Semi-automatic Acquisition of Two-Level Morphological Rules for Iban Language

Suhaila Saee[1,2], Lay-Ki Soon[2], Tek Yong Lim[2],
Bali Ranaivo-Malançon[1], and Enya Kong Tang[3]

[1] Faculty of Computer Science and IT, Universiti Malaysia Sarawak,
Jalan Datuk Mohd Musa, 94300 Kota Samarahan, Sarawak, Malaysia
[2] Faculty of Computing and Informatics, Multimedia University,
Persiaran Multimedia, 63100 Cyberjaya, Selangor, Malaysia
[3] School of Computer Science and Information Technology,
Linton University College, Mantin, Negeri Sembilan, Malaysia

Abstract. We describe in this paper a semi-automatic acquisition of morphological rules for morphological analyser in the case of under-resourced language, which is Iban language. We modify ideas from previous automatic morphological rules acquisition approaches, where the input requirements has become constraints to develop the analyser for under-resourced language. This work introduces three main steps in acquiring the rules from the under-resourced language, which are morphological data acquisition, morphological information validation and morphological rules extraction. The experiment shows that this approach gives successful results with 0.76 of precision and 0.99 of recall. Our findings also suggest that the availability of linguistic references and the selection of assorted techniques for morphology analysis could lead to the design of the workflow. We believe this workflow will assist other researchers to build morphological analyser with the validated morphological rules for the under-resourced languages.

Keywords: Morphological rules, Rules extraction, Under-resourced language, Morphological analyser.

1 Introduction

Morphological analyser is a first processing task requires in Natural Language Processing (NLP). Morphological rules are crucial components in the analyser in order to analyse and generate the input word. The conventional method in acquiring the rules for morphological analyser was done using handcrafted, which has led to an ambiguity [1]. Therefore, the acquisition of the rules has received much attention from the researchers to automate the acquisition of the rules [2]. To automate the acquisition of the rules, there are two main components required as input: a) sufficient of linguistic references i.e. dictionary with stems and inflected words, the classification of words and affixes as well as a training data set and b) the selected techniques to acquire the rules that are depending

A. Gelbukh (Ed.): CICLing 2013, Part I, LNCS 7816, pp. 174–188, 2013.

on the availability of input types. The aim to acquire the rules automatically has been achieved with the sufficient linguistic references and precise techniques. However, we encountered the needed requirements became constraints to develop the analyser for under-resourced language (U-RL) by taking Iban language as a study case. This is because, at the morphological level, Iban language has insufficient linguistic references in terms of morphological rules, no morphological analyser for Iban yet and lack of linguists. While, selection of various techniques available for morphological analyser and generator that accommodate with Iban language has led to the other constraint at the linguistic level. Iban language is spoken by the Iban, a largest ethnic group in Sarawak, Malaysia formerly known as the Sea Dayaks [3]. Since 2008, Iban language has been adapted as one of the Malaysian Certificate of Education (fifth form) examination subject due to the significant of the language [4].

An objective of our work is to determine the morphological rules for building Iban morphological analyser. Previous works have only focused on the method of acquiring the rules automatically from the sufficient resources. Since this is a first research work conducted for under-resourced languages in Sarawak, we are hoping to fill the gap in term of acquiring morphological rules by inducing morphological information from raw text in a semi-automatic way, in the case of Iban language. The result from the induction process will be validated to ensure the correctness of the acquired morphological information in the later stage. The validated information will be used to extract the candidate rules and later to be applied in the two-level morphology.

Section 2 surveys the works related to acquisition of corpus and morphological rules and formalism to build the morphological analyser. Our proposed semi-automatic workflow and its components from the acquisition of corpus to Iban morphological analyser are presented in section 3. Section 4 describes standard metrics to be used in evaluation, and discusses the results from the analyser.

2 Related Works

Corpus and morphological rules are main requirement input for the morphological analyser to analyse and generate a structure of word. As one could derive the rules from the corpus; therefore, corpus acquisition plays an important role in building the automatic morphological analyser. For resourced languages like English, the corpus can be obtained freely from the Internet or any trusted sources such as linguists or web crawler. However, a main problem encountered for under-resourced languages is no resources at all in term of unavailability of the internal structure when word is analysed. For the case of under-resourced languages, the corpus can be obtained from the three types of sources that are dictionary, written text, and reference grammar book. The sources can be in either hardcopy or softcopy version. According to [5], if the sources are available in the hardcopy version, transformation into digitisation needs to be done

urgently in order to get the softcopy version. Then, once can proceed to the next process, which is acquiring the morphological rules including morphographemic rules and morphotactics (also known as morphosyntax) information.

