USING ACONCORD AND ANTCONC TO COMPARE GARFIELD'S CONTINUATION OF DICKENS' NOVEL AND THE ORIGINAL TEXT OF QURAN WITH A TRANSLATED TEXT

Abdulaziz Al Jumaia
University of Wolverhampton
A.Aljumaia@wlv.ac.uk

***Abstract*—** This work compares firstly, the differences that occur in "The Mystery of Edwin Drood" which was written by two different authors, Garfield and Dickens. And, secondly, the differences of the original text of the Quran and a translated version of it by Abdullah Yusuf Ali. In translation there are differences which occur in the frequency of the prophets' names. Thus, this work seeks to find the occurrences of Moses and Jesus in the Arabic text and compare them to the translated text. Both of the goals in this project are achieved using Antconc and Aconcord, tools provided with many features that help the computation of words frequencies.

Keywords— Collocation, NLP, Computational Linguistics, Corpus, Arabic, Quran.

I. Introduction

Individuals differ from each other in many aspects such as in the way of speaking, working, thinking and even the way of eating can differ from one to another. That is because of several factors like the way of living, the people they speak with and maybe the positions that they hold. Similarly, there are many authors with their own unique style such as Agatha Christie, a famous author of crime fiction. She had a very simple writing style, so that her audience could read her works easily (freelanceWriting. 2016). However, other authors might imitate her works as they are influenced by her style, but, there would still be some differences between her writing style and the other authors'. Here we will discuss a case where two authors have worked on the same

novel. According to Sparknotes (2016) Charles Dickens could not finish his novel "The Mystery of Edwin Drood" as he died in 1870. The novel had been unfinished for 9 decades. However, there were a number of attempts to finish the novel. One of the authors who finished the novel was Leon Garfield. He finished the story in 1980.

However, as it had been 9 decades since the death of Dickens and the continuation of Garfield, there might be some differences between their writing styles. Therefore, this paper's aim is to find (1) the frequency of each author's words in the novel, (2) the use of contractions and n-grams, which are the consecutive words that usually occur together, of both writers, (3) the frequency of the novel characters with both authors. We use the AntConc tool that counts and shows the words in context. While this paper seeks to find the differences that occur in individuals' writings, it also seeks to find the differences that occur between the original Quranic text and the translation of Abdullah Yusuf Ali which is recommended to be the best translation for the original verses (Usman Farooq, 2010). (4) Two different software packages are used to observe the frequency of two prophets' names, Moses and Jesus, in the original and translated text.

II. Literature Review

There have been so many different corpus-based works within the last five years. However, the literature review in this paper is based on the fourth chapter of Cheng's book (2012). In this chapter she discusses three points related to corpus search and analysis methods which are: (1) The description of the basic of corpus search function and the methods of analysis of corpus linguistics. (2) Introduction to functions available in the software packages of corpus linguistics, say, *WordSmith 5.0* such as the generation of word frequency lists and keyword lists. (3) Introduction to a less common function "*ConcGram*". It is developed in order to find the phraseological variation (Greaves, C. 2009; Cheng, W. 2012).

To begin with, Cheng illustrates that it is simple to generate a whole list of words with their frequency and percentage of occurrence. The list of the words is sorted either alphabetically or based on the words frequency. Of course, that depends on the programs used in the procedure. Also, several programs count "isn't", "don't and "aren't" as a single word instead of two words. On the other hand, there are other programs that count the same mentioned words above as two words as they originally consist of two words rather than one. However, Cheng includes an example which shows that the program used to count the words does not only count "s" as a separate type but also distinguishes its type as in "she's" and "Susan's" (see figure 1 rank 36 in BNC). This is useful because the corpus used is the British National Corpus (BNC). This corpus is part-of-speech tagged. Thus, the "s" in "it's" is tagged as a verb on its own. Therefore, the software is able to recognize it as a single word. Cheng also makes a comparison among three different corpora: BNC, HKFSC (Hong Kong Financial Corpus Services) and HKEC (Hong Kong Engineering Corpus) with 100 million, 7.3 million and 9.2 million words respectively. It appears that HKFSC and HKEC do not have 's appearing as frequently as in the BNC (see table 1).

