The Development of Malaysian Corpus of Financial English (MaCFE)

Roslan Sadjirin
roslancs@pahang.uitm.edu.my
Universiti Teknologi MARA, Cawangan Pahang,
Pahang, Malaysia

Roslina Abdul Aziz
leenaziz@pahang.uitm.edu.my
Universiti Teknologi MARA, Cawangan Pahang,
Pahang, Malaysia

Noli Maishara Nordin
nolinordin@pahang.uitm.edu.my
Universiti Teknologi MARA, Cawangan Pahang,
Pahang, Malaysia

Mohd Rozaidi Ismail
rozaidi@pahang.uitm.edu.my
Universiti Teknologi MARA, Cawangan Pahang,
Pahang, Malaysia

Norzie Diana Baharum
norziediana@pahang.uitm.edu.my
Universiti Teknologi MARA, Cawangan Pahang,
Pahang, Malaysia

ABSTRACT

This paper presents the processes involved in the design and development of the Malaysian Corpus of Financial English (MaCFE); a specialized corpus containing a wide range of online/internet documents (i.e. communiqué) from various financial institutions in Malaysia. It describes in detail the processes involved in the collection and selection of data and preprocessing of raw data, which includes data digitizing, cleansing and tagging. This paper also introduces the user interface for MaCFE with its built-in linguistic analysis features. MaCFE was designed and developed with the intention of providing corpus linguistic researchers with the avenue to explore the field and for ESP/EAP practitioners in Malaysia, as the resources for the development of local-based ESP/EAP curriculum and teaching and learning materials. It would also serve as a learning avenue for future financial professionals in their training. MaCFE corpus has approximately 4.3 million words from 1472 electronic documents retrieved from banks and financial institutions' official websites. At present, users can make queries to the MaCFE database using its built-in concordancer. In the future, its language-data-processing facilities will be expanded to include tools for keyword, wordlist and word collocations queries.

Keywords: corpus linguistics; specialized corpus; financial English; ESP; EAP

INTRODUCTION

A corpus is a subset of electronic texts library developed on a large scale, which contains extensive collections of transcribed utterances or written texts (McEnery & Hardie, 2011). It is built according to explicit design criteria for a specific purpose which not only serves as a

basis for linguistics analysis, improves description and uses of languages, but is also used in various applications including processing of natural language by computer and understanding how to learn or teach a language (Atkins, Clear & Ostler, 1991; Bennett, 2010; Kennedy, 1998).

Various corpora have been compiled and designed to serve different purposes, which in turn influences the design, size, and structure of the individual corpus. General corpora like the British National Corpus (BNC), the Longman Spoken and Written English Corpus (LSWEC) or the American National Corpus (ANC), which were developed to be representative of language in general, are larger in size (e.g. BNC-100 million words, ANC-100 million words, LSWEC- 40 million words) and contain a wide variety of texts and text types, both spoken and written. Specialized corpora for instance the Michigan Corpus of Academic Spoken English (MICASE), the Hong Kong Engineering Corpus (HKEC) and the Hong Kong Financial Services Corpus (HKFSC), which were assembled to answer very specific questions or to represent the language of specific discourse communities are usually smaller than generalized corpora. They may contain only one register and very specific texts, text types, moves or functions. MICASE for instance, consists of only spoken events in a university setting, while HKFSC and HKEC as indicated by the names are collections of texts or publications from financial services and engineering bodies in Hong Kong. Specialized corpora are often used in the Language for Specific Purposes (LSP) settings, hence they are most useful in the teaching of ESP/EAP, such as the teaching of discipline-specific, genre-specific or rhetorical-specific reading or writing skills. Nevertheless, they are not easily accessible. To date, only a small number of specialized corpora can be publicly accessed online (Nesselhauf, 2005).

MaCFE is the first specialized corpus to contribute to the building of a larger and more comprehensive Malaysian Corpus of English for Specific Purposes or MaCESP. The research team envisions building several specialized corpora representing various major industries in Malaysia including for example, financial, business, engineering, hospitality, and/or law. These specialized corpora will then contribute to MaCESP (Roslina et al., 2015) The decision to choose financial English as the first in the series of specialized corpora compiled for MaCESP, was made due to the importance of English in the Malaysian finance sector. English has been and still is a language of choice for the financial sector in this country, even after the enactment of The National Language Act 1963/67. The Act only applies to the public sector (Ain Nadzimah & Rosli, 2002), granting the private sector especially banking and finance the freedom to operate in two languages; Malay and English (Ain Nadzimah & Rosli, 2002). The integration of international and local markets and businesses has also created a multicultural and multilingual finance community in Malaysia, where the use of English has become indispensable. It is common for communication in this sector; both spoken and written, to be conducted in English. Hence, the ability to read, write and communicate in English is pivotal for finance professionals. MaCFE, apart from being a tool for researchers and language practitioners, is also a resource for finance professionals in Malaysia in developing their professional communication competency in English.

This paper aims to present in detail the design and development process of MaCFE and at the same time to introduce the MaCFE prototype.

CORPORA DEVELOPMENT AND CORPUS-BASED STUDIES IN MALAYSIA

Corpora development and corpus-based studies in Malaysia are still very new but steadily growing in momentum. A bibliographic analysis on Malaysian corpus research by Siti Aeisha and Hajar (2014) provides a fundamental understanding of corpus studies in the country. Studies on corpus in Malaysia according to Siti Aeisha and Hajar (2014), are mainly based on

five themes: English language use in Malaysia, Malaysian English learner language, Malaysian textbook content, Malay language and lexicography and corpora development (p. 19). The bibliographic analysis of 42 published articles discusses the use and/or the development of Malaysian based corpora such as the corpus of Malaysian English (ME) short stories, Malaysian English Newspaper Corpus (MEN-Corpus), English of Malaysian School Students (EMAS), Corpus Archive of Learner English in Sabah-Sarawak (CALES), Business and Management English Language Learner Corpus (BMELC), the Dewan Bahasa dan Pustaka (DBP) Malay corpus, the Malay Practical Grammar Corpus (MPGC), the MaLay LEXicon (MALEX), Malaysian Corpus of Learner English (MACLE), Corpus of Malaysian English (COMEL), Malaysian International Corpus of English (ICE Malaysia) and the Engineering Lecture Corpus (ELC). This paper shall provide an overview of the major corpora in Malaysia and review some of the works conducted thus far using these corpora.

The first corpus developed in Malaysia and by far the largest Malay corpus with 128 million words is the DBP corpus. It is extensively used to develop the other Malay corpora including the on-going MPGC and MALEX (Hajar, 2014). Imran, Zaharani, Rusdi, Nor Hashimah and Idris's (2004) on-going project on the Malay Practical Grammar Corpus (MPGC) or better known as the DBP-UKM corpus is a collection of Malay newspapers, magazines and books compiled from the DBP database. With an initial size of 5 million words, the project aims at examining aspects of Malay grammar derived from its authentic use in printed texts which, as suggested by the researchers, can be utilized for various types of language analysis like vocabulary, lexical, grammar and discourse as well as language teaching and testing. The corpus has so far been used to analyse the prepositions *antara*/*di antara*, *adalah*/*ialah* and the pronouns *ia*/*ianya*, among many others. Another Malay corpus that utilizes the extensive data from the DBP database is the MaLay LEXicon (MALEX) developed by Zuraidah (2010). Using novels, newspapers, speeches of the 4th Malaysian Prime Minister, Dr. Mahathir Mohamad and academic texts, this approximately 7,120,000-Malay-word collection is used as a "relational database" where information "for grammatical tagging, stemming and lemmatisation, parsing, and for generating phonological representations" (p. 90) is made accessible for future research. MALEX includes the spoken data in its word collection and this project is seen as having great potential for translation field as the work will later be extended to include semantic (Zuraidah & Knowles, n.d).

The Malaysian Corpus of Learner English (MACLE) was developed under the supervision of a team of researchers; Knowles, Zuraidah, Jariah, Rajeswary, Janet, Sathia, Asha and Su'ad from the University of Malaya, Kuala Lumpur (Knowles et al., 2006). It is a mono-generic corpus consisting of argumentative essays written by second to fourth year undergraduates at the University of Malaya between 2004 and 2005. At present, MACLE is the largest learner corpora in Malaysia with approximately 800,000 word tokens. MACLE was developed to represent the Malaysian learner English in the International Corpus of Learner English (ICLE) (Granger, 1998; Granger, 2002). It was, therefore, designed following the criteria set for ICLE. Most recently, the data from this corpus were used in a study by Zuraidah and Sridevi (2017), which analysed the use of conjunctive adjuncts in the subset of 54 argumentative essays written by Law students. The study identified 307 conjunctive adjuncts, which were grouped under three main categories namely (i) Elaboration, (ii) Extension and (iii) Enhancement, following Halliday and Matthiesen's (2014) framework. Other studies utilizing the data from MACLE include Roslina and Zuraidah (2013) on omission of *BE* and Roslina and Zuraidah (2014) on *BE* overgeneration.

The English of Malaysian School Students (EMAS) corpus is an "untagged and unedited learner corpus" (Arshad, 2004, p. 44) which houses the written and spoken language production of Malaysian Primary 5, Form 2 and Form 4 students. Developed by Arshad (2002), this corpus has been extensively used by the research community with various

findings contributing to the studies of English teaching and learning as a second language in Malaysia. An error analysis study by Arshad and Hawanum (2010) for instance made use of the data from this corpus to investigate the use of auxiliary *BE* in the essays written by Malaysian Primary 5 students. The study found many instances where students overgeneralized the use of *was* to show past tense and were unable to differentiate between the use of *BE* as an auxiliary and as a main verb. Besides that, Rafidah (2013) in her investigation of the use of six phrasal verbs with particle *UP* by Malaysian ESL learners had also made use of EMAS corpus. The Malaysian learners' use of phrasal verbs was compared to that of native speakers' from Bank of English (BoE) corpus. The findings revealed that wrong usage of common phrasal verbs (e.g. pick up, wake up, get up) has strong association with the learners' lexical knowledge, their awareness of common collocates, familiarity with the context of use and their mother tongue. The appropriateness in the use of phrasal verbs was also found to improve over time, suggesting that learners had benefited from longer exposure to the target language. EMAS was also utilized by Zarifi and Jayakaran (2014) in a corpus-based analysis of the creativity and unnaturalness in the use of phrasal verbs among Malaysian ESL learners. The acceptability of the phrasal verbs used or created by learners was judged with the help of dictionaries and those without dictionary entry were judged against BNC. Learners were found to use phrasal verbs quite frequently, however, some of the phrasal verbs created by the learners appeared unnatural. In discussing the pedagogical implications of the study, the researchers suggested that material developers and teachers should emphasize on distinguishing the semantic functions of every single particle and the way to combine them with various lexical verbs.

The Corpus Archive of Learner English in Sabah-Sarawak (CALES) developed by Botley and Doreen (2007) is a complementary corpus for the University of Malaya's MACLE. As of 2007, the corpus contains around 400,000-word argumentative essays produced by diploma and degree students taking English proficiency courses at four public universities in East Malaysia namely UiTM Sarawak, UiTM Sabah, Universiti Malaysia Sarawak (UNIMAS) and Universiti Malaysia Sabah (UMS). The learner corpus is closely modelled after the International Corpus of Learner English (ICLE) (Granger, 1998; Granger, 2002). Among the studies utilizing the corpus archive is the one by Botley and Doreen (2007) which analysed spelling errors in the 281 essays selected from the corpus. The errors were grouped according to the framework developed by James (1998 as cited in Botley et al., 2007) which sees mechanical errors like doubling (*abbuse*), omission (*vacum*) and mis-ordering (*frobidden*), mis-spellings (*prostitude*, *sofisticated*), interlingual mis-encoding (*accaunting*, *karier*) in the selection of CALES texts.