There are two possible ways to acquire the morphographemic rules, for under-resourced languages and resourced languages, either manual or automatic. For a manual acquisition, the linguists are required to hand-craft the rules rather applying machine learning technique in an automatic way. The two-level morphology invented by [1] had successfully implemented the rule-based approach that the acquisition of the rules is hand-crafted by the linguists in two-level format. Since the process requires a lot of linguistic works and expertise in preparing hundred or more input, [2] proposed an automatic acquisition of two-level morphological rules. The proposed approach involving segmentation of a set of pairs stem and inflected word, and from that determines the desirable two-level rule set. The word pairs were extracted from machine-readable dictionaries of Xhosa, agglutinative language, and African. In this approach, the dictionary that provides stems and inflected words was the main requirement. More recently, [6] adapted and upgraded [2] model into different languages in their works. They had implemented the model for agentive nouns in two languages i.e. English and Macedonian. They also used the word pairs of base and derived words and segmented it using Brew edit distance, the best result gained after compared with the other edit distance algorithms, rather sequence edit distance as implemented in the model. Besides, [7] also segmented the word pairs of base and inflected word of Tagalog, a morphologically complex language, to acquire the rules. The word list used to supervise the analyser was derived from Tagalogs Handbook.

For the morphotactics information, the acquisition could be in either hand-crafted rules/ lexicon building or re-write for learned. There have been a number of studies on lexicon building of the information. In [8] work, the requirement of large lexicon building for the prototype of morphological analyser for Aymara, a highly agglutinating language, was the crucial part. Indeed, the lexicon included two types of dictionaries that written in XML format which were root and suffix. In contrast, [9] approach has applied morphological paradigms in their morphological analysis for Hindi, Telugu, Tamil and Russian languages. As our work is closest to Bharati's approach, we thus adapted the paradigm method into our work. However, in his work, the paradigms were including the morphographemic and morphotactis (suppletion) processes. Meanwhile, we only adapted the morphotactics information in the paradigm class without part of speech as we only have stem and segmented word at the moment.

3 Our Approach

Four main steps involved in this work including wordlist acquisition, stems and prefixes acquisition, converting stems and prefixes into the two-level format and Iban morphological analyser. Fig. 1 depicts the morphological rules acquisition methods applied in the existing approaches that lead to the proposed approach.

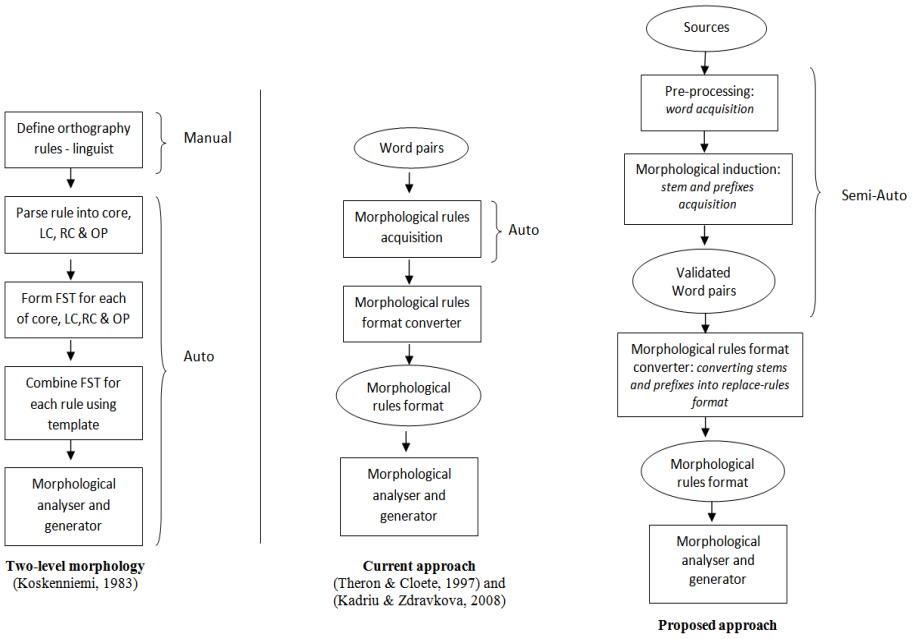

Fig. 1. Comparison between the existing and proposed approaches

In this section, we shall discuss further on each of the steps required from the proposed approach.

