International Journal of Psychosocial Rehabilitation, Vol. 24, Issue 06, 2020
ISSN: 1475-7192

Part-of-Speech (POS) Tagger for Malay Language using Naïve Bayes and K-Nearest Neighbor Model

[1]Shamsan Gaber, [*2]Mohd Zakree Ahmad Nazri,
[3]Nazlia Omar, [4]Salwani Abdullah

Abstract--- *Part-of-Speech (POS) tagging effectiveness is essential in the era of the 4th industrial revolution as high technology machines such as cars and smart homes can be controlled using human voice command. POS tagger is important in many domains, including information retrieval. POS tags such as verb or noun, in turn, can be used as features for higher-level natural language processing (NLP) tasks such as Named Entity Recognition, Sentiment Analysis, and Question Answering chatbots. However, research on developing an effective part-of-speech (POS) tagger for the Malay language is still in its infancy. Many existing methods that have been tested in English have not been tested for the Malay language. This study presents an experiment to tag Malay words using the supervised machine learning (ML) approach. The purpose of this work is to investigate the performance of the supervised ML approaches in tagging Malay words and the effectiveness of the affixes-based feature patterns. The Naïve Bayes and k-nearest neighbor models have been used to assign a specific tag for the words. A corpus obtained from Dewan Bahasa dan Pustaka (DBP) has been used in this experiment. DBP has defined 21 tagsets (categories) for the corpus. We have used two sizes of corpora for the tests, which have 20,000 tokens and 40,000 tokens. Moreover, affixes-based feature pattern engineering has been extracted from the corpora to improve the process of tagging.*

Keywords--- *Natural Language Processing, Machine Learning, Part-of-speech Tagging, Malay Language.*

I. INTRODUCTION

President Obama's victory in 2008 is considered one of the breakthrough of natural language processing as it played an essential role in his campaign for the US presidency. Now, political campaigns, marketing, product research, and the news media are convinced that the Internet can also be mined for useful insight or patterns about public opinion. NLP-based applications for education are being used for assisting the progress and improvement in the learning ability of students (Alhawiti, 2014). The application of NLP in the education system is also beneficial for analyzing errors such as grammatical and stylistic errors. Teachers can easily mark these errors in the papers of students. Political campaigns, marketing, businesses, and education are using NLP powered applications to support their everyday decision makings. Natural language processing (NLP) can be defined as an area of research and application that explores how computers can be used to perform useful tasks involving human language that enables human-machine communication (Chowdhury, 2003; Jurafsky & Martin, 2014). NLP generally involves six phases, including phonetics and phonological analysis, morphological analysis, syntactic analysis, semantic analysis, pragmatic analysis, and discourse integration (Allen, 1995).

POS disambiguation is the ability to computationally determine the POS of a word that is activated by its use in a particular context. It can also be defined as the process of assigning an appropriate POS tag for each word in a sentence.

[1,2,3,4]Center for Artificial Intelligence Technology, Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia, Bangi, Selangor, Malaysia, E-mail: zakree@ukm.edu.my

Received: 19 Jan 2020 | Revised: 06 Feb 2020 | Accepted: 12 Mar 2020

Fine-grained POS (morpho-syntactic or morphological) tagging is the process of assigning the POS, tense, number, gender, and other morphological information to each word in a sentence (Feldman, 2006; Schmid & Laws, 2008). POS tagging has the necessary and fundamental language analysis tasks in most of the NLP systems, such as corpus annotation projects, information extraction, and word-sense disambiguation. The output of the POS taggers is usually forwarded to another high-level language analysis task, such as named entity recognition (Benajiba, Diab, & Rosso, 2008) and syntactic parsing (E. S. Mohamed, 2010).

Research in Part-of-Speech (POS) tagging has been documented since the mid-sixties (Flanagan, Rabiner, & Schafer, 1974; Lindberg, 1960; Simmons, Klein, & McConlogue, 1962). However, various methods and strategies have been produced and successfully applied in many applications, especially during World War II and the Cold War between the United States and the Soviet Union. Rival factions use computers in their quest for intelligence to address the fear of nuclear war and Soviet spies. Fear sparked the development of natural language processing in order to build a machine that can translate the Russian language to English. Today, the technologies culminating with modern marvels such as Watson, mobile device application Siri and cars that operate on voice command.