In addition to the corpora reviewed, there are also the Malaysian Corpus of Students' Argumentative Writing – MCSAW[1] developed by Jayakaran and Rezvani Kalajahi (2013) and the Written English Corpus for Malay ESL Learners (WECMEL), a collection of 470,000 word argumentative essays produced by Universiti Teknologi MARA pre-Law students (Shazila & Noorzan, 2013).

The literature proves that corpus-based research is growing synchronously with corpora development in the country. This is especially true for the Malaysian learner corpora. However, development of specialized corpus in the country has been rather limited. So far, only one specialized corpus containing data from Malaysia has been developed i.e. Corpus of Malaysia Memoranda of Understanding (MoA), which contains legal documents compiled by Su'ad (1999, 2003). Considering the importance of specialized corpora in ESP/EAP contexts and the need to provide language instructors and learners with data relevant to the local setting, the Malaysian Corpus of Financial English (MaCFE) was developed.

[1] For further reading on data-driven studies utilizing MCSAW see Janaki, Chithra and Karen (2013), Darina, Juliana and Norin (2013) and Mohamed Ismail, Begi and Vaseghi (2013).

DESIGN AND DEVELOPMENT OF MaCFE

MaCFE is designed and developed following the current methodology of corpus linguistics. In its construction, the research team has adhered as closely as possible to the corpus design principles posited by Sinclair (2004), which are summarized below:

1. The contents of a corpus should be selected according to their function in the community in which they arise.
2. The corpus should be as representative as possible of the chosen language.
3. Only components in the corpus that are designed to be independently contrasted are contrasted.
4. Criteria determining the structure of the corpus are small in number, separate from each other, and efficient at delineating a corpus that is representative.
5. Any information about a text is stored separately from the plain text and only merged when needed.
6. Samples of language for the corpus, whenever possible, consist of entire texts.
7. The design and composition of the corpus are fully documented with full justifications.
8. The corpus design includes, as target notions, representativeness, and balance.
9. The control of subject matter in the corpus is imposed by the use of external, and not internal, criteria.
10. The corpus aims for homogeneity in its components while maintaining adequate coverage, and rogue texts should be avoided.

(cited in Warren, 2010, p. 170)

In addition, the work has also benefitted from previous practices of specialized corpus building. Much of the design framework especially in data compilation (i.e. setting external criteria and text categories) generally follows the framework established by Warren (2010) in building HKFSC. Nevertheless, some adjustments had to be made on the design whenever needed, for instance the text categories finalized in MaCFE did not include some of the text categories used for the development of HKFSC due to issues on confidentiality and accessibility.

Furthermore, MaCFE has also adapted the Aksan and Aksan (2009) workflow packages. The corpus development is divided into 4 major processes namely; (1) data collection and selection, (2) data preprocessing which includes data digitizing, data cleansing, part-of-speech (POS) and meta-linguistic tagging, (3) user interface, and (4) text and linguistic analysis. This section briefly discusses these processes, and Fig. 1 depicts the framework of the MaCFE design.

eISSN: 2550-2131
ISSN: 1675-8021

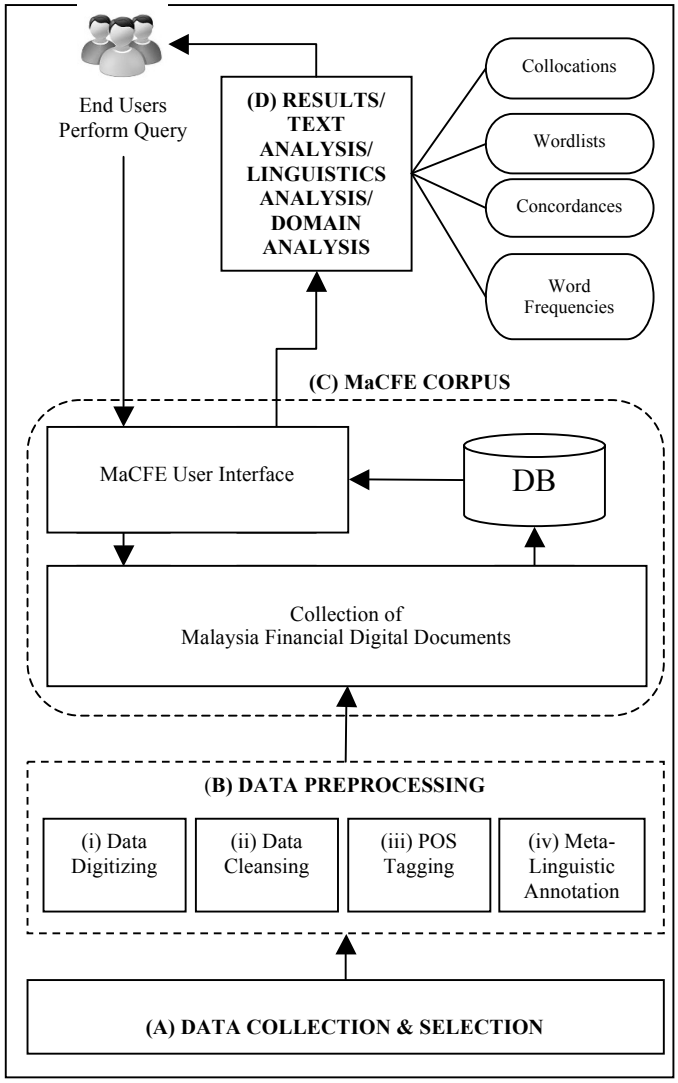

FIGURE 1. MaCFE design framework

A corpus is designed to constitute a representative sample of a defined language type (Atkins et al., 1991). Therefore, data selection is key to the successful design and development of the specialized corpus. As mentioned earlier MaCFE has adopted the text categories of the HKFSC (Warren, 2010). In determining that the range of text types is representative of the English used by professionals in the financial sectors in Hong Kong, Warren (2010) has sought expert advice of professional bodies, government departments, private sectors as well as individual professionals from the financial service sector. Based on the experts' advice, HKFSC comprises of 26 text types, all of which characterize the language read and written by financial professionals in Hong Kong. Most of the text types also typify the written language of financial institutions bodies in Malaysia and some adjustment had to be made to the text categories to suit the Malaysian finance situation for instance the descriptions of the products offered by the banking institutions (insurance, investment, credit cards, etc.) are categorized into two; Islamic and Conventional. Several categories (i.e. Code of Practice/Ethics, Circulars, Prospectus, Rules, Standards and Result Announcement) were merged into other categories for example Prospectus was merged with Annual Report and Circulars was categorized under Corporate Announcements. In addition, 4 other text types, which are not available in HKFSC, namely Advertisement, Corporate Social Responsibility (CSR) Reports, Terms and Conditions, Media Coverage, and Publication are

included in MaCFE. These additional text types are available in all banking institutions involved in this study and considered important as they are means for the banks to communicate with their clients (e.g. Publication), to disclose information to the general public, internal and external stakeholders (e.g. Media Coverage, Publication, CSR Reports) and to advertise their products (e.g. Advertisements). Table 1 summarizes the text types for MaCFE.

TABLE 1. Text types for MaCFE

Text Type	Abbreviation	Text Type	Abbreviation
Advertisements	ad	Interim Reports / Quarterly Reports	ir
Agreements	agr	Media Releases	mr
Annual Reports	ar	Media Coverage	mc
Brochures	br	Ordinance	ord
Bank Service Charges	bsc	Policies	pol
Corporate Announcement	ca	Principles	pri
Corporate Social Responsibility Reports	csr	Product Description_Conventional	pdc
Financial Reports	fr	Product Description_Islamic	pdi
Fund Descriptions	fd	Publications	pub
Fund Reports	fr	Speeches	spc
Guidelines	gl	Terms & Conditions	tc
General Meetings	gm		

Adapted from Warren (2010)

Malaysia operates a dual-banking system; conventional banking system operating in tandem with Islamic banking system. Since the enactment of the Islamic Banking Act 1983 and the establishment of Malaysia's first Islamic Bank, a significant number of full-fledged Islamic banks have been established in the country including Bank Islam Malaysia Berhad and Bank Muamalat Malaysia Berhad. In recent years, Malaysia has also seen the increase of local conventional banks establishing Islamic subsidiaries offering various products and services complying with Sharia Law (e.g. Public Islamic Bank Berhad, CIMB Islamic Bank Berhad, RHB Islamic Bank Berhad). The liberalization of the Islamic financial system and government-facilitated business environment have also attracted a number of foreign-owned financial institutions to set their Islamic banks and subsidiaries in the country (e.g. Al Rajhi Banking and Investment Corporation, OCBC Al-Amin Bank Berhad, Standard Chartered Saadiq Berhad). In fact, Islamic banking has become an integral part of the financial system in Malaysia that at present, Malaysia's Islamic banking assets have reached USD65.6 billion with an average growth rate of 18-20% annually (Bank Negara, 2017). Due to this development, the data from local as well as international Islamic financial entities are gathered for the development of MaCFE. The final release of MaCFE will cover four major categories of finance institutions; Local Islamic Bank, Foreign Islamic Bank, Local Conventional Bank and Foreign Conventional Bank as displayed in Fig. 2.

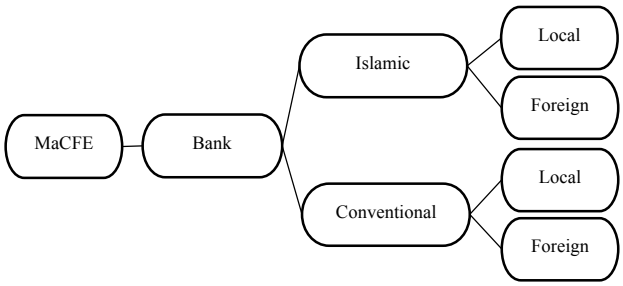

FIGURE 2. Decision tree for document selection

Presently, 1472 electronic documents related to the Malaysian financial domain have been gathered and compiled amounting to a total number of approximately 4,373,230 million tokens. These electronic documents were retrieved and collected from banks' official websites, which are accessible via the public domain.

DATA PREPROCESSING

After the targeted data were selected and collected, preprocessing steps were applied. According to Zimmermann and Weißgerber (2004), preprocessing has a direct impact on the quality of the results returned by an analysis. MaCFE underwent four stages of data preprocessing; (i) data digitizing, (ii) data cleansing, (iii) part-of-speech tagging, and (iv) meta-linguistic annotation/markup. Each of the stages is explained in the following sub-sections.

1. Data Digitizing

In order to transform the collected data into machine readable texts and integrate them with MaCFE's user interface, all the documents compiled have to be converted into text files. Text file format is a human-readable sequence of characters, which can be encoded into machine readable formats. Each converted file will be renamed as follows:

a. *Naming convention for bank documents:*

```
{Bank}{Conventional|Islamic}
{Local|Foreign}{BankName}
{TypeOfDocument}{YearPublished}
{SequenceOfDocument}+{Month}+
```

Example: BCFHSBC_ar20101Dec

The plus (+) sign in the naming convention for `{SequenceOfDocument}` and `{Month}` indicates that encoding is optional, because some documents only provide the year of publication and do not include the sequence and month of publication. Table 2 shows the text types and the respective codes assigned for document naming convention and Table 3 presents the examples of documents in the MaCFE text collection.