3.1 Step 1: Wordlist Acquisition

The first step in this work requires three activities in preparing the wordlist, which are data collection, data transformation, and data compilation. The purpose is to create a corpus and wordlist from the raw text, which will be used as input in the later processes. The activities shall be discussed as follows.

Data Collection. The required linguistic references to construct the corpus are a dictionary with derivation words, written texts, and a reference grammar book with examples of sentences. Similar to this work, [10] and [11] also used reference grammar books to avoid labour-intensive resources creation. Moreover, Feldman discovered that the reference grammar books are a perfect starting point for automatic morphology analysis.

Transformation. Digitisation including Optical Recognition Characters (OCR) and conversion is the process taken to deal with either a hardcopy or softcopy format of materials.

Compilation. Since our research is on building a morphological analyser, our linguistic units are words. Thus, we need to extract the words from all previously acquired linguistic references. We extracted the specific information from each of the linguistic references, then, we created new text files for each of the extracted information. Lastly, we compiled the information all together into one text file to get a wordlist form. To ensure the quality of our wordlist, we corrected all typos or unknown characters. The created wordlist should be error free, in other words, cleaned data. This wordlist will be the input of the morphology induction. Table 1 indicates a summary of the required references.

Table 1. Linguistic references for wordlist acquisition

References	Electronic version	Hardcopy version
Dictionary	Extraction of entry and sub-entry	Scanning, OCR & post-editing
Grammar book	Words extraction from example of sentences	Manually type the examples
Written text	Words extraction	Scanning, OCR & post-editing

3.2 Step 2: Acquiring Candidate Stems and Prefixes

At this second step, we shall present only the work done on the automatic recognition of prefixes and stems. Therefore, the morphology induction process has been settled to acquire stems and prefixes from the wordlist. This process enabled us to induce morphological information without prior to linguistic knowledge. Indirectly, the process could minimize human expertise control. This sub-section described our morphology induction workflow.

Feeding the Wordlist to *Linguistica*. The software application used to induce the morphological information from the wordlist was *Linguistica*, an open source tool [12]. Similar with other softwares i.e. Morfessor [13], Paramorph [14], *Linguistica* also applies an unsupervised machine learning technique that based on the frequencies of patterns of words that can be calculated at different levels of granularity. The requirement size of data from *Linguistica* is minimum 5000 words up to 500,000 words in order to get accuracy result. In this work, we have chosen *Linguistica* as we would like to show that our workflow is able to acquire the morphological rules without considering a good or poor result produced by one tool.

In general, there were two types of results produced by *Linguistica* which were unsegmented and segmented words. For the unsegmented words, it was produced due to its low frequency of words in the segmentation and detected to appear only one time. Thus, this file would be a potential test set for the morphological analyser later. While, the segmented words file used and analysed according to users need. For example, the previous works used *Linguistica* to find the allomorphy [15] and to see the morphological patterns from the generated signature of the desired languages [16]. More details on the *Linguistica* can be

found in [12]. Fig. 2 shows the interface of *Linguistica* which its screen is divided into two parts. The first part, on the left, is known as the tree that showing general information of the selected corpus. While, the second part, on the right, is known as the collections and text. From the collections, we could get detailed information of corpus like list of stems, affixes and signatures. In contrast, the text is used for feedback to user in a few cases.