Approaches in NLP can be categorized into two main groups based on the nature of the knowledge they use: the linguistic and machine learning (ML) family. Linguistic-based taggers represent the knowledge involved as a set of rules or constraints written by linguists, while the ML is based on intelligent computational algorithms. Most of the new approaches in NLP originate from the field of ML. Part-of-speech tagging which is also called grammatical tagging or word-category disambiguation, is tretated as a classification problem by the ML researchers because the tagsets are the classes of grammatical type. Therefore, the output of an ML method is a model that can be used as an automatic classification that assigns each occurrence of a word to one category based on the evidence from the context. Theare are two type of learning which are the unsupervised method and supervise method. Unsupervised method for POS tagging means the method does not require a man-annotated corpora to build the model. Thus, unsupervised learning method is less accurate than supervised learning (Qin & Schuurmans, 2005). Supervised approaches proved to be more precise than other methods, given a large amount of manually tagged corpora (Navigli, 2009). If a supervised machine learning method is not given enough 'training materials', i.e. tagged corpora, the supervised method tagging accuracies drop substantially.

The lack of language resources, i.e., annotated training corpora, is a general problem even for well-studied languages (Marques & Lopes, 2001). The Malay language also suffers from the same problem. The Malay language is widely spoken in South East Asia, including the southern Philippine and East-Timor, with approximately 300 million users (El-Imam & Don, 2005). Tan (2003) states that Malay language is an inflectional language. The Cambridge Dictionary defines inflected language as "a language that changes the form or ending of some words when the way in which they are used in sentences changes", in which the language performs massive affixation, reduplication, and composition.

Malay language is a type of Indo-European language in which Baldwin and Awab (2006) describe as "a severely underrepresented (language) in text processing terms". To date, compared to English, Malays language still has relatively few resources with a small corpus of texts. A corpus is a computer-based collection of natural language that is designed and developed to be representative of a language through careful selection. A POS tagged corpus is essential to this research as it provides the required source of learning for the selected machine learning algorithm. Given a tagged corpus as training 'dataset', a machine learning algorithm produces a model that can be used to classify other untagged words. Even though the Malay language is known for its unique characteristics and has attracted many linguists from the west to study the language, the severe shortage of Malay tagged corpora has impeded Malay NLP researchers in developing Malay NLP technologies and applications. Annotated Malay corpora are limited and not publicly available. Examples of private data include the Malay Practical Grammar Corpus (I. H. Abdullah, Ahmad, Ghani, Jalaludin, & Aman, 2004), the Dewan Bahasa Pustaka (DBP) Database Corpus1, the Malay Corpus by Unit Terjemahan Melalui Komputer from the University Science of Malaysia (Ranaivo-Malancon, 2005) and more recently the MAlay LEXicon (MALEX) (Zuraidah, 2010).

This work describes two supervised tagging models using supervised machine learning algorithms. The models that we conducted the study on are Naïve Bayes (NB) and k-nearest neighbor (KNN). Moreover, some languages have a richer morphology than others, requiring the POS-tagger to have a broader set of feature patterns (Giménez & Màrquez, 2006). The Malay language is one of the languages that are rich morphologically. Therefore, the morphology can be used as a feature pattern to help the models improve the performance of the tagger model. However, most of the feature patterns discovered by Giménez & Màrquez (2006) or Brants (2000) have not been tested on the Malay language. The trade-off of using different features patterns for the Malay languages has still not been discovered. Therefore, this research will conduct a comparative study on the feature set effectiveness to predict the POS tag of the words.

The rest of the paper is organized as follows: Section 2 discusses the related works. Section 3 describes the used corpora and feature selection, as well as the description of the NB and KNN tagging approaches. Chapter 4 gives the experimental results and discusses them. Lastly, conclusions and future work appear in Section.