TABLE 2. Type and code for document naming convention

Type	Example Code
Bank	B
Insurance	I
Conventional	C
Islamic	Is
Local	L
Foreign	F
BankName	HSBC
TypeOfDocument	ar
YearPublished	2010
SequenceOfDocument	1
Month	Dec

TABLE 3. Samples of MaCFE text collection

Islamic Banking	Conventional Banking
• BIsFHSBC_mr20111Apr	• BCFHSBC_ar20101Dec
• BIsFHSBC_ar20101Dec	• BCFHSBC_ar20102Dec
• BIsFHSBC_ar20102Dec	• BCFHSBC_ar20111Dec
• BIsFHSBC_ar20111Dec	• BCFHSBC_ar20112Dec
• BIsFHSBC_ar20112Dec	• BCFHSBC_ar20121Dec

Islamic Banking	Conventional Banking
• BIsFHSBC_ar20121Dec	• BCFHSBC_ar20122Dec
• BIsFHSBC_ar20122Dec	• BCFHSBC_ar20131Dec
• BIsFHSBC_ar20131Dec	• BCFHSBC_ar20132Dec
• BIsFHSBC_ar20132Dec	• BCFHSBC_ar20141Dec
• BIsFHSBC_ar20141Dec	• BCFHSBC_ar20142Dec
• BIsFHSBC_ar20142Dec	• BCFHSBC_ir20101Jun

2. Data Cleansing

The next stage in preprocessing is data cleansing. Data cleansing, also known as data cleaning or data scrubbing, involves the process of removing or eliminating noise from the data, which includes tables, images and special characters (refer to Table 4 for examples of special characters). According to Chu et al. (2016), failure in data cleansing leads to inaccurate analysis and unreliable decision. As an example, tables and images need to be removed as they contain isolated terms and figures that would be counted by lexical analysis software in its overall analysis, thus affecting the overall statistical findings of wordlists and concordances. In general, too much noise in the datasets might render the data unfit and unsuitable for data analytics. As for MaCFE, there are four mandatory data cleansing procedures required, which are:

i. Remove/eliminate tables
ii. Remove/eliminate images
iii. Correct misspelling
iv. Remove/eliminate special characters (e.g. ^ % #)

Tables and images were automatically removed during data digitizing process. This process involves converting the data sources into text files using PDF Foxit Reader software. During the conversion, tables and images were simultaneously removed. Spelling correction was performed with the aid of Microsoft Word spelling checker, which was used to identify and correct misspelled words. Finally, special characters were removed automatically using RapidMiner Studio Educational (7.5.001) Text Processing Package by utilizing an algorithm as shown in Figure 3.

Table 4 displays some examples of special characters that need to be removed from the text collection. The algorithm for removing the special characters is presented in Fig. 3.

TABLE 4. Examples of special characters

Code							
{	å	É	Ü	ñ	«	¦	Î
|	ç	æ	ø	Ñ	»	−	Ï
}	ê	Æ	£	ª	Ò	+	¦
~	ë	ô	Ø	º	õ	¤	
Ç	è	ö	×	¿	Á	ð	¯
ü	ï	ò	ƒ	®	Â	Đ	Ì
é	î	û	á	¬	À	Ê	¯
â	ì	ù	í	½	¢	Ë	Ó
ä	Ä	ÿ	ó	¼	¥	È	ß
à	Å	Ö	ú	¡	©	ı	Ô

```
BEGIN
READ document
WHILE document <> NULL
   READ line_of_text
      WHILE line_of_text <> NULL
          PERFORM TOKENIZATION ON line_of_text
                IF token IS special_characters
                   THEN REMOVE token FROM document
                ELSE WRITE token INTO document
                ENDIF
             READ line_of_text
        ENDOFWHILE
   READ document
ENDOF WHILE
END
```

FIGURE 3. Algorithm to remove non-letters and special characters

3. Part-of-Speech (POS) Tagging

The next stage of preprocessing is part-of-speech (POS) tagging. POS tagging is a basic form of syntactic analysis (Gimpel et al., 2011) and according to Leech (1997) is the most frequently used form of annotation. POS tagging involves assigning each lexical unit in the datasets a code to indicate its part of speech, for example NNP for singular proper noun, RB for adverb or JJ for adjective. Information regarding the parts of speech is primary in increasing the specificity of data retrieval and an important foundation for further forms of analysis such as syntactic parsing and semantic field annotation (McEnery & Hardie, 2011). Additionally, it could also contribute to various computational linguistic applications.

Nevertheless, to manually POS tag each lexical unit in a large corpus is time-consuming and a tedious process. Therefore, MaCFE was tagged using an automated POS tagger developed by Toutanova and Manning (2000) at Stanford University. The tagger was further improved by Toutanova, Klein and Manning (2003). The Tautanova and Manning's POS tagger can be retrieved and downloaded from https://nlp.stanford.edu/software/tagger.shtml. Table 5 illustrates the encoding of POS tagsets and the respective descriptions, which are based on the tagsets of the Penn Treebank (Marcus, Santorini & Marcinkiewicz, 1993). The complete Penn Treebank tagsets can be viewed in Santorini (1990).

TABLE 5. Part-of-speech tagsets used in coding MaCFE

Tag	Description	Tag	Description
CC	Coordinating conjunction	PRP$	Possessive pronoun
CD	Cardinal number	RB	Adverb
DT	Determiner	RBR	Adverb, comparative
EX	Existential there	RBS	Adverb, superlative
FW	Foreign word	RP	Particle
IN	Preposition or subordinating conjunction	SYM	Symbol
JJ	Adjective	TO	to
JJR	Adjective, comparative	UH	Interjection
JJS	Adjective, superlative	VB	Verb, base form
LS	List item marker	VBD	Verb, past tense
MD	Modal	VBG	Verb, gerund, or present participle
NN	Noun, singular or mass	VBN	Verb, past participle
NNS	Noun, plural	VBP	Verb, non-3rd person singular present
NNP	Proper noun, singular	VBZ	Verb, 3rd person singular present
NNPS	Proper noun, plural	WDT	Wh-determiner
PDT	Predeterminer	WP	Wh-pronoun
POS	Possessive ending	WP$	Possessive wh-pronoun
PRP	Personal pronoun	WRB	Wh-adverb

(Marcus et al., 1993)

MaCFE is still at the initial stage of development and has yet to be equipped with its own range of text processing facilities. RapidMiner and an in-house stand-alone Java program were employed to generate the wordlist for MaCFE. The wordlist produced would then be used to evaluate the suitability of the texts chosen to represent the financial domain. Table 6 presents the first fifty high frequency words ranked in MaCFE. The wordlist was obtained using the following steps and procedures.

Step 1: In this step, works done by Verma and Gaur (2014) and Shterev (2013) were adapted. At this stage, the RapidMiner Studio Educational (7.5.001) Text Processing Package (see Appendix A for steps taken to generate wordlist performed on RapidMiner) was employed, and the operators utilized for the process are in the following orders:

a. Transform Cases: This operator transforms all characters into lowercase.
b. Tokenize (mode: non-letters): Split text document containing non-letters into single token.
c. Tokenize (mode: linguistic sentences; language: English): Split text document containing linguistics sentences into single word token.
d. Tokenize (mode: linguistic tokens; language: English): Split word token into single character.
e. Tokenize (mode: specify character): Split word token into single character with specified delimiter.
f. Filter Special Characters (Dictionary): Remove special characters (refer to Table 4). Although special characters have been removed during data cleansing, this operation needs to be performed to ensure the texts are free from all possible special characters.
g. Filter Stopwords[2] (English): Remove tokens that are English stopwords (refer to Appendix C for samples).

After performing all the actions in Step 1, a list that contains three tuples, namely *Attribute Name*, *Total Occurrences*, and *Document Occurrences* was produced. The list generated is shown in Table 6. The explanation of each tuple is as follows:

- *Attribute Name*: Contains a set of word tokens extracted from the text collection.
- *Total Occurrences*: Contains the number of occurrences of each token in a whole text collection.
- *Document Occurrences*: Contains the number of document in which the token appeared.

TABLE 6. Wordlist containing 50 most frequent words produced after completing Step 1

Attribute Name	Total Occurrences	Document Occurrences	Attribute Name	Total Occurrences	Document Occurrences
bank	38086	1000	cash	4921	273
financial	20048	754	conditions	4916	421
customer	18418	270	committee	4785	198
group	14801	275	terms	4754	408
account	12738	399	value	4726	339
credit	11134	481	interest	4446	427
risk	10376	383	assets	4427	357
card	8399	176	information	4303	432
management	7167	462	rate	4194	402
million	6839	273	financing	4189	401
growth	6551	418	loss	4023	323
banking	6233	547	due	3965	492
cardholder	6086	106	loans	3880	299
market	5903	480	payment	3818	235
business	5597	545	date	3817	259
year	5572	497	board	3749	223

[2] Refer to Fox (1989) for more details on stopwords.

capital	5489	390	global	3672	431
services	5241	503	including	3629	429
time	5240	420	profit	3548	264
income	5169	404	amount	3493	328

Step 2: The next step is tagging each of the extracted token with its POS tag using the automated POS Tagger developed by Toutanova and Manning (2000) (refer to Appendix B for detailed steps). The list of tokens after POS tagging was performed is shown in Table 7.

TABLE 7. Wordlist after POS tagging process

• bank_NN	• cash_NN
• financial_JJ	• conditions_NNS
• customer_NN	• committee_NN
• group_NN	• terms_NNS
• account_NN	• value_NN
• credit_NN	• interest_NN
• risk_NN	• assets_NNS
• card_NN	• information_NN
• management_NN	• rate_NN
• million_CD	• financing_NN
• growth_NN	• loss_NN
• banking_NN	• due_JJ
• cardholder_NN	• loans_NNS
• market_NN	• payment_NN
• business_NN	• date_NN
• year_NN	• board_NN
• capital_NN	• global_JJ
• services_NNS	• including_VBG
• time_NN	• profit_NN
• income_NN	• amount_NN

4. Meta-Linguistic Annotation/Markup

The final step in data preprocessing is meta-linguistic markup. Meta-linguistic annotation or markup is a process of adding description to the datasets, for instance information about a text; text type, year published, gender of author and etc. For MaCFE, the added markup includes the title of the document, type of document and year of publication. The markup was administered manually using the system presented in Table 8. Basically, common markup system includes <, ! and >, however, for the MaCFE datasets, those symbols were omitted because they are considered as noise.

Typically, a markup system would also involve adding codes to indicate features of the original structure of a text, such as paragraph/sentence/chapter start/end points/page breaks/headings so that a word can be searched together with a markup code. As an example the use of pronoun *we* in the introduction section of scientific journal articles. However, the markup system applied in MaCFE was specifically designed to provide textual information of a text (or the header) i.e. title of document, type of document and year/month of publication. Other elements in the text (paragraph/sentence/chapter start/end points/page breaks/headings) were not annotated. The lack of markup system to set boundaries on paragraphs/sentences in the text would not, however, affect results of wordlist and concordance enquiries, as sentence and paragraph boundaries can still be distinguished through the use of punctuation (full stop) and spaces respectively. Table 8 presents the meta-linguistic markup system used for MaCFE, while Fig. 4 depicts the overview of text documents after performing meta-linguistic markup.

eISSN: 2550-2131
ISSN: 1675-8021

TABLE 8. MaCFE meta-linguistic markup system

			Description
Markup	**Open**	macfeBegin	Indicates the ***beginning*** of metalinguistics of text document.
		macfeTitleBegin	Indicates the ***beginning*** of metalinguistics for document title.
		macfeDocTypeBegin	Indicates the ***beginning*** of metalinguistics for type of document.
		macfeYearBegin	Indicates the ***beginning*** of metalinguistics for type of year of publication.
	Close	macfeEnd	Indicates the ***ending*** of metaliguistics of text document.
		macfeTitleEnd	Indicates the ***ending*** of metalinguistics for document title.
		macfeDocTypeEnd	Indicates the ***ending*** of metalinguistics for type of document.
		macfeYearEnd	Indicates the ***ending*** of metalinguistics for type of year of publication.

```
macfeBegin
                macfeTitleBegin
                                Investing in the Human Spirit
                macfeTitleEnd
                macfeDocTypeBegin
                                AR – Annual Report
                macfeDocTypeEnd
                macfeYearBegin
                                2016
                macfeYearEnd
macfeYearEnd

Profit before tax of $6,9582 million in 2013 down 7 per cent from $7,5182
million in 2012.
Statutory profit before taxation was $6,064 million down 11 per cent. Statutory
profit attributable to ordinary shareholders was $3,989 million ..... ...
```

FIGURE 4. Overview of text with meta-linguistic markup

The MaCFE PROTOTYPE

MACFE is built entirely using the Hypertext Preprocessor or PHP, an open source scripting language for building web applications and MySQL, an open source relational database management system. The PHP codes are executed on the MySQL server to render interaction with users via a web browser (i.e. Internet Explorer, Chrome, Firefox, Safari etc.). The corpus can be accessed at http://learningdistance.org/mycorpus/macfe/.