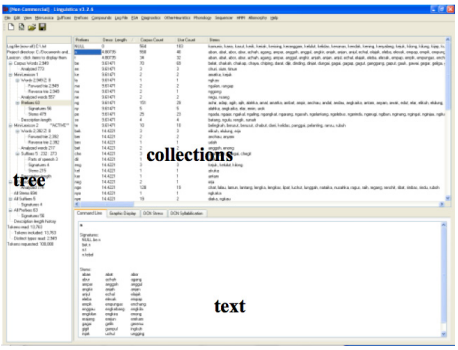

Fig. 2. An interface of *Linguistica*

In this work, we demanded the information of prefixes and stems therefore; *Linguistica* generated four main files which were list of *signatures*, *words*, *stems* and *prefixes*. The information from the *signature* file returned a maximal set of stems and suffixes with the property that all combinations of stems and suffixes were found in the wordlist. While, the *list of words* file held frequency of one word appeared in the wordlist. Since our work focused on the acquisition of prefixes and stems, we could extract the required information from the *list of stems* file by referring to the information in the *corpus count* and *affix count* columns. The *list of prefixes* file gave us the number of prefixes extracted from the wordlist are those that we have considered as candidate prefixes. We used the term candidate because, at this stage, we are still unsure about their real status yet. Fig. 3 shows a result generated from the *list of stems* file.

Extracting Result from *Linguistica*. To get a list of possible candidate prefixes and stems, we automatically extracted the morphological information, namely, list of stems and prefixes from the list of *stems* file in the *Linguistica* result [17]. The extracted information will be input to the validation process for a later stage. Fig. 4 shows an example of the list of candidate.

3.3 Step 3: Validating Candidate Stems and Prefixes

We proceeded with an automatic validation with a purpose to ensure the generated rules later were already validated. The availability of the list of candidate

```
# Stem Count
# ------------
  477

# Index | Stem          | Confidence | Corpus Count | Affix Count | Affixes
# ------------------------------------------------------------------------------
  1       gagai           PF_1         447            3             NULL be n
  2       ngajar          PF_1         215            2             NULL pe
  3       undan           PF_1         118            2             NULL ng
  4       nataika         PF_1         98             3             NULL nge ngea
  5       ngasuh          PF_1         73             2             NULL pe
  6       diaka           PF_1         38             2             nye nyen
  7       gumpul          PF_1         38             3             NULL be n
  8       ngarang         PF_1         31             2             NULL pe
  9       nyamai          PF_1         23             2             NULL pe
  10      engkebang       PF_1         16             2             n t
```

Fig. 3. Example of a result from the *list of stems* file

INDEX	STEM	PREFIXES
1	gagai	NULL be n
2	ngajar	NULL pe
3	undan	NULL ng
4	nataika	NULL nge ngea

Fig. 4. Example of the list of candidate stems and prefixes

prefixes and stems as listed in Fig. 4 above would be checked with existing morphology resources. The morphology resources are referring to the Iban dictionary and list of prefixes [1].

Table 2. Cases for Validation

File type	Cases
ValidatedWordList	Case A: stem = Yes & prefix = Yes Case B: stem = Yes & prefix = NULL Case C: stem = No & prefix = Yes
ValidatedWord	Case D: stem = No & (prefix + stem) =Yes
InvalidatedWordList	Case E: stem = Yes & prefix = No Case F: stem = No & (prefix + stem) = No Case G: stem = No & prefix= NULL

Table 2 above highlights seven cases used in validating the list of candidate stems and prefixes. The seven cases resulted three types of validated files that are

[1] The list of prefixes is obtained from the *list of prefixes* file as discussed in the section 3.2

ValidatedWordList, ValidatedWord and *InvalidatedWordList*. From the *ValidatedWordList* file, we obtained morphological information for simple concatenation morphology. The information could be used as input in generating the morphological rules. In the *ValidatedWord* file, a rule of thumb is a combination of stem and prefix are accepted as the word existed in the dictionary. Meanwhile, the *InvalidatedWordList* file considers the existence stem should be rejected because the prefix either was not existed or NULL from the list of prefixes. We attempted to minimize the number of invalid data in this file to get maximum data of *ValidatedWordList*. In fact, we have found the morphological information for non-concatenation morphology like alternation phenomena from the last two files. For instance, phonological changes of *n* to *t* from two words: *n*aban and *t*aban.

