output our MA would give for the above example is:

```
^khaaide/khaaide<cat:v>
<gen:any><num:sg>
<per:m_h0><tam:imper>$
```

## 5 Conclusion and Future Work

In this paper we presented a paradigm based MA for Oriya using Lttoolbox from the Apertium toolkit. It is based on the concept of morphological paradigms. Currently it handles only inflectional morphology, and nouns, pronouns,

adjectives, verbs, compound verbs and indeclinables have been included in its morphological dictionary. Since the MA is currently in its preliminary stage, addition of remaining categories and increasing the dictionary size for existing categories will improve its performance and increase its coverage. Using the Oriya MA for other NLP tools such as part of speech tagger, chunker, spell checker, machine translation system for Oriya can also be created in future. These would be a useful resource for the language.

## References

1. Antony, P.J., Soman, K.P.: Computational morphology and natural language parsing for Indian languages: a literature survey. *Int. J. Comput. Sci. Eng. Technol.* 136–146 (2012)
2. Bapat, M., Gune, H., Bhattacharyya, P.: A paradigm-based finite state morphological analyzer for Marathi. In: 23rd International Conference on Computational Linguistics, pp. 26–34 (2010)
3. Bharati, A., Chaitanya, V., Sangal, R., Ramakrishnamacharyulu, K.: *Natural Language Processing: A Paninian Perspective*. Prentice-Hall, India (1995)
4. Bharati, A., Sangal, R., Sharma, D.: SSF: Shakti standard format guide. Technical report, IIT Hyderabad (2007)
5. Butt, M.: The light verb jungle. In: *Workshop on Multi-verb Constructions* (2003)
6. Forcada, M., Bonev, B., Rojas, S., Ortiz, J., Sánchez, G., Martínez, F., Armentano-Oller, C., Montava, M., Tyers, F.: Documentation of the open-source shallow-transfer machine translation platform *apertium* (2008)
7. Faridee, A.Z.M., Tyers, F.M., Others.: Development of a morphological analyser for Bengali. *Universidad de Alicante, Departamento de Lenguajes y Sistemas Informáticos* (2009)
8. Jena, I., Chaudhury, S., Chaudhry, H., Sharma, D.M.: Developing Oriya morphological analyzer using Lt-toolbox. In: Singh, C., Singh Lehal, G., Sengupta, J., Sharma, D.V., Goyal, V. (eds.) *ICISIL 2011. CCIS, vol. 139*, pp. 124–129. Springer, Heidelberg (2011)
9. Kar, K.C.: *Taruna Sabdakosha, vol. 1. Grantha Mandir, Cuttack* (2000)
10. Kulkarni, A., Shukla, D.: Sanskrit morphological analyzer: some issues. In: *Festschrift, B.K. (ed.) Volume by LSI* (2009)
11. Lttoolbox. <http://wiki.apertium.org/wiki/Lttoolbox>
12. Mahapatra, B.P.: *A Synchronic Grammar of Oriya*. Udaya Narayana Singh, Central Institute of Indian Languages, Mysore (2007)
13. Parameshwari, K.: An implementation of APERTIUM morphological analyzer and generator for Tamil. In: *Parsing in Indian Languages*, p. 41 (2011)
14. Sahoo, K.: Oriya nominal forms: a finite state processing. In: *TENCON 2003, Conference on Convergent Technologies for Asia-Pacific Region, vol. 2*, pp. 730–734. IEEE (2003)
15. Shabadi, K.: Finite state morphological processing of Oriya verbal forms. In: *Proceedings of EACL-2003 Workshop on Computational Linguistics for the Languages of South Asia: Expanding Synergies with Europe*, pp. 49–56 (2003)
16. Vaidya, A., Sharma, D.: Using paradigms for certain morphological phenomena in Marathi. In: *7th International Conference on NLP (ICON-2009)*, pp. 132–139 (2009)