As shown in Fig. 5 below, the interface has a basic, clean design with a welcome page and only 3 options: 'Home' will bring the user back to the welcome page, 'Login' to start using MaCFE, and 'Register' which the user has to first complete before they can log in into the corpus.

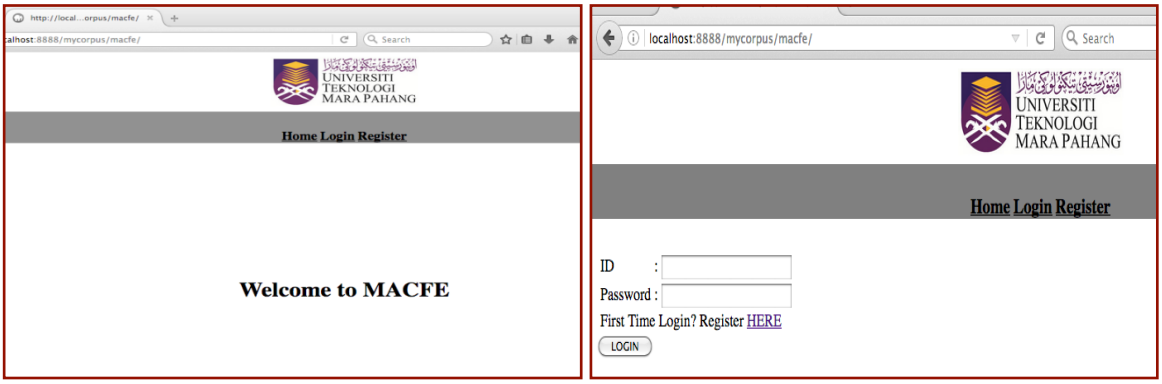

FIGURE 5. MaCFE user-interface

Once logged in, users will be able to make queries to the MaCFE database. Using this prototype, users can generate concordance lines of the MaCFE database. A concordance line is a line of text from a corpus. It can be at the beginning, middle or end of the texts; made up of one sentence, part of a sentence or part of two sentences. To make a query the user enters the target word in word search box: i.e. 'finance' (see Fig. 6). The 'context' option allows the user to decide the number of words before and after the target word. In this case, 12 words before and after the target word 'finance'.

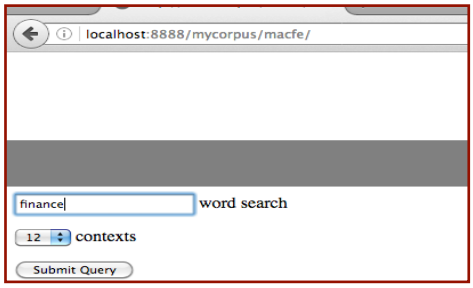

FIGURE 6. Query facility

Each concordance line (see Fig. 7) includes the target word, i.e. the word being studied. The target word is always in the middle of the concordance line. So when users search for a word in a set of concordance lines, they can see its context or the words, which are used before and after it. Note that there are complete sentences, incomplete sentences and also lines showing only part of the sentences.

At the bottom of each query results table, the users have the option of navigating through all the instances of the word 'finance'. The frequency counts of the search enquiry will be displayed at the bottom right of the result table. By analyzing a set of concordance lines, users can analyze how a target word is used in context. They will also be able to analyze other linguistics elements relevant to the target word being studied.

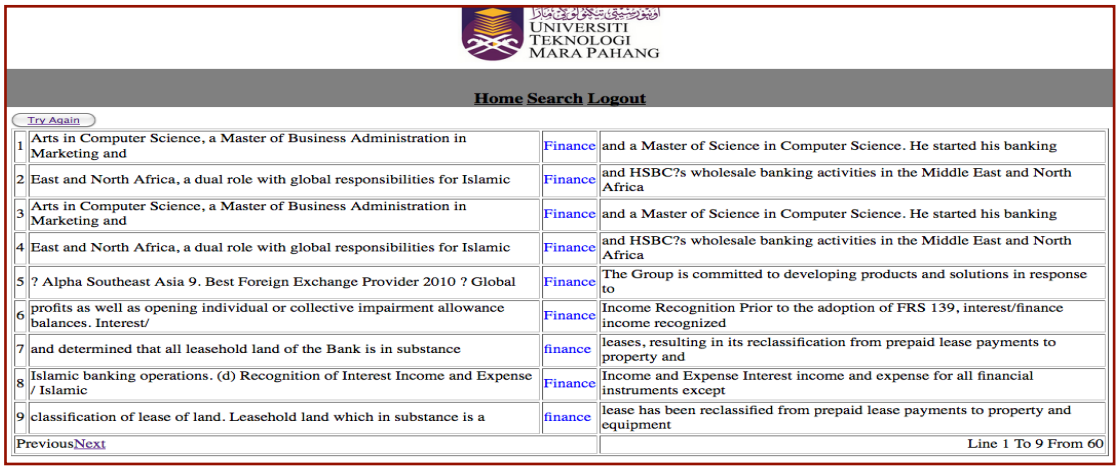

FIGURE 7. Concordance to *finance*

Obviously, the MaCFE prototype is still presently quite basic. Further upgrades and improvements are definitely necessary and are currently underway. At present the research team is adding several other query options, which include enabling users to search according to types of banks, types of documents, year and month. The process is still ongoing and is projected to complete in June 2018. When completed, users are able to narrow their queries to specific areas of the data, depending on the purpose and scope of their analysis.

PROBLEMS ENCOUNTERED IN BUILDING MaCFE

MaCFE as much as possible aims to represent the language written and read by the professionals in the financial sector in Malaysia as well as achieve the desired balance in her language representation. Nevertheless, compiling a large amount of data is not without its challenges. One of the major issues concerning data compilation is obtaining documents that were not accessible to the public. Documents like minutes of 'General Meetings' and 'Agreements' are generally not published online. Gaining access to these documents has proven difficult as most banking institutions were generally reluctant to grant access due to issues of security and confidentiality. As a result, comparatively fewer numbers of these documents were included in MaCFE. Nonetheless, the team had obtained the summary of the 'minutes', which are generally available online. The issue on 'Agreements' was resolved by compiling personal copies of 'Agreements' from clients of the financial institutions involved in this study. The number of 'Agreements' available in MaCFE is relatively small at present, which prompted for future works on MaCFE to include efforts to increase its number.

Data preprocessing of a sizeable corpus like MaCFE also involved laborious and tedious works. Some of the processes were not entirely automated, therefore, requiring some forms of manual labor. Data cleaning process for instance, required for each text to be examined manually to identify misspelled words and special characters. In checking and correcting spelling mistakes, the team had utilized Microsoft Word spell checker, which to an extent had improved the speed of the process. Nonetheless, due to the number and length of the documents involved, the entire process took the research team several months to complete. Removing special characters from the data was also a time-consuming process, as it had to be administered to each individual document. Nevertheless, with the aid of an algorithm system written in Java, the processing time was approximately reduced to half. Each document regardless of the length can be processed in less than 5 minutes, instead of 10 to 20 minutes taken when administered manually.

FUTURE DIRECTION

Future planning of the corpus is to include language data-processing tools that would enable lexical analysis other than concordancing, to be administered using the MaCFE platform. The research team is considering incorporating RapidMiner 7.5.001[3] (Text Processing Package) to generate wordlists, word occurrences, document occurrences and n-grams (bi-gram and tri-gram) and a Java program for the computation of word-form frequency and to generate the association of n-grams. In doing so the team needs to conduct preliminary analysis using these tools in order to gauge the suitability and reliability of the tools. The analysis will also determine if the engine currently employed to operate MaCFE would be able to support these additional software and program. As mentioned earlier, MaCFE utilizes MySQL management system and in order to support future extension, the system has to be upgraded to MySQLi.

The completion of MaCFE has also enabled efforts in designing and developing discipline-specific language materials for EAP/ESP settings. The corpus will be utilized as the reference tool (Yoon, 2011), where samples of authentic language will be extracted to be used in the development of online language modules. The rich collection of authentic language data will be mined to provide authentic phrases, expressions or short passages for the language activities designed. Samples of how the language is used in the forms of concordances will also be available for the learners to analyze. In order to complete the

[3] For the purpose of obtaining the wordlist presented in Table 6, RapidMiner 7.5.001 was administered to the datasets independent of the MaCFE interface.

language activities learners would be required to consult the concordances extracted. This approach to learning language promotes inductive learning (Johns, 1991). Johns (1991) in advocating data-driven learning (DDL), pointed out that the use of corpora can foster inductive learning through learners' active participation in analyzing the language sample. More importantly, the learners will also be presented with authentic language and benefit from the abundance of samples of how the language is actually used in the written communications transpiring in the financial sector in Malaysia. The modules, when complete is hoped to prepare learners with the language skills they would require to function effectively in the financial, business and corporate settings.

Efforts are also underway for the designing and development of training modules for future and current financial professionals in the country. Presently, the research team is preparing to conduct needs analysis on the language needs and requirements of financial professionals serving the local as well as international financial institutions in the country. The findings from the analysis will then be utilized in designing and developing the said modules. Upon completion, the modules will be the first to offer corpus-based training materials that would cater to the needs and requirements of financial professionals in this country and beyond.

CONCLUSION

MaCFE was designed and developed with the intention of providing corpus linguistic researchers and ESP/EAP practitioners in Malaysia, with the avenue to expand research in the field and the resource for the development of local-based ESP/EAP curriculum and teaching and learning materials. Currently, MaCFE has gathered and compiled 1472 electronic financial documents retrieved and collected from banks' official websites. It now contains approximately 4.3 million words. Its final release covers four major categories of finance institutions; Local Islamic Bank, Foreign Islamic Bank, Local Conventional Bank and Foreign Conventional Bank. MaCFE has also employed a computer-based methodology, RapidMiner Studio Educational (7.5.001) Text Processing Package (Shterev, 2013; Verma & Gaur, 2014) to produce its wordlist and the automated POS Tagger (Tautanova & Manning, 2000) to facilitate the team in POS tagging the datasets. The online MaCFE, which was built entirely using the Hypertext Preprocessor or PHP and MySQL, can be freely accessed at http://learningdistance.org/mycorpus/macfe/ via a web browser such as Internet Explorer, Chrome, Firefox, and Safari among others. Upon logged in, users are able to make queries to the MaCFE database and to generate concordance lines of searched items.