3.4 Step 4: Representing the Validated Rules in Two-Level Format

Similar to the previous works [2,6], we were applying the two-level approach and implementing the Iban morphological analyser in *XFST* tool, *Xeror* software. An objective of this step is to show that the validated morphological rules could be applied in the two-level morphology. Thus, our next step was to convert the list of candidate stems and prefixes from the *Linguistica* result in the two-level format in an automatic mean. Then, manual correction of the converted rules in the required format was taking place. Following were activities taken in the process:

Morphological Rules Extraction. After the validation process, we now consider the validated list of stems and prefixes as the validated morphological rules. The information that we can derive from the rules is morphographemic and morphosyntactic information. However, the acquisition process will take place to differentiate the information. At this step, we are required to decide which formalism suits best with the information. Lastly, the information will be fed into *XFST* tool. The activities involve shall be discussed as follows.

Morphological Acquisition. The morphographemic information was acquired from the construction of stems and prefixes, which obtained from the *ValidatedWordList*. Besides, we could derive the morphotactics information from both two validated files that were *ValidatedWord* and *InvalidatedWordList*. We applied morphological paradigm in acquiring the morphotactics information since we only have the stems and construction of the stem and prefix. In this work, we applied paradigm prefixes that are considered similar to paradigm classes with the purpose to put in one group of prefix the stem and derivation words. This is consequent to the insufficient morphological information. See Fig. 5. We categorised the list of morphotactics information based on the type of prefixes. An objective is to identify the similarities of the words in term of its pattern.In

fact, the prefixes are obtained from the list of prefixes with the highest frequent occurrence. The list of morphotactics information consists of the prefixes that attached to its stems and derivation words.

Stem	Construction
1. andau	*ng*andau
2. ajar	*ng*ajar
3. udah	*ng*udah
4. antam	*ng*antam

Fig. 5. Example of morphological paradigm

Morphological Formalism. Two-level morphology formalism is used in this experiment and implemented in the *XFST* tool as mentioned earlier. Thus, the morphographemic information would be represented in a replace-rules format. Fig. 6 depicts the morphographemic rules in the replace-rules format. Meanwhile, the morphotactics information would be written in lexc, a lexicon to describe the spelling structure, which will be represented in a pattern-root format. See Fig. 7.

```
clear stack
define lbraces "^[" | "[" | "{" ;
define  IbanAssRules [f] ->l || .#. _
.o.
0 -> n g || %^N lbraces* _ ,, %^N -> 0 || _ lbraces* [ a | e | i
| u ] ! pengurus -> urus+peN- / ngarah -> arah+N-
.o.
[ g | k ] -> n g || %^N lbraces* _ ,, %^N -> 0 || _ lbraces* [ g
| k] ! pengurus -> kurus+peN- / ngurus -> kurus+N-
.o.
[ b | p ]-> n g e m || %^N lbraces* _ ,, %^N -> 0 || _ lbraces* [
b | p ] ! pengemesai -> besai+peN-/ ngemesai -> besai+N
```

Fig. 6. Morphographemic rules in replace-rules format

3.5 Step 5: Iban Morphological Analyser

The rules then would be fed into the *XFST* tool, to implement the Iban morphological analyser. The morphographemic rules were compiled into a rule transducer while, the morphotactics information compiled into a lexicon transducer. The lexicon and rule transducers were combined to create a single lexical transducer (a single FST) that could be applied to perform either analysis or generation. The analyser requires lexicographer to review the results in deciding which roots are valid since it analysed all possible stems. Fig. 8 shows a sample of output from the Iban morphological analyser.

```
! definitions of the Cons and Vows
Definitions
C = [b|c|d|f|g|h|j|k|l|m|n|p|q|r|s|t|v|w|x|y|z] ;
V = [a|e|i|o|u] ;

LEXICON Root
        FDPre3 ;

LEXICON FDPre3
< [ 0 .x. b e r ]"@P.PREF.ber@" > Stems3 ;
< [ 0 .x. b e s ]"@P.PREF.bes@" > Stems3 ;
< [ 0 .x. p e r ]"@P.PREF.per@" > Stems3 ;

LEXICON                         Stems3
<[V C (V)] ^ {2,4}>             Pref3;      ! amal
<[(V) (C) C] ^ {2,4}>           Pref3;      ! antam
<[V (C) (C)] ^ {2,4}>           Pref3;      ! andau
<[C V (C)] ^ {2,4}>             Pref3;      ! lari
<[(V) (C) (C)] ^ {2,4}>         Pref3;      ! anchau

LEXICON    Pref3
      ber- ;
      bes- ;
      per- ;

LEXICON ber-
< "+ber-":0 "@R.PREF.ber@"    >     #     ;

LEXICON bes-
< "+bes-":0 "@R.PREF.bes@"    >     #     ;

LEXICON per-
< "+per-":0 "@R.PREF.per@"    >     #     ;
```