II. LITERATURE REVIEW

Research and Development (R&D) in the area of natural language processing for machine translation (MT) begin at the end of World War II. About 70 years have passed since then, and computing power has reached a level where personal assistants such as Siri and Cortana become a common thing. However, limited research is available for POS tagging for the Malay language. The lack of linguistic tools and limited access to computational resources daunt researchers from conducting further investigations on this language. One of the earliest papers discussing the Malay Natural language Processing, in particular, part-of-speech tagger, is the work of Al-Adhaeleh et al. (Al-Adhaileh, Kong, & Melamed, n.d.). They proposed an approach to construct a Bilingual Knowledge Bank. They used bitext alignment tools that have been proven their efficiency on the Malay language such as SIMR: a bitext mapping tool and GSA: a segment alignment tool. They tagged English sentence with part of speech (POS) and phrase structure tree produced by the Apple Pie Parser (APP). Later, the annotated English sentences with POS are compiled into a Structured String-Tree Correspondence SSTC structure. Next, the Malay SSTC structure of each Malay sentence will be generated based on the corresponding English SSTC.

Research on Malay linguistics has been explored thoroughly by Ranaivo-Malaicon in a series of publications (Ranaivo-Malancon, 2005). The studies include lexical and morphological analyses and tagging. The POS tags are inferred from the rule-based morphological analyzer. Building a morphological analyzer is computationally expensive and laborious. A study on Malay POS tagging to complement MALEX, the annotated Malay lexicon, by focusing on the problem of syntactic drift has been conducted (Knowles & Don, 2003). The tagsets are identified using a data-driven approach, and the study presented a list of possible syntactic drifts for the Malay language. Besides, the researchers performed a corpus-based approach for an analysis of the grammatical class in Malay (Knowles & Don, 2003, 2004).

An open-source corpus with over 26,000 Malay words that was extracted from the World Wide Web was used to develop a Malay sentence tokenizer, lemmatizer, and POS-tagger (Baldwin & Awab, 2006). However, the tags were not purely generated but were partially taken from the KAMI Malay-English lexicon of various genres (Quah, Bond, & Yamazaki, 2001), and the work was reported as incomplete. Nevertheless, this supervised approach of lemmatization achieved a 94.5% overall accuracy. A closely related work on bitext mapping is reviewed in (Al-Adhaileh et al., n.d.). A pattern recognition algorithm is known as the smooth injective map recognizer, and the geometric segment alignment algorithm is used to align the English and Malay texts.

Additionally, the prototype required a translation lexicon that is constructed from a machine-readable English-Malay dictionary and a lemmatizer. Tagging and lemmatization are performed using Brill's tagger (Brill, 1995). However, no work on POS tagging is involved in this research.

An approach called the “lazy man’s way” is proposed by Norshuhani, Oxley, Zainab, and Syed (2012). They implemented a statistical-based approach for word alignment by automatically projecting part-of-speech tags. This unsupervised learning method combines the N-gram and Dice Coefficient similarity function to align English texts with

Malay text. The experiment was performed on 25 terrorism news articles written in Malay text, which has 5413 word tokens. The results reached values of 86.87% for precision and 72.56% for recall.

In 2018, Ariffin & Tiun (2018) developed a Malay POS tagger, which was created using the QTAG model (Tufis & Mason, 1998). QTAG is a supervised machine learning (ML) POS tagging approach that requires a large amount of annotated training corpus data to tag the identified data accurately.

Moreover, a trigram hidden Markov model (HMM) for tagging Malay texts was introduced by Mohamed et al. (2011). It is a statistical POS tagger approach that predicts the POS for unseen words in the training corpus and can guess a word's POS based on the surrounding information's "Affixed-base". The POS tagger predicts the tag of the word based on the affixes, suffixes, and circumfixes. The best way to predict the POS of unknown words is obtained through the prefix information by considering the first three characters of the word. The accuracy of the tagger reached up to 67.9% for the unknown words. The tagset had an average of 1840 test tokens. The results show that the HMM is a promising method to predict tags for Malay words. However, the overall process of preparing the Malay corpus involves a costly morphological analysis.

III. METHODOLOGY/MATERIALS

As stated in the introduction, the primary goal of this research is to design and implement efficiently supervised machine learning POS tagging for the Malay language using the affixes-based features.