MaCFE is seen as a significant language resource not only for linguistic researchers and ESP/EAP practitioners, but also financial professionals in their pursuit to further enhance their professional communicative competence. Thus, it is imperative to inform professionals, researchers and EAP/ESP practitioners of MaCFE's existence and to encourage and promote the specialized corpus as an invaluable resource capable of further enhancing their professional communication, expanding their research horizon and enriching their teaching and learning avenue. In achieving these aims, the research team strives to publish as many works on MaCFE as possible in the local as well as international journals and conferences. At the same time we intend to reach a number of professional bodies, organizations and individual professionals by conducting a series of training workshops on how to use MaCFE as a language learning resource. Finally, it is hoped that the establishment of MaCFE will provide an impetus for the development of other specialized corpora, which consequently would benefit not only researchers and language practitioners, but also professionals and stakeholders in the respective sectors.

eISSN: 2550-2131
ISSN: 1675-8021

ACKNOWLEDGEMENT

This study was funded by Ministry of Higher Education (Malaysia) and Universiti Teknologi MARA (UiTM) under Research Acculturation Grant Scheme (RAGS) (RAGS/1/2014/SSI01/UITM/2).

REFERENCES

Ain Nadzimah Abdullah, Rosli Talif (2002). The Sociolinguistics of Banking: Language Use in Enhancing Capacities and Opportunities. *Pertanika Journal of Social Sciences & Humanities. Vol. 10*(2), 109-116.

Aksan, Y., & Aksan, M. (2009). Building a National Corpus of Turkish: Design and Implementation. *Working Papers in Corpus-Based Linguistics and Language Education. Vol. 3*, 299–310.

Arshad Abdul Samad (2002). The English of Malaysian School Students (EMAS) Corpus. Retrieved April 20, 2017 from http://works.bepress.com/arshad_abdsamad/2/

Arshad Abdul Samad, Hawanum Hussein (2010). Teaching Grammar and What Student Errors in the Use of the English Auxiliary 'be' Can Tell Us. *The English Language Teacher. Vol. 39*, 164-178.

Arshad Abdul Samad (2004). Beyond Concordance Lines: Using Concordances to Investigating Language Development. *Internet Journal of e-Language Learning & Teaching. Vol. 1*(1), 43-51.

Atkins, S., Clear, J. & Ostler, N. (1991). Corpus Design Criteria. Retrieved January 11, 2017 from http://www.natcorp.ox.ac.uk/archive/vault/tgaw02.pdf

Bank Negara Malaysia (2017). Islamic Banking and Takaful. Retrieved March 2, 2017 from http://www.bnm.gov.my/index.php?ch=fs_mfs_banks&act=55

Bennett, G. R. (2010). *Using Corpora in the Language Learning Classroom: Corpus Linguistics for Teachers*. Ann Harbour: University of Michigan Press. https://doi.org/10.3998/mpub.3715

Botley, S. & Doreen Dillah (2007). Investigating Spelling Errors in a Malaysian Learner Corpus. *Malaysian Journal of ELT Research*. *Vol. 3*, 74–93.

Chu, X., Ilyas, I. F., Krishnan, S., & Wang, J. (2016). Data cleaning: Overview and emerging challenges. 2016 International Conference on Management of Data Conference Proceedings, 26 June-01 July, California, USA (2016). https://doi.org/10.1145/2882903.2912574

Darina Lokeman Lok, Juliana Ali, Norin Norain Zulkifli Anthony (2013). A Corpus Based Study on the Use of Preposition of Time 'on' and 'at' in Argumentative Essays of Form 4 and Form 5 Malaysian Students. *English Language Teaching*. *Vol. 6*(9), 128-135.

Fox, C. (1989). A Stop List for General Text. *ACM SIGIR Forum*. *Vol. 24*(1–2), 19–21. https://doi.org/10.1145/378881.378888

Gimpel, K., Schneider, N., O'Connor, B., Das, D., Mills, D., Eisenstein, J.. & Smith, N. A. (2011). Part-of-speech tagging for twitter: Annotation, features, and experiments. 49th Annual Meeting of the Association for Computational Linguistics: Shortpapers Conference Proceedings. https://doi.org/10.1.1.206.3224

Granger, S. (1998). *Learner English on Computer*. London and New York: Addison Wesley Longman.

Granger, S. (2002). A Bird's-eye View of Learner Corpus Research. In S. Granger, J. Hung & S. Petch-Tyson (Eds.), *Computer Learner Corpora, Second Language Acquisition and Foreign Language Teaching* (pp. 3-33). Amsterdam: John Benjamins.

Hajar Abdul Rahim (2014). Corpora in Language Research in Malaysia. *Kajian Malaysia. Vol. 32*(1)*, 1–16.*

Imran Ho-Abdullah, Zaharani Ahmad, Rusdi Abdul Ghani, Nor Hashimah Jalaluddin, Idris Aman (2004). A practical grammar of Malay – A corpus-based approach to the description of Malay. First COLLA Regional Workshop Conference Proceedings, 28–29 June, Putrajaya, Malaysia (2004).

James, C. (1998). *Errors in Language Learning and Use. Exploring Error Analysis*. Haslow, Essex: Addison-Wesley Longman.

Janaki Manokaran, Chithra Ramalingam, Karen Adriana (2013). A Corpus-based Study on the Use of Past Tense Auxiliary 'be' in Argumentative Essays of Malaysian ESL Learners. *English Language Teaching. Vol. 6*(10), 111-119.

Jayakaran Mukundan, Rezvani Kalajahi, S. A. (2013). *Malaysian Corpus of Student Argumentative Writing*. Australia: Australian International Academic Centre.

Johns, T. (1991). Should You be Persuaded: Two Samples of Data-driven Learning Materials. *ELR Journal*. *Vol. 4*, 1–16.

Kennedy, G. (1998). *An Introduction to Corpus Linguistics*. London and New York: Longman.

Knowles, G., Zuraidah Mohd Don, Jariah Mohd Jan, Rajeswary Sargunam, Janet Yong, Sathia Devi, Asha Doshi, Su'ad Awab (2006). The Malaysian Corpus of Learner English: A Bridge from Linguistics to ELT. In H. Azirah & H. Norizah (Eds.), *Varieties of English in Southeast Asia and Beyond* (pp. 257-268). Kuala Lumpur: University of Malaya Press.

Leech, G. N. (1997). Introducing Corpus Annotation. In R.G. Garside, G. Leech, A.M. McEnery (Eds). *Corpus Annotation: Linguistic Information from Computer Text Corpora* (pp. 1-18). London and New York: Longman.

Marcus, M. P., Santorini, B. & Marcinkiewicz, M. A. (1993). Building a Large Annotated Corpus of English: The Penn Treebank. *Computational Linguistics*. *Vol. 19*(2), 313–330. https://doi.org/10.1162/coli.2010.36.1.36100

McEnery, A. & Hardie, A. (2011). *Corpus Linguistics: Method, Theory and Practice.* Cambridge: Cambridge University Press.

Mohamed Ismail Abdul Kader, Neda Begi, Reza Vaseghi (2013). A Corpus-based Study of Malaysian ESL Learners' Use of Modals in Argumentative Compositions. *English Language Teaching*. *Vol. 6*(9), 146-157.

Nesselhauf, N. (2005). Corpus Linguistics: A Practical Introduction. https://www.scribd.com/document/215218285/Corpus-Linguistics-Practical-Introduction-pdf

Rafidah Kamarudin (2013). A Corpus-based Study on the Use of Phrasal Verbs by Malaysian Learners of English: The Case of Particle *UP*. In S. Ishikawa (Ed.). *Learner Corpus Studies in Asia and The World Vol. 1* (pp. 255-270). Japan: Kobe University.

Roslina Abdul Aziz, Zuraidah Mohd Don (2013). The *BE* Verb Omission Among Advanced L1-Malay ESL Learners: What Corpus-based Study can Reveal. In S. Ishikawa (Ed.). *Learner Corpus Studies in Asia and the World Vol. 1* (pp. 121-138). Japan: Kobe University.

Roslina Abdul Aziz, Zuraidah Mohd Don (2014). The Overgeneration of *BE*+verb Construction in the Writing of L1-Malay ESL Learners in Malaysia. *Research in Corpus Linguistics., Vol. 2*, 35-44.

Roslina Abdul Aziz, Noli Nordin, Mohd Rozaidi Ismail, Norzie Diana Baharum, Roslan Sadjirin (2015). Building the Malaysian Corpus of Financial English (MaCFE). 2nd International Conference on Language, Education, Humanities and Innovation Conference Proceedings, 29-30 December, Kuala Lumpur, Malaysia.

Santorini, B. (1990). Part-of-speech Tagging Guidelines for the Penn Treebank Project (3rd Revision). University of Pennsylvania 3rd Revision 2nd Printing, 53(MS-CIS-90-47), 33. https://doi.org/10.1017/CBO9781107415324.004

Shazila Abdullah, Noorzan Mohd Noor (2013). Contrastive Analysis of the Use of Lexical Verbs and Verb-noun Collocation in Two Learner Corpora: WECMEL vs. LOCNESS. In S. Ishikawa (Ed.). *Learner Corpus Studies in Asia and the World Vol. 1* (pp. 139-160). Japan: Kobe University. Retrieved from http://www.lib.kobe-u.ac.jp/handle_kernel/81006680.

Shterev, Y. (2013). Demo: Using RapidMiner for Text Mining RapidMiner Possibility for Text Mining, *Digital Presentation and Preservation of Cultural and Scientific Heritage. Vol. 3*, 354-356.

Sinclair, J. (2004). Corpus and Text — Basic Principles. In M. Wynne (Ed.) *Developing Linguistic Corpora: A Guide to Good Practice* (pp. 1-16). Oxbow Books: Oxford. http://ahds.ac.uk/linguistic-corpora/.

Siti Aeisha Joharry, Hajar Abdul Rahim (2014). Corpus Research in Malaysia: A Bibliographic Analysis. *Kajian Malaysia*. *Vol. 32*(1), 17–43.

Su'ad Awab (1999). Multi-word Units in a corpus of Memoranda of Understanding. Modal multi-word units. Unpublished Ph.D Thesis, Lancaster University, UK.

Su'ad Awab (2003). Identifying an Unexplored Genre: Memoranda of Understanding. In Zubaidah Ibrahim et al. (Eds.) *Language, Linguistics and the Real World: Language Practices in the Workplace* (pp. 199-220). Kuala Lumpur: UM Publication.

Toutanova, K., Klein, D. & Manning, C. D. (2003). Feature-rich part-of-speech tagging with a cyclic dependency network. Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Vol. 1 Conference Proceedings, May 27-June 01, Edmonton, Canada (2003). https://doi.org/10.3115/1073445.1073478

Toutanova, K. & Manning, C. D. (2000). Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora Held in Conjunction with the 38th Annual Meeting of the Association for Computational Linguistics Conference Proceedings, 7-8 October, Hong Kong (2000). https://doi.org/10.3115/1117794.1117802

Verma, T. & Gaur, D. (2014). Tokenization and Filtering Process in RapidMiner. *International Journal of Applied Information Systems*. *Vol. 7*(2), 16–18.

Warren, M. (2010). Online Corpora for Specific Purposes. *ICAME Journal*. *Vol. 34*, 169–188. Retrieved from http://icame.uib.no/ij34/warren.pdf.

Yoon, H. (2011). Concordancing in L2 Writing Class. An Overview of Research and Issues. *Journal of English for Academic Purposes. Vol. 10*, 130-139.

Zarifi, A., & Jayakaran Mukundan (2014). Creativity and Unnaturalness in the Use of Phrasal Verbs in ESL Learner Language. *The Southeast Asian Journal of English Language Studies. Vol. 20*(3), 51-62.

Zimmermann, T., & Weißgerber, P. (2004). Preprocessing CVS Data for Fine-grained Analysis. 26th International Conference on Software Engineering Conference Proceedings, 25 May, Edinburgh, UK (2004). https://doi.org/10.1049/ic:20040466

Zuraidah Mohd Don (2010). Processing Natural Malay Texts: A Data-driven Approach. *TRAMES*. *Vol. 14*(64/59), 90–103.