Fig. 7. Morphosyntactic information in lexc format

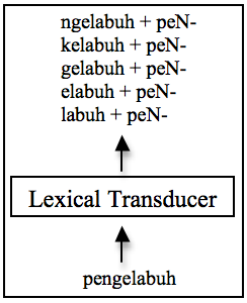

Fig. 8. Result from the Iban morphological analyser

4 Evaluation

In this section, we evaluated the Iban morphological analyser in terms of morphological rules generation and the performance of analyser.

4.1 Experimental Setup

Corpus Creation. We used list of entries under letter *N* in preparing data sets to induce the morphological information. The consequence was that letter *N* has the highest number of possible prefixes among other letters in alphabet.

Test Set Preparation. As mentioned earlier in the section 3.2, we are using the unsegmented data produced by *Linguistica*. Thus, we have two types of test sets that were a) 2400 unsegmented data and b) Top 100 words and 100 random words from the 2400 unsegmented data. The objectives of using a different test sets were because we attempted:

- To see:
  - the coverage in term of the number of morphological phenomena and word-formation that have been covered and yet to be covered with the generated rules.
  - the accuracy of the generated rules in analysing and generating the input word (test set (a)).
- To understand in details type of errors in the morphological analyser which are a) 100 top words and b) 100 random words both from the 2400 unsegmented data (test set (b)).

Evaluation Metric. We used two standard metrics that were precision and recall for evaluating the coverage and accuracy of the analyser.

4.2 Result Analysis

To discover types of morphological phenomena and word-formation analysed by the Iban morphological analyser, we have conducted a quantitative analysis. For the first analysis, we were using test set (a), 2400 unsegmented data. The following sub-sections shall discuss in details the overall analyses. See Fig. 9.

A pie chart is showing the number of correct and incorrect analyses with the 25 generated rules evaluated using 2400 unsegmented data. The 25 rules are *n, ny, ng, ngem, nge, ne, be, bes, ber, bete, beke, ke, per, dipe, che, te, se, me, en, eng, pe, pen, penge, sepe* and *sepen*. Out of 25 rules, six rules were never applied in the data, e.g. *be, ber, bes, beke, bete* and *dipe*. Although the validated rules were totally correct from the *Linguistica*, these rules have not been applied in either analysing or generating the unsegmented data. While, the other 19 validated rules have analysed 937 correct data and 1463 incorrect data. As we can see from Fig. 9, the data contains 47% of root words. This indicates that the root words were the majority in the test sets. On the other hand, 53% of words have been analysed as 41% of correct and 12% of incorrect analysis. From the analysis, it shows the morphological analyser was able to analyse simple concatenation and non-concatenation morphology. However, there were a number of morphological phenomena and word-formations that the morphological analyser still unable to cover when we see from the percentage of the incorrect analysis which will be further explained in error-analysis section.

Morphological analyser can be evaluated by using standard measures, which are precision and recall. In this work, we evaluated the accuracy of the extracted morphological rules from the precision. From the precision result, 0.7656, it shows the accuracy of the morphological analyser is nearly to 1.000. This was due to the nonexistent of the root word in the dictionary. For instance, the word

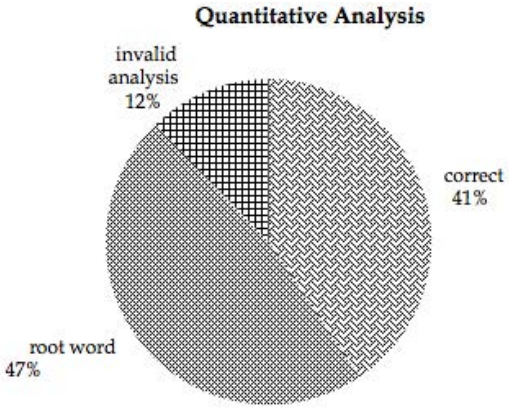

Fig. 9. Validated rules applied on 2400 unsegmented data

temegah should not be analysed as *te+megah* as it is a root word. Instead, the analyser should analyse the word as *temegah*. Since the morphological analyser found prefix *te-* existed in the list of prefix, thus, the analyser has analysed as *te+megah*. Therefore, there is still a room for improving the rules so that the morphological analyser able to differentiate between the one should be analysed and a root word. In order to achieve 1.000, we should enhance the lexical entry from the dictionary with the derivation words.