As shown in Figure 1, the methodology adopted for this work is as follows:

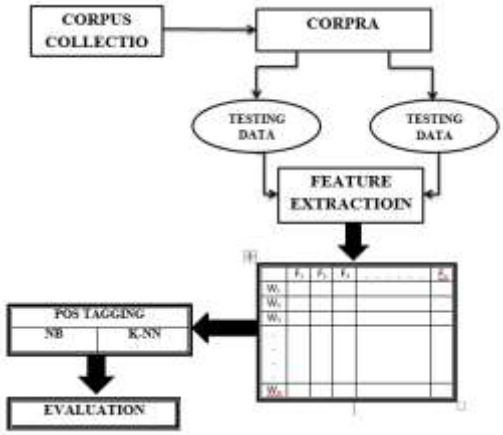

Figure 1: Part-of-Speech Methodology

3.1 Corpus Collection

The Dewan Bahasa Pustaka (DBP) tagset is the most acceptable tagset for the Malay language because DBP is the government body responsible for coordinating the use of the Malay language and Malay-language literature in Malaysia. The corpus, which is developed by DBP, has more than 115,000 tokens. In this work, we have implemented two sizes of sub-corpuses with 20,000 and 40,000 tokens for the experiments. The number for the tagset defined in the corpus is 21 (H. Abdullah, 1972).

3.2 Feature Extraction

There are several types of feature patterns that can be extracted from the corpus in our study. We focused on affix features. We have used the combination of multiple prefixes, suffixes, and circumfixes. Two of the prefixes have been tested at the beginning, as well as three prefixes, four prefixes, five prefixes, suffixes, and circumfixes.

Table 1: The List of Affix Features
({F1:value1, F2:value2,…,Fm: value m},"TAG")

Feature	Feature Name
Affix Features	F1. Five prefixes.

	F2. Four prefixes. F3. Three prefixes. F4. Two prefixes. F5. Five suffixes. F6. Four suffixes. F7. Three suffixes. F8. Two suffixes. F9. Five circumfixes. F10. Four circumfixes. F11. Three circumfixes. F12. Two circumfixes.

In our study, we applied these features to the model for classification and observed their effectiveness. The following Table 1 shows the list of features extracted from the datasets.

Affixes are divided into several categories, depending on their position concerning the stem. Table 2 explains the prefix, suffix, and circumfix affixes.

Table 2: Affix Examples and Description

Affix	**Example**	**Schema**	**Description**
Prefix	Meng-ajar	Affix-Stem	Appears before the stem
Suffix	Makan-an	Stem-Affix	Appears after the stem
Circumfix	Men-doa-kan	Affix-Stem-Affix	One portion appears before the stem, and the other portion appears after the stem

3.3 POS Tagging Models

The following sub-sections describe the models which are used for the POS tagging.

3.3.1 K-NN Tagger

The k-nearest neighbor (KNN) algorithm is a supervised machine learning algorithm for developing a classification or regression model and considered as one of the most popular classification techniques because of its simplicity. The KNN algorithm is designed based on the idioms 'birds of a feather flock together,' which means the algorithm search for similarity between a query and the available examples from the dataset. Therefore, KNN is a supervised machine learning technique which requires a training data set. The 'k' is a parameter that represents how many nearest neighbors to include in the majority of the voting process. Let's say we have a new unlabeled or unknown word that we want to classify (i.e., query) its type, either noun or verb. The KNN classifier looks for the k-nearest neighbor among the example in a dataset. If the parameter k is set 3, KNN will look for the 3-nearest neighbor. KNN will calculate the distance between the query example (i.e., the new word) and the examples from the data. It computes the similarity between the query and K training data. Those neighbors are ranked or sort orderly in ascending order based on their similarity scores. Then, the next process is to pick the first K entries from the sorted list and get the labels of the selected K. If the majority of the labels are a verb, then the unlabeled word would be labeled (i.e., classify) as a verb. The most common distance function is the Euclidean distance, which represents the usual manner in which humans think of distance in the real world:

$$D_{\text{Euclidean}}(x, y) = \sqrt{\sum_{i=1}^{m} (x_i - y_i)^2}$$

where x= (x1, x2,..., xm) and (y1,y2,..., ym) represent the m attribute values of two records.

To determine the class of a new test document:

Calculate the distance between the original test document and all of the instances in the training set.

Set the k-nearest document to the new test document in the training set.

Assign the new test document to the most common class among its KNN.