Zuraidah Mohd Don, Sridevi Sriniwass (2017). Conjunctive Adjuncts in Malaysian Undergraduate ESL Essays: Frequency and Manner of Use. *Moderna Språk*. *Vol. 1*, 99-117.

APPENDIX A

Step in generating wordlist using RapidMiner Studio Educational (7.5.001) Text Processing Package:

Step 1: Create a process named "Process Documents from Files"

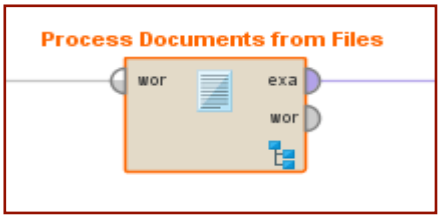

FIGURE 8. Process

Step 2: Assign the source of the folders and documents (text directories), and the value of vector creation on the parameters of the process.

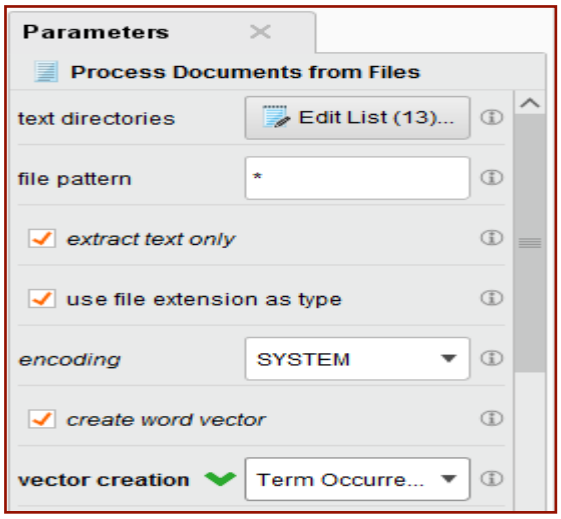

FIGURE 9. Parameter Window

Step 3: Edit parameter list: text directories by clicking on the 'Edit List'.

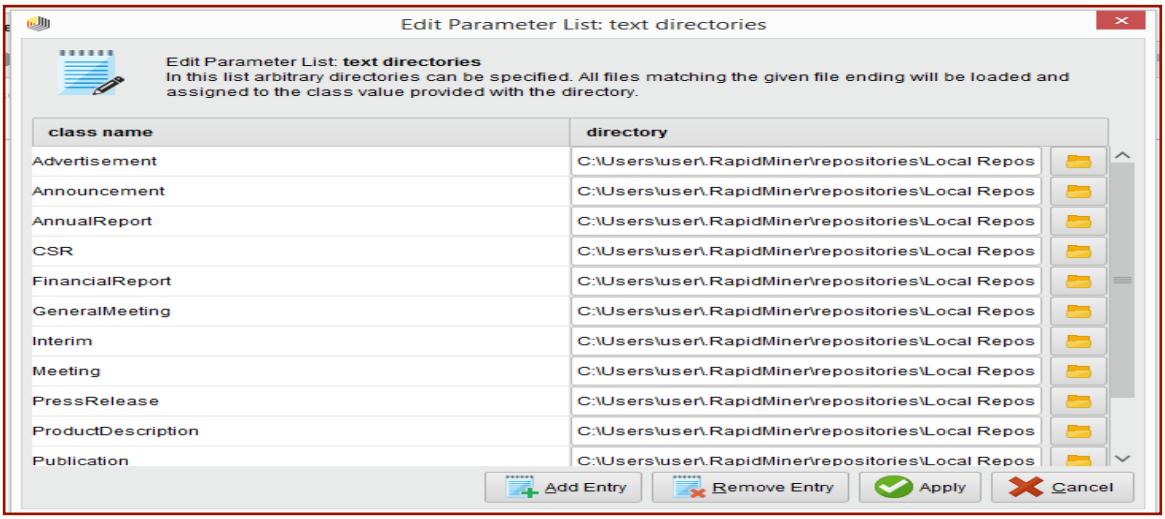

FIGURE 10. Process

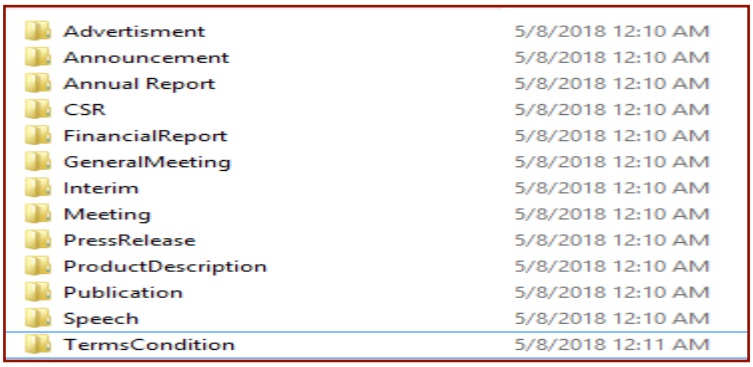

FIGURE 11. Snapshot of folders to be processed

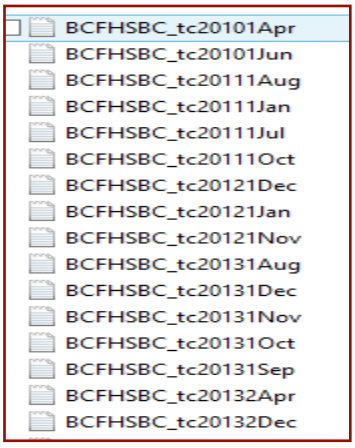

FIGURE 12. Snapshot of documents inside folder

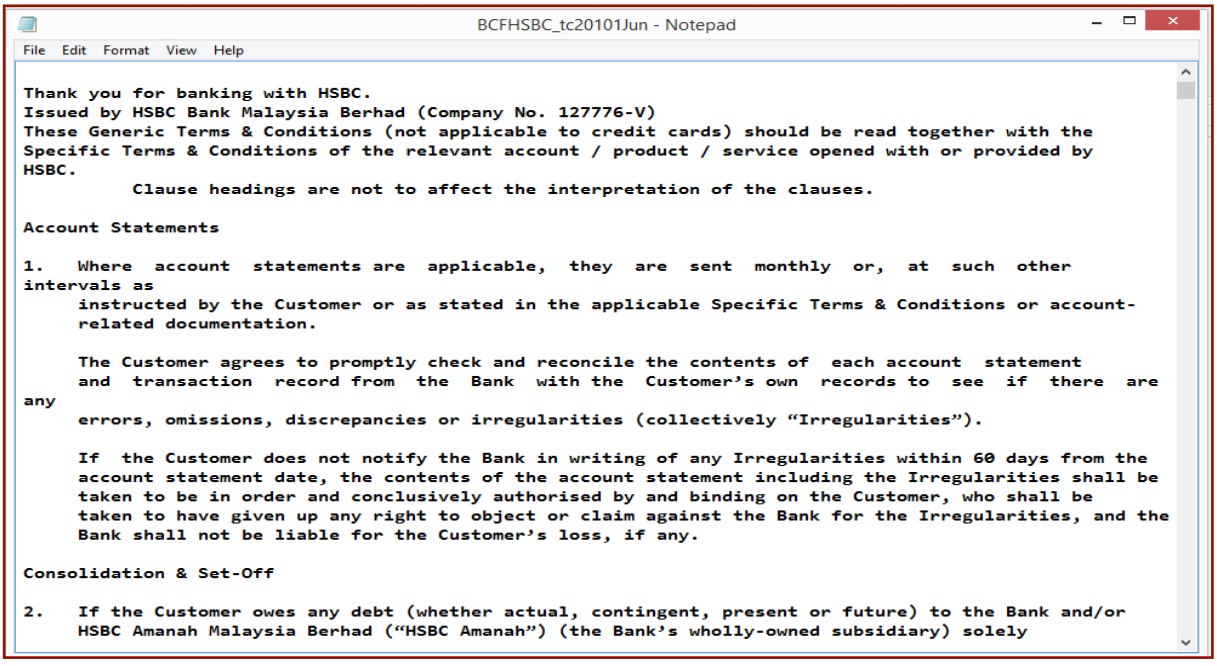

FIGURE 13. Snapshot of document content

Step 4: Create the jobs of the process as shown in the following figure. Simply run the process to produce the list of words, occurrences and the number of documents in which the words occurred.

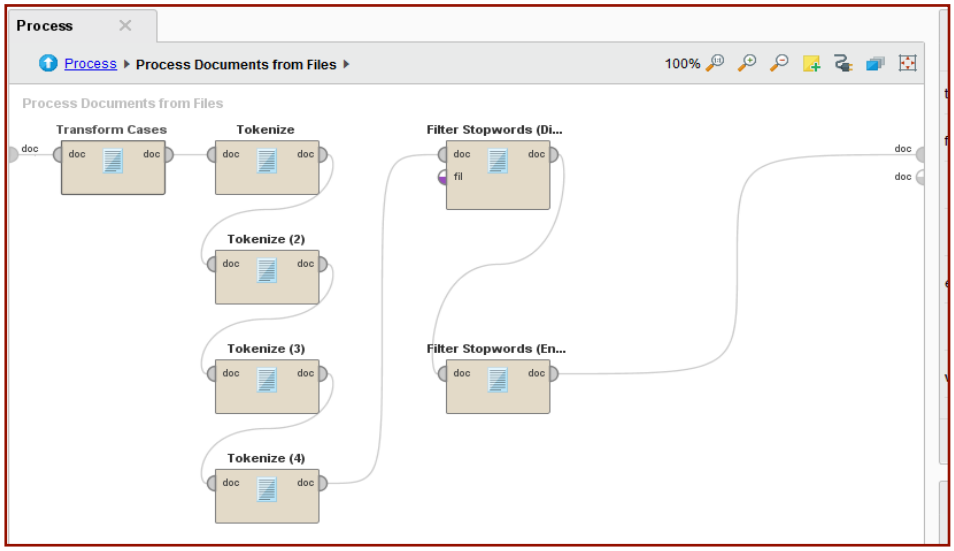

FIGURE 14. The jobs of process

bank	38086	1000
financial	20048	754
customer	18418	270
group	14801	275
account	12738	399
credit	11134	481
risk	10376	383
card	8399	176
management	7167	462
million	6839	273
growth	6551	418
banking	6233	547
cardholder	6086	106
market	5903	480
business	5597	545
year	5572	497
capital	5489	390
services	5241	503
time	5240	420
income	5169	404
cash	4921	273
conditions	4916	421
committee	4785	198
terms	4754	408
value	4726	339
interest	4446	427
assets	4427	357

FIGURE 15. Snapshot of words, occurrences and number of documents in the collection in which the words have occurred

APPENDIX B

Steps in POS tagging wordlist:

Step 1: Employ stanford-postagger-2016-10-31 to tag the word to its part-of-speech. Figure 16 below shows the POS tagged list of words generated from MaCFE.