To determine the coverage of the Iban morphological analyser, we measured the coverage of the analyser from a recall. From the recall result, 0.9928, the analyser still has not covered yet for overall full reduplication and circumfixation, although it covers only two types of partial reduplication i.e. *che* and *ne*. After examined on the real data, the first test set (a), the morphological analyser could handle the non-concatenative morphology besides simple concatenation such as alternation, which are *peN* and *sepeN*.

We tested on the test set (b) to investigate the error analysis for the rules extraction in a quantitative analysis which were a) 100 top words and b) 100 random words both from the 2400 unsegmented data. From here, we have manually identified the reasons of the failures.

Results of the error analysis are shown in Fig. 10. There are two broad categories of error types which: *wrong segmentation* and *the rules were not applied* in the segmentation. Specifically, the 100 top data reached the highest peak of 23 data of the wrong segmentation at the circumfixation error type as the inexistent of the circumfix rules in the morphological analyser. Similar to the 100 random data, which has the same level of the highest peak of 23 data of the wrong segmentation at the root word error type. This was due to the insufficient of lexical entry in the Iban dictionary. From the graph, the top data felt down when there were no particular rules to be applied on the full and partial reduplications. For the random data, none of the rules had been applied on 7 root words as there were no prefixes matches to the inputs.

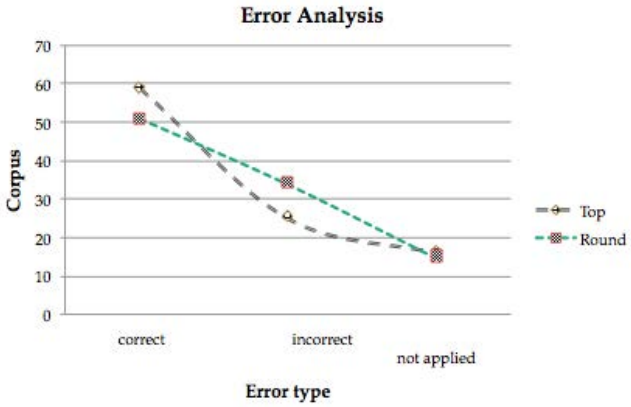

Fig. 10. Error Analysis for 100 top and 100 random unsegmented data

The error analysis we examined from the second test sets (b) is discussed as follows.

1. Wrong segmentation:
   (a) Unknown prefixes - The morphological analyser returned unknown prefixes for the simple concatenation. For example, *tebe, s, diper, sel* and *ter*. The analyser was able to analyse it because *te* existed in the list of prefixes. However, a word of *teberumpang* has been analysed as *berumpang* + *te-* which supposed to be segmented as *rumpang* + *tebe-*. When checked in the Iban grammar book, *tebe* is one of the existing prefix in the Iban language.
   (b) Unseen root word - *temegah* is a root word that does not list in the Iban dictionary. However, the analyser has analysed *temegah* as *megah*+*te-* because *te* exists in the list of prefix and *megah* exists in the Iban dictionary. This case happened due to the dictionary has a lack of Iban entries.
   (c) Unable to analyse - The analyser was unable to analyse a morphological phenomena with more than one combination of affixes. For instance, *ngenchuri* has been analysed as *ng* + *en* + *churi* instead *ng* + *enchuri*. This was because the phenomena do not exist in the list of segmented data.
2. Rules never applied in the analysis:
   (a) Full reduplication - The Iban morphological analyser was unable to analyse the full reduplication due to inexistent of the rule to recognise hyphen (-) from the input, e.g. *chamang-chamang*.
   (b) Partial reduplication - The analyser only analysed *che* and *ne* as the rules are listed in the list of rules.