3.3.2 NB Tagger

Naive Bayes (NB) is an effective classification algorithm that is widely used for document classification. As a probabilistic model, the Naive Bayes classifier uses the joint probabilities of terms and their categories to estimate the probabilities of the categories given as a test document. The main advantages of Naïve Bayes are that it is simple, easy to implement, and has a superior performance algorithm. Two NB models are used for text classification. These models are the multinomial model and the multivariate Bernoulli model (McCallum, Nigam, & others, 1998). Based on the following Bayes' formula, the NB model is used in text classification:

$$P(C_i|d) = \frac{P(C_i)P(d|C_i)}{P(d)}$$

where P(C_i |d) is the posterior probability of class Ci given a new document d and P(C_i) is the probability of class Ci, which can be calculated by:

$$P(C_i) = \frac{N_i}{N}$$

where Ni is the number of documents assigned to class Ci, N is the number of classes, P(d|C_i) is the probability of a document d given a class Ci, and P(d) is the probability of document d.

Because of the independence assumption of the NB model, the probability of the document d can be calculated by:

$$P(C_i|d) = P(C_i)\prod_{k=1}^{n}(t_k|C_i)$$

where tk is a feature that co-occurs with class Ci. Additionally, we can calculate (t_k |C_i) by:

$$P(t_k|C_i) = \frac{1 + n_{ki}}{l + \sum_{h=1}^{l} n_{hi}}$$

where nhi is the total number of documents that contain feature tk and belong to class Ci, and l is the total number of distinct features in all training documents that belong to class Ci.

The NB model calculates the posterior probability for each class and then assigns document d to the highest posterior probability's class, i.e.,

$$C(d) = \underset{i=}{\operatorname{argmax}}|C|\ (P(C_i|d))$$

IV. RESULTS AND FINDINGS

In this section, the results of the two models are compared. This section also discusses the influence of the feature patterns that have been used in the model. Moreover, we compare the performance of the tagging process using the two-sized corpuses of 20,000 tokens and 40,000 tokens.

4.1 Experiments Settings

To perform part-of-speech tagging for the datasets that are obtained from Dewan Bahasa dan Pustaka (DBP), we split the experiments into two steps. In the first step, the datasets divided into two parts: the training dataset consists of 90% of the data as the training set, and the testing data consists of 10% of the data as the testing set as Table 3 shows. Moreover, we have chosen three values for the K value for the KNN model in the experiments, which are (1, 5, and 10).

Table 3: Corpus Setting

Corpus	Testing	Training
20,000	2,000	18,000
40,000	4,000	36,000

On the other hand, the other step uses the machine learning approach models (Naïve Bayes model, k-nearest neighbor) to predict the part-of-speech tag for the words. Afterward, there is an investigation summarization regarding the whole test models.

For the experiments conducted and reported in this section, the tagging accuracy is defined as the ratio of the correctly tagged words to the total number of words. Every tagged word in the dataset has only two possibilities (the tag is correct or incorrect) when compared to the correct tagging. The tagging accuracies of the words has been calculated in the following formula:

$$\boldsymbol{Accuracy}\ (\%) = \frac{\textbf{Correctly Tagged words}}{\textbf{Total no. of words in the evaluation set}} * \mathbf{100}$$

4.2 Experimental Results

The following table (Table 4) shows the results of tagging using NB and the KNN.

Table 4: The Experimental Results for the 20,000 Tokens Dataset

Features	NB	KNN k=1	KNN k=5	KNN k=10
1 Prefix	51.67%	53.63%	56.93%	**58.45%**
2 Prefixes	61.15%	63.95%	**65.82%**	65.32%
3 Prefixes	**75.10%**	68.32%	68.42%	67.34%
4 Prefixes	**82.22%**	69.94%	68.17%	66.45%
5 Prefixes	**84.48%**	69.25%	66.21%	64.24%
1 Suffix	47.69%	45.63%	48.97%	**49.12%**
2 Suffixes	55.35%	54.37%	54.72%	**55.60%**
3 Suffixes	**67.39%**	65.57%	66.80%	65.82%
4 Suffixes	**77.46%**	70.58%	69.84%	66.75%
5 Suffixes	**81.34%**	70.48%	68.61%	65.91%
1 Circumfix	57.32%	66.45%	**70.09%**	69.16%
2 Circumfixes	70.68%	**83.40%**	82.47%	80.45%
3 Circumfixes	**85.31%**	85.27%	79.62%	75.39%
4 Circumfixes	**86.20%**	81.09%	77.95%	72.45%
5 Circumfixes	**88.26%**	83.79%	74.46%	68.47%

The prediction of the POS tag is provided by using the prefix, suffix, and circumfix. The size of the data set is 20,000 tokens and is divided into 10% testing data and 90% training data. The best way to predict the POS tag is by using five circumfixes for NB and three circumfixes for KNN.