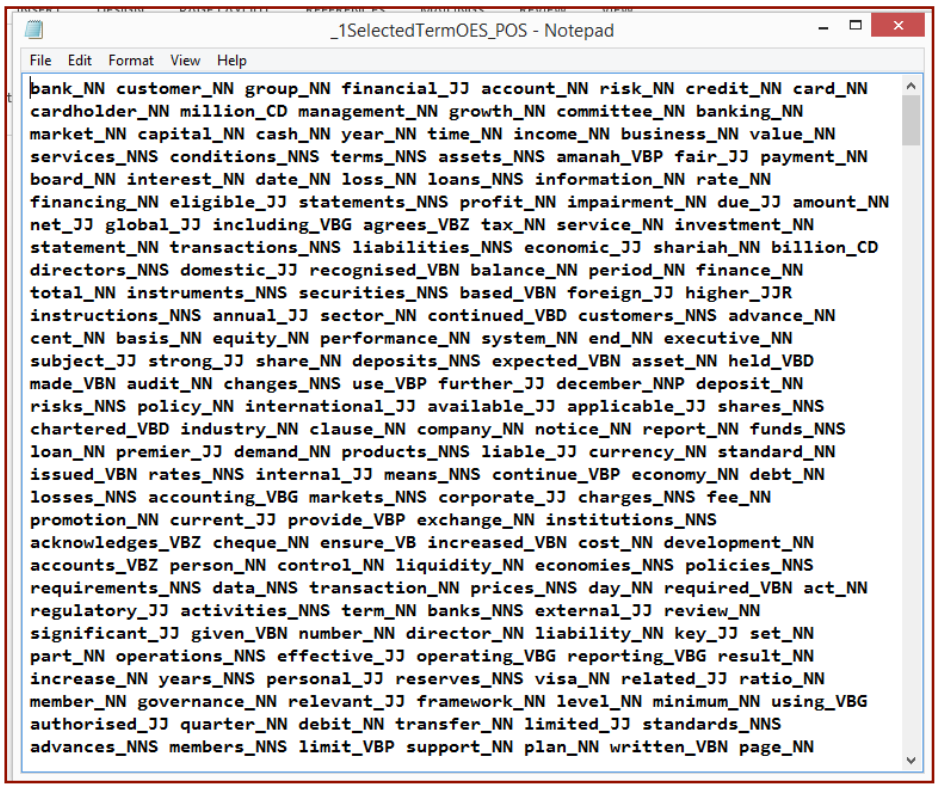

FIGURE 16. Snapshot of POS-tagged wordlist

Step 2: Execute the following Java application program (as shown in Table 9) to produce the list of formatted POS-tagged words and the frequency as shown in Figure 17 and Figure 18.

TABLE 9. Java Application Program

```
import java.io.*;
import java.util.*;

public class MyMaCFEApps
{
    public static void main(String[] args) throws FileNotFoundException, IOException {

        String infile ="_1SelectedTermOES_POS.txt";
        String outfile0 ="_2SelectedTermReportOES.txt";
        String outfile1 ="_3SelectedTermOES.txt";
        String outfile2 ="_4SelectedTermOccOES.txt";
        String outfile3 ="_5SelectedTermFreqOES.txt";
        String outfile4 = "_6SelectedTermOES_POS.txt";

        double totalToken=0.00000000;
        double CC, CD, DT, EX,FW, IN, JJ, JJR, JJS, LS;
        double MD, NN, NNS, NNP, NNPS, PDT, POS, PRP, PRP1, RB;
        double RBR, RBS, RP, SYM, TO, UH, VB, VBD, VBG, VBN;
        double VBP, VBZ, WDT, WP, WP1, WRB;

        CC = CD = DT = EX = FW = IN = JJ = JJR = JJS = LS = 0.00000000;
        MD = NN = NNS = NNP = NNPS = PDT = POS = PRP = PRP1 = RB = 0.0000000;
        RBR = RBS = RP = SYM = TO = UH = VB = VBD = VBG = VBN = 0.00000000;
        VBP = VBZ = WDT = WP = WP1 = WRB = 0.00000000;

        try {
            BufferedReader br = new BufferedReader (new FileReader (new File(infile)));
```

```
            PrintWriter pw0 = new PrintWriter (new FileWriter (new File (outfile0)));
            PrintWriter pw1 = new PrintWriter (new FileWriter (new File (outfile1)));
            PrintWriter pw2 = new PrintWriter (new FileWriter (new File (outfile2)));
            PrintWriter pw3 = new PrintWriter (new FileWriter (new File (outfile3)));
            PrintWriter pw4 = new PrintWriter (new FileWriter (new File (outfile4)));

            String str = br.readLine();
            while(str != null) {
                 StringTokenizer parse = new StringTokenizer(str, " ");
                 while (parse.hasMoreTokens()) {
                     String w = parse.nextToken();
                     String tag = "";
                     for (int i=0; i<w.length(); i++) {
                         if (w.charAt(i) == '_') {
                             tag = w.substring(i+1,w.length());
                             totalToken++;
                             break;
                         }
                     }

                     pw4.println(w);

                     if (tag.equalsIgnoreCase("CC")) CC++;
                     else if (tag.equalsIgnoreCase("CD")) CD++;
                     else if (tag.equalsIgnoreCase("DT")) DT++;
                     else if (tag.equalsIgnoreCase("EX")) EX++;
                     else if (tag.equalsIgnoreCase("FW")) FW++;
                     else if (tag.equalsIgnoreCase("IN")) IN++;
                     else if (tag.equalsIgnoreCase("JJ")) JJ++;
                     else if (tag.equalsIgnoreCase("JJR")) JJR++;
                     else if (tag.equalsIgnoreCase("JJS")) JJS++;
                     else if (tag.equalsIgnoreCase("LS")) LS++;

                     else if (tag.equalsIgnoreCase("MD")) MD++;
                     else if (tag.equalsIgnoreCase("NN")) NN++;
                     else if (tag.equalsIgnoreCase("NNS")) NNS++;
                     else if (tag.equalsIgnoreCase("NNP")) NNP++;
                     else if (tag.equalsIgnoreCase("NNPS")) NNPS++;
                     else if (tag.equalsIgnoreCase("PDT")) PDT++;
                     else if (tag.equalsIgnoreCase("POS")) POS++;
                     else if (tag.equalsIgnoreCase("PRP")) PRP++;
                     else if (tag.equalsIgnoreCase("PRP$")) PRP1++;
                     else if (tag.equalsIgnoreCase("RB")) RB++;

                     else if (tag.equalsIgnoreCase("RBR")) RBR++;
                     else if (tag.equalsIgnoreCase("RBS")) RBS++;
                     else if (tag.equalsIgnoreCase("RP")) RP++;
                     else if (tag.equalsIgnoreCase("SYM")) SYM++;
                     else if (tag.equalsIgnoreCase("TO")) TO++;
                     else if (tag.equalsIgnoreCase("UH")) UH++;
                     else if (tag.equalsIgnoreCase("VB")) VB++;
                     else if (tag.equalsIgnoreCase("VBD")) VBD++;
                     else if (tag.equalsIgnoreCase("VBG")) VBG++;
                     else if (tag.equalsIgnoreCase("VBN")) VBN++;

                     else if (tag.equalsIgnoreCase("VBP")) VBP++;
                     else if (tag.equalsIgnoreCase("VBZ")) VBZ++;
                     else if (tag.equalsIgnoreCase("WDT")) WDT++;
                     else if (tag.equalsIgnoreCase("WP")) WP++;
                     else if (tag.equalsIgnoreCase("WP$")) WP1++;
                     else if (tag.equalsIgnoreCase("WRB")) WRB++;
                 }

                str = br.readLine();
            }

            pw0.println("Total term;" +  totalToken);

            pw0.println("CC Coordinating conjunction;" + CC + ";" + Math.log10(CC / totalToken));
            pw0.println("CD Cardinal number;" + CD + ";" + Math.log10(CD / totalToken));
            pw0.println("DT Determiner;" + DT + ";" + Math.log10(DT / totalToken));
            pw0.println("EX Existential there;" + EX + ";" + Math.log10(EX / totalToken));
            pw0.println("FW Foreign word;" + FW + " " + Math.log10(FW / totalToken));
            pw0.println("IN  Preposition or subordinating conjunction;" + IN + ";" + Math.log10(IN /
totalToken));
            pw0.println("JJ  Adjective;" + JJ + ";" + Math.log10(JJ / totalToken));
            pw0.println("JJR Adjective, comparative;" + JJR + ";" + Math.log10(JJR / totalToken));
            pw0.println("JJS Adjective, superlative;" + JJS + ";" + Math.log10(JJS / totalToken));
            pw0.println("LS  List item marker;" + LS + ";" + Math.log10(LS / totalToken));

            pw0.println("MD  Modal;" + MD+ ";" + Math.log10(MD / totalToken));
            pw0.println("NN  Noun, singular or mass;" + NN + ";" + Math.log10(NN / totalToken));
            pw0.println("NNS Noun, plural;" + NNS + ";" + Math.log10(NNS / totalToken));
            pw0.println("NNP Proper noun, singular;" + NNP+ ";" + Math.log10(NNP / totalToken));
            pw0.println("NNPS    Proper noun, plural;" + NNPS + ";" + Math.log10(NNPS / totalToken));
            pw0.println("PDT Predeterminer;" + PDT + ";" + Math.log10(PDT / totalToken));
            pw0.println("Possesive Ending;" + POS + ";" + Math.log10(POS / totalToken));
```

```
            pw0.println("PRP Personal pronoun;" + PRP + ";" + Math.log10(PRP / totalToken));
            pw0.println("PRP$    Possessive pronoun;" + PRP1+ ";" + Math.log10(PRP1 / totalToken));
            pw0.println("RB  Adverb;" + RB + ";" + Math.log10(RB / totalToken));

            pw0.println("RBR Adverb, comparative;" + RBR + " " + Math.log10(RBR / totalToken));
            pw0.println("RBS Adverb, superlative;" + RBS + " " + Math.log10(RBS / totalToken));
            pw0.println("RP  Particle;" + RP + ";" + Math.log10(RP / totalToken));
            pw0.println("SYM Symbol;" + SYM + ";" + Math.log10(SYM / totalToken));
            pw0.println("TO  to;" + TO + ";" + Math.log10(TO / totalToken));
            pw0.println("UH  Interjection;" + UH + ";" + Math.log10(UH / totalToken));
            pw0.println("VB  Verb, base form;" + VB + ";" + Math.log10(VB / totalToken));
            pw0.println("VBD Verb, past tense;" + VBD + ";" + Math.log10(VBD / totalToken));
            pw0.println("VBG Verb, gerund, or present participle;" + VBG + ";" + Math.log10(VBG /
totalToken));
            pw0.println("VBN Verb, past participle;" + VBN + ";" + Math.log10(VBN / totalToken));

            pw0.println("VBP Verb, non-3rd person singular present;" + VBP + ";" + Math.log10(VBP /
totalToken));
            pw0.println("VBZ Verb, 3rd person singular present;" + VBZ + ";" + Math.log10(VBZ /
totalToken));
            pw0.println("WDT Wh-determiner;" + WDT + ";" + Math.log10(WDT / totalToken));
            pw0.println("WP  Wh-pronoun;" + WP + ";" + Math.log10(WP / totalToken));
            pw0.println("WP$ Possessive wh-pronoun;" + WP1 + ";" + Math.log10(WP1 / totalToken));
            pw0.println("WRB Wh-adverb;" + WRB + ";" + Math.log10(WRB / totalToken));

            pw1.println("CC Coordinating conjunction ");
            pw1.println("CD Cardinal number " );
            pw1.println("DT Determiner " );
            pw1.println("EX Existential there" );
            pw1.println("FW Foreign word " );
            pw1.println("IN Preposition or subordinating conjunction ");
            pw1.println("JJ  Adjective " );
            pw1.println("JJR Adjective, comparative ");
            pw1.println("JJS Adjective, superlative ");
            pw1.println("LS  List item marker " );

            pw1.println("MD  Modal " );
            pw1.println("NN  Noun, singular or mass ");
            pw1.println("NNS Noun, plural " );
            pw1.println("NNP Proper noun, singular ");
            pw1.println("NNPS    Proper noun, plural " );
            pw1.println("PDT Predeterminer ");
            pw1.println("POS  Possessive Ending " );
            pw1.println("PRP Personal pronoun ");
            pw1.println("PRP$    Possessive pronoun " );
            pw1.println("RB  Adverb ");

            pw1.println("RBR Adverb, comparative ");
            pw1.println("RBS Adverb, superlative ");
            pw1.println("RP  Particle ");
            pw1.println("SYM Symbol ");
            pw1.println("TO  to " );
            pw1.println("UH  Interjection ");
            pw1.println("VB  Verb, base form " );
            pw1.println("VBD Verb, past tense ");
            pw1.println("VBG Verb, gerund, or present participle ");
            pw1.println("VBN Verb, past participle ");

            pw1.println("VBP Verb, non-3rd person singular present ");
            pw1.println("VBZ Verb, 3rd person singular present " );
            pw1.println("WDT Wh-determiner " );
            pw1.println("WP  Wh-pronoun " );
            pw1.println("WP$ Possessive wh-pronoun ");
            pw1.println("WRB Wh-adverb ");

            pw2.println(CC);
            pw2.println(CD);
            pw2.println(DT);
            pw2.println(EX);
            pw2.println(FW);
            pw2.println(IN);
            pw2.println(JJ);
            pw2.println(JJR);
            pw2.println(JJS);
            pw2.println(LS);

            pw2.println(MD);
            pw2.println(NN);
            pw2.println(NNS);
            pw2.println(NNP);
            pw2.println(NNPS);
            pw2.println(PDT);
            pw2.println(POS);
            pw2.println(PRP);
            pw2.println(PRP1);
            pw2.println(RB);
```

```
            pw2.println(RBR);
            pw2.println(RBS);
            pw2.println(RP);
            pw2.println(SYM);
            pw2.println(TO);
            pw2.println(UH);
            pw2.println(VB);
            pw2.println(VBD);
            pw2.println(VBG);
            pw2.println(VBN);

            pw2.println(VBP);
            pw2.println(VBZ);
            pw2.println(WDT);
            pw2.println(WP);
            pw2.println(WP1);
            pw2.println(WRB);

            pw3.println(Math.log10(CC / totalToken));
            pw3.println(Math.log10(CD / totalToken));
            pw3.println(Math.log10(DT / totalToken));
            pw3.println(Math.log10(EX / totalToken));
            pw3.println(Math.log10(FW / totalToken));
            pw3.println(Math.log10(IN / totalToken));
            pw3.println(Math.log10(JJ / totalToken));
            pw3.println(Math.log10(JJR / totalToken));
            pw3.println(Math.log10(JJS / totalToken));
            pw3.println(Math.log10(LS / totalToken));

            pw3.println(Math.log10(MD / totalToken));
            pw3.println(Math.log10(NN / totalToken));
            pw3.println(Math.log10(NNS / totalToken));
            pw3.println(Math.log10(NNP / totalToken));
            pw3.println(Math.log10(NNPS / totalToken));
            pw3.println(Math.log10(PDT / totalToken));
            pw3.println(Math.log10(POS / totalToken));
            pw3.println(Math.log10(PRP / totalToken));
            pw3.println(Math.log10(PRP1 / totalToken));
            pw3.println(Math.log10(RB / totalToken));

            pw3.println(Math.log10(RBR / totalToken));
            pw3.println(Math.log10(RBS / totalToken));
            pw3.println(Math.log10(RP / totalToken));
            pw3.println(Math.log10(SYM / totalToken));
            pw3.println(Math.log10(TO / totalToken));
            pw3.println(Math.log10(UH / totalToken));
            pw3.println(Math.log10(VB / totalToken));
            pw3.println(Math.log10(VBD / totalToken));
            pw3.println(Math.log10(VBG / totalToken));
            pw3.println(Math.log10(VBN / totalToken));

            pw3.println(Math.log10(VBP / totalToken));
            pw3.println(Math.log10(VBZ / totalToken));
            pw3.println(Math.log10(WDT / totalToken));
            pw3.println(Math.log10(WP / totalToken));
            pw3.println(Math.log10(WP1 / totalToken));
            pw3.println(Math.log10(WRB / totalToken));

            br.close();
            pw0.close();
            pw1.close();
            pw2.close();
            pw3.close();
            pw4.close();

        }

        catch(FileNotFoundException e1) {System.err.println(e1.getMessage());}
        catch(IOException e2) {System.err.println(e2.getMessage());}

    }

}
```