5 Conclusions and Future Work

In this study, we presented a semi-automatic method for acquiring morphological rules of under-resourced language, in the case of Iban language. The workflow of acquiring the morphological rules plays an important role in developing the morphological analyser for under-resourced languages. We observed that the availability of linguistic references and the selection of assorted techniques for morphology analysis could lead to the design of the workflow. Furthermore, we discovered *Linguistica* tool could generate the non-concatenation morphology, besides simple concatenation. The results of this work indicate that the morphological induction and rules validation were the crucial processes. Although we have achieved 0.76 of precision but, there are limitations e.g. the unseen root words and the analyser was unable to analyse more than one combination of affixes. These errors occurred due to insufficient references of one language. To further our research we intend to improve our workflow in a number of ways. First, noisy data including typo errors and no standardization in the Iban spelling can be overcame by avoid erroneous data as this could lead to the inaccurate results. Second, improving the Iban morphological analyser to wider the coverage by forming new rules from the incorrect results, so that it able to handle reduplications, circumfixation and combination of more than one affixes. Third, amend the lexical entry and its derivation words in the dictionary to provide better results. Finally, enhance the computer involvement in the analyser at the evaluation phase owing to time consuming. Nevertheless, we hope our work will serve as a starting point for future studies mainly for the under-resourced languages.

Acknowledgments. We gratefully acknowledge Dr Kadriu A. for her valuable suggestions and discussions on the theoretical part of automatic rules acquisition. We also thank Dr Beesley K.R. who gave us much valuable advice in the early stages of this work.

References

1. Koskenniemi, K.: Two-Level Morphology: A General Computational Model for Word-form Recognition and Production. PhD thesis, University of Helsinki (1983)
2. Theron, P., Cloete, I.: Automatic acquisition of two-level morphological rules. In: Proceedings of the Fifth Conference on Applied Natural Language Processing, pp. 103–110. Association for Computational Linguistics (1997)
3. Sarawak Board Tourism: `http://www.sarawaktourism.com/en/component/jumi/about-people`
4. Ling, S.: Iban language for spm in 2008 (2008)
5. Karagol-Ayan, B.: Resource generation from structured documents for low-density languages. PhD thesis, University of Maryland, College Park (2007)
6. Kadriu, A., Zdravkova, K.: Semi-automatic learning of two-level phonological rules for agentive nouns. In: 10th International Conference on Computer Modelling Simulation (2008)

7. Yturralde, B.: Morphological rule acquisition for tagalog words using moving contracting window pattern algorithm. In: Proceedings of the 10th Philippine Computing Science Congress, Ateneo de Davao University (2002) ISSN 1908-1146
8. Beesley, K.R.: Computational Morphology and Finite-State Methods. IOS Press (2003)
9. Akshar, B., Rajeev, S., Dipti, M.S., Radhika, M.: Generic morphological analysis shell. In: SALTMIL Workshop on Minority Languages, (2004)
10. Feldman, A., Hana, J., Brew, C.: A cross-language approach to rapid creation of new morpho-syntactically annotated resources. In: LREC 2006, pp. 549–554 (2006)
11. Cucerzan, S., Yarowsky, D.: Bootstrapping a multilingual part-of-speech tagger in one person-day. In: Proceeding of the 6th Conference on Natural Language Learning - COLING 2002, pp. 1–7 (2002)
12. Goldsmith, J.: Unsupervised learning of the morphology of a natural language. Computational Linguistics 27(2), 153–198 (2001)
13. Creutz, M., Lagus, K.: Unsupervised morpheme segmentation and morphology induction from text corpora using morfessor 1.0, Helsinki University of Technology (2005)
14. Monson, C., Carbonell, J., Lavie, A., Levin, L.: Paramor: Finding paradigms across morphology (2009)
15. Karasimos, A., Petropoulou, E.: A crash test with linguistica in modern greek: The case of derivational affixes and bound stems. In: International Conference on Language Resources and Evaluation, LREC 2010 (2010)
16. Blancafort, H.: Learning morphology of romance, germanic and slavic languages with the tool linguistica. In: International Conference on Language Resources and Evaluation, LREC 2010 (2010)
17. Dasgupta, S., Vincent, N.: Unsupervised morphological parsing of bengali. Language Resources and Evaluation 40(3-4), 311–330 (2007)