The following Table 5 shows the results of tagging using NB and KNN. The prediction of the POS tag is provided using the prefix, suffix, and circumfix. The size of the dataset is 40,000 tokens and is divided into 10% testing data and 90% training data. The best way to predict the POS tag is by using five circumfixes for NB and two circumfixes for KNN.

Table 5: The Experimental Results for the 40,000 Tokens Dataset

Features	NB	KNN k=1	KNN k=5	KNN k=10
1 Prefix	60.68%	31.99%	30.17%	30.00%
2 Prefixes	68.71%	59.81%	60.51%	57.80%
3 Prefixes	78.92%	59.50%	60.98%	54.77%
4 Prefixes	83.87%	57.12%	58.99%	51.15%
5 Prefixes	86.59%	55.18%	55.81%	45.77%
1 Suffix	58.04%	42.03%	19.55%	32.91%
2 Suffixes	64.59%	43.56%	33.28%	48.82%
3 Suffixes	73.68%	57.07%	49.16%	55.91%
4 Suffixes	81.28%	55.86%	45.94%	54.35%

5 Suffixes	83.99%	53.97%	42.66%	49.94%
1 Circumfix	65.53%	44.46%	35.14%	52.07%
2 Circumfixes	76.04%	74.12%	64.98%	63.81%
3 Circumfixes	86.05%	72.06%	58.60%	55.98%
4 Circumfixes	88.87%	70.48%	56.27%	52.00%
5 Circumfixes	89.11%	70.06%	56.26%	51.82%

V. CONCLUSION

POS tagging is one of the most critical tasks in NLP. In this study, we compare and contrast the affix-based POS tagging strategies and study the influence of each strategy on the tagging performance of the Malay POS tagging models. We conducted a series of experiments using two versions of the Malay language corpus: 20,000 tokens and 40,000 tokens. The results show that the tagging models satisfactorily when the feature pattern used for the tagging is a circumfix rather than a prefix or suffix. Besides, we can conclude that the larger the data set is for the experiments, the better the results are when using the hidden Markov model and the Naïve Bayes model. On the other hand, the k-nearest neighbor model did not achieve good results when using 40,000 tokens.

ACKNOWLEDGMENT:

This research work is supported by Universiti Kebangsaan Malaysia Key Research Area research grants, Project KRA-2018-014, and KRA-2018-015.

REFERENCES

1. Abdullah, H. (1972). The morphology of Malay.
2. Abdullah, I. H., Ahmad, Z., Ghani, R. A., Jalaludin, N. H., & Aman, I. (2004). A Practical Grammar of Malay--a corpus based approach to the description of Malay: extending the possibilities for endless and lifelong language learning. The National University of Singapore.
3. Al-Adhaileh, M. H., Kong, T. E., & Melamed, I. D. (n.d.). Malay-English Bitext Mapping and Alignment Using SIMR/GSA Algorithms.
4. Alhawiti, D. K. M. (2014). Natural Language Processing and its Use in Education. Computer Science Department, Faculty of Computers and Information Technology, Tabuk University, Tabuk, Saudi Arabia.
5. Allen, J. (1995). Natural Language Understanding. Pearson.
6. Ariffin, S. N. A. N., & Tiun, S. (2018). Part-of-Speech Tagger for Malay Social Media Texts. GEMA Online®Journal of Language Studies, 18(4).
7. Baldwin, T., & Awab, S. (2006). Open source corpus analysis tools for Malay. Proceedings of the 5th International Conference on Language Resources and Evaluation, LREC 2006.
8. Benajiba, Y., Diab, M., & Rosso, P. (2008). Arabic Named Entity Recognition using optimized feature sets. EMNLP 2008 - 2008 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference: A Meeting of SIGDAT, a Special Interest Group of the ACL. https://doi.org/10.3115/1613715.1613755
9. Brants, T. (2000). TnT: a statistical part-of-speech tagger. Proceedings of the Sixth Conference on Applied Natural Language Processing, 224–231.
10. Brill, E. (1995). Transformation-Based Error-Driven Learning and Natural Language. Computational Linguistics.
11. Chowdhury, G. G. (2003). Natural language processing. Annual Review of Information Science and Technology.
12. El-Imam, Y. A., & Don, Z. M. (2005). Improved synthesis of standard Malay. Proceedings of the Seventh IASTED International Conference on Signal and Image Processing, SIP 2005.
13. Feldman, A. (2006). Portable Language Technology: a Resource-Light Approach to Morpho-Syntactic Tagging. Ohio State University.