eISSN: 2550-2131
ISSN: 1675-8021

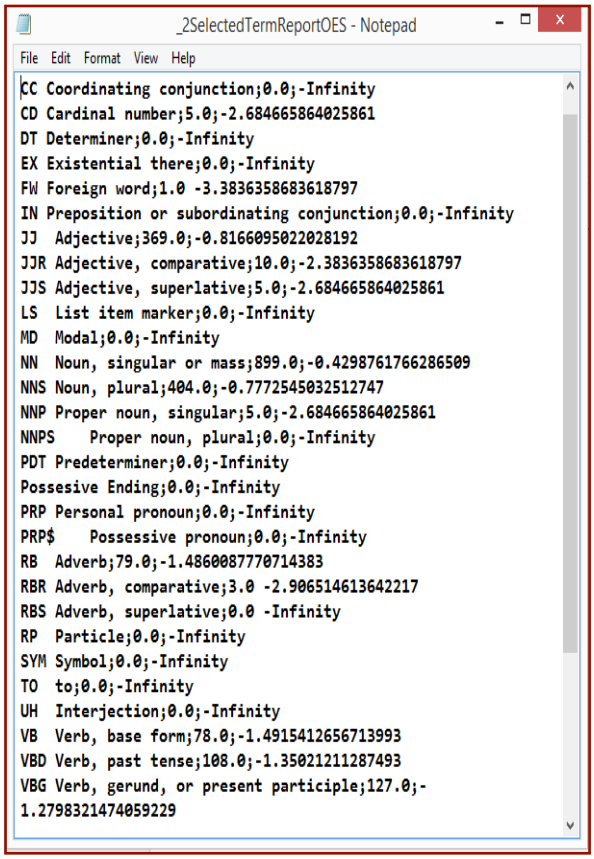

FIGURE 17. The frequency of POS-Tags of selected wordlist

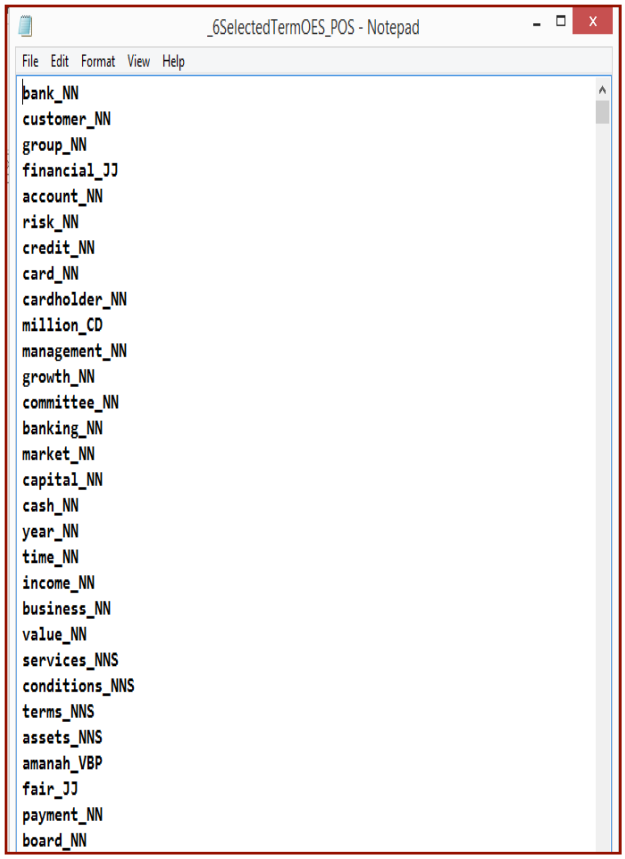

FIGURE 18. Selected wordlist and its POS-Tags

APPENDIX C

Samples of English Stopwords

1	a	26	can	51	having	76	it
2	about	27	can't	52	he	77	it's
3	above	28	cannot	53	he'd	78	its
4	after	29	could	54	he'll	79	itself
5	again	30	couldn't	55	he's	80	let's
6	against	31	did	56	her	81	me
7	all	32	didn't	57	here	82	more
8	am	33	do	58	here's	83	most
9	an	34	does	59	hers	84	mustn't
10	and	35	doesn't	60	herself	85	my
11	any	36	doing	61	him	86	myself
12	are	37	don't	62	himself	87	no
13	aren't	38	down	63	his	88	nor
14	as	39	during	64	how	89	not
15	at	40	each	65	how's	90	of
16	be	41	few	66	i	91	off
17	because	42	for	67	i'd	92	on
18	been	43	from	68	i'll	93	once
19	before	44	further	69	i'm	94	only
20	being	45	had	70	i've	95	or
21	below	46	hadn't	71	if	96	other
22	between	47	has	72	in	97	ought
23	both	48	hasn't	73	into	98	our
24	but	49	have	74	is	99	ours
25	by	50	haven't	75	isn't	100	ourselves

ABOUT THE AUTHORS

Roslan Sadjirin is a Senior Lecturer of Computing Sciences at the Faculty of Computer and Mathematical Sciences at Universiti Teknologi MARA Cawangan Pahang. He received his MSc. in Computer Science in 2008 and BSc (Hons.) in Information Technology in 2005 from Universiti Teknologi MARA (UiTM) Malaysia. His teaching expertise includes data structures, computer programming and problem solving. He has published papers in the areas of computer sciences education, information retrieval and text processing. His research interests lie in computational linguistics, big data and text processing. His previous research grant was in classification of word usage based on the gender preferences.

Roslina Abdul Aziz is currently attached to Akademi Pengajian Bahasa (APB), Universiti Teknologi MARA Cawangan Pahang and has had more than 20 years of experience as an English language instructor at the same university. She received her B. Ed TESL (Hons.) and M.A in Language Studies from Universiti Kebangsaan Malaysia (UKM) and she is currently pursuing her Ph.D in Corpus Linguistics in University of Malaya, Kuala Lumpur. Her research interests include areas in Corpus Linguistics and Language for Specific Purposes.

Noli Maishara Nordin received her bachelor's degree from International Islamic University Malaysia (IIUM) in English Language & Literature in 2006 and her M.A. in Applied Linguistics at Universiti Putra Malaysia (UPM) in 2010. She has been a member of the Akademi Pengajian Bahasa (APB), Universiti Teknologi MARA Cawangan Pahang since 2010 where she is currently a lecturer of English Language. She has published papers in the areas of education, sociolinguistics and computational linguistics. Her current areas of interest are in corpus linguistics and linguistic landscape.

Mohd Rozaidi Ismail is a Senior English Language Lecturer with the Akademi Pengajian Bahasa (APB), Universiti Teknologi MARA (UiTM) Pahang. He holds a Bachelor of Education in TESL and Masters of Arts in English Language Studies from Universiti Kebangsaan Malaysia (UKM). He has been with UiTM Pahang for 16 years and his primary academic, research and publication interests are Open Source Systems and Technology, Online Application Design and Development, ESL Education, Instructional Design Technology and E-Learning.

Norzie Diana Baharum has been teaching English proficiency courses at UiTM Cawangan Pahang for 9 years. She obtained her Bachelor Degree in English Language and Literature and Master's in English Literary Studies from International Islamic University of Malaysia (IIUM). Her research interests include sociolinguistics, critical thinking and literatures in English with women's writings being her keen interest.

eISSN: 2550-2131
ISSN: 1675-8021