14. Flanagan, J., Rabiner, L., & Schafer, R. (1974). Speech synthesis by concatenation of formant encoded words. Google Patents.

15. Giménez, J., & Màrquez, L. (2006). Technical Manual v1. 3. Universitat Politecnica de Catalunya, Barcelona.

16. Jurafsky, D., & Martin, J. (2014). Speech and Language Processing. In Speech and Language Processing.

17. Knowles, G., & Don, Z. M. (2003). Tagging a corpus of Malay texts, and coping with 'syntactic drift.' Proceedings of the Corpus Linguistics 2003 Conference, 422–428.

18. Knowles, G., & Don, Z. M. (2004). The notion of a "lemma": Headwords, roots, and lexical sets. International Journal of Corpus Linguistics, 9(1), 69–81.

19. Lindberg, J. (1960). Handling Lexicalised Phrases for Natural Language Processing. Stockholm University.

20. Loftsson, H. (2008). Tagging Icelandic text: A linguistic rule-based approach. Nordic Journal of Linguistics. https://doi.org/10.1017/S03325S6508001820

21. Marques, N. C., & Lopes, G. P. (2001). Tagging with small training corpora. International Symposium on Intelligent Data Analysis, 63–72.

22. McCallum, A., Nigam, K., & others. (1998). A comparison of event models for naive Bayes text classification. AAAI-98 Workshop on Learning for Text Categorization, 752(1), 41–48.

23. Mohamed, E. S. (2010). Orthographic enrichment for Arabic grammatical analysis. Indiana University, United States, Indiana.

24. Mohamed, H., Omar, N., & Ab Aziz, M. J. (2011). Statistical Malay part-of-speech (POS) tagger using the Hidden Markov approach. 2011 International Conference on Semantic Technology and Information Retrieval, STAIR 2011. https://doi.org/10.1109/STAIR.2011.5995794

25. Navigli, R. (2009). Word sense disambiguation: A survey. ACM Computing Surveys. https://doi.org/10.1145/1459352.1459355

26. Qin, I. W., & Schuurmans, D. (2005). Improved estimation for unsupervised part-of-speech tagging. Proceedings of 2005 IEEE International Conference on Natural Language Processing and Knowledge Engineering, IEEE NLP-KE'05. https://doi.org/10.1109/NLPKE.2005.1598738

27. Quah, C. K., Bond, F., & Yamazaki, T. (2001). Design and construction of a machine-tractable Malay- English lexicon. Asialex 2001 Proceedings.

28. Ranaivo-Malancon, B. (2005). Malay lexical analysis through the corpus-based approach. Proceedings of International Conference of Malay Lexicology and Lexicography (PALMA), Kuala Lumpur, Malaysia.

29. Schmid, H., & Laws, F. (2008). Estimation of conditional probabilities with decision trees and an application to fine-grained POS tagging. Cooling 2008 - 22nd International Conference on Computational Linguistics, Proceedings of the Conference. https://doi.org/10.3115/1599081.1599179

30. Simmons, R. F., Klein, S., & McConlogue, K. (1962). Toward the synthesis of human language behavior. Behavioral Science, 7(3), 402.

31. Tan, Y. L. (2003). A minimally-supervised Malay affix learner. Proceedings of the Class of 2003 Senior Conference, Computer Science Department, Swarthmore College.

32. Tufis, D., & Mason, O. (1998). Tagging Romanian texts: a case study for qtag, a language-independent probabilistic tagger. Proceedings of the First International Conference on Language Resources and Evaluation (LREC). https://doi.org/10.1.1.33.3453

33. Zamin, N., Oxley, A., Bakar, Z. A., & Farhan, S. A. (2012). A lazy man's way to part-of-speech tagging. Pacific Rim Knowledge Acquisition Workshop, 106–117.

34. Zuraidah, M. D. (2010). Processing natural Malay texts: A data-driven approach. Trames.