Rank	Word	Frequency	%	Rank	Word	Frequency	%	Rank	Word	Frequency	%
1	the	6,184,700	6.29	1	the	484,439	7.32	1	the	582,437	6.81
2	of	2,939,100	2.99	2	of	297,781	4.50	2	of	335,785	3.92
3	and	2,681,700	2.73	3	and	203,418	3.07	3	and	265,945	3.11
4	a	2,162,600	2.20	4	to	175,283	2.65	4	to	204,117	2.39
5	in	1,821,400	1.85	5	in	167,158	2.52	5	in	168,798	1.97
6	to	1,628,400	1.66	6	a	99,448	1.50	6	a	132,246	1.55
7	it	1,087,500	1.11	7	for	70,878	1.07	7	for	117,325	1.37
8	is	998,200	1.01	8	or	60,299	0.91	8	be	95,221	1.11
9	to	934,300	0.95	9	is	57,610	0.87	9	is	84,323	0.99
10	was	923,600	0.94	10	as	55,712	0.84	10	on	59,914	0.70
11	I	887,500	0.90	11	on	54,750	0.83	11	with	58,878	0.69
12	for	841,200	0.86	12	by	50,530	0.76	12	by	53,059	0.62
13	that	730,800	0.74	13	be	41,224	0.62	13	or	51,863	0.61
14	you	695,400	0.71	14	with	38,725	0.58	14	as	51,413	0.60
15	he	681,000	0.69	15	at	38,351	0.58	15	are	45,190	0.53
16	be	664,400	0.68	16	are	37,198	0.56	16	at	42,836	0.50
17	with	657,500	0.67	17	Hong	37,171	0.56	17	that	38,881	0.45
18	on	647,500	0.66	18	Kong	36,184	0.55	18	from	37,521	0.44
19	by	509,600	0.52	19	that	33,563	0.51	19	Hong	33,999	0.40
20	at	479,000	0.49	20	from	31,166	0.47	20	Kong	32,806	0.38
21	have	473,500	0.48	21	company	30,466	0.46	21	should	27,911	0.33
22	are	470,700	0.48	22	which	27,059	0.41	22	this	27,797	0.32
23	not	462,600	0.47	23	any	26,536	0.40	23	shall	26,990	0.32
24	this	462,300	0.47	24	HK	26,191	0.40	24	will	25,730	0.30
25	's	459,900	0.47	25	an	25,683	0.39	25	not	25,356	0.30

Table 1: A comparison among three corpora.
BNC is on the left, HKFSC is on the center and HKEC is on the right (Cheng, W. 2012).

We now present an overview of other list generated features of software packages. The feature of "keywords" provides details about specific types of words to the software user. For instance, there are words that do not appear frequently in some general corpora, but, appear more than usual in some specialised corpora. The word "tsunami" is a notable example of the words that appear in the Hong Kong Budget speech for the first time in 2010. The frequency of the word is 16 in 2010 and its rank is 1. "N-grams" is another feature of the software packages mentioned earlier. It shows the consecutive words that appear over and over again together. For instance, "A lot of" consists of three words which appear frequently (see table 2).

N-gram	Frequency	Part-of-speech
I do n't	36,863	PNP VDB XX0
one of the	35,273	CRD PRF AT0
the end of	20,998	AT0 NN1 PRF
part of the	16,444	NN1 PRF AT0
do n't know	15,779	VDB XX0 VVI
some of the	15,149	DT0 PRF AT0
a number of	15,126	AT0 NN1 PRF
there is a	15,008	EX0 VBZ AT0
a lot of	14,561	AT0 NN1 PRF
# and #	14,451	CRD CJC CRD

Table 2: Example of three words $N-grams$

Finally, we describe the features that are less common for users. As mentioned previously, the feature is programmed in order to search for phraseological variation in corpora. Indeed, it is developed to fully retrieve co-occurrences of between two and five words such as "risk

management" and "risk controlled approach to portfolio management" as Cheng illustrated. The co-occurrences are called "concgrams" (Cheng et al.,2006). However, there may be a lot of congrams in a corpus, such as HKFSK, and displaying them may take a while depending on the power of the computer (see table 3). Furthermore, Cheng notes that once we find a two-word congram, then it is possible to find other two-word congrams within the same corpus. Likewise, the other congrams like three-word and four-word. If we find a congram with co-occuring words, then we can select it and see what words co-occur with it (see table 4). Moreover, the congram feature can nominate up to five – or as many as the user likes- co-occurrences and display them by using the search function of congram "user-nominated" (see table 5).

Two-word congram	Frequency
1. of/the	436,870
2. the/to	204,920
3. and/the	204,876
4. in/the	198,052
5. and/of	125,874

Table 3: The frequency of two – word congrams. The list is huge and may take too long time to display.

Two-word congram	Co-occuring word	Frequency
1. environmental/waste	the	35
2. environmental/waste	and	34
3. environmental/waste	on	17
4. environmental/waste	of	16
5. environmental/waste	protection	15
6. environmental/waste	in	13
7. environmental/waste	to	9

Table 4: The chosen congram with a list of co – occur words

2	isciplined and proactive **management** of such **risk.** The fram
3	d the need for effective **management** of credit **risk.** Despit
4	e high level centralised **management** of credit **risk** for HSB
5	. HSBC aims, through its **management** of market **risk** in non-
6	latory requirements. Our **management** of operational **risk** be
7	aluation, acceptance and **management** of some degree of **risk**

Table 5: The required congram with a list of up to five co – occur words.

III. Methodology

In this section we present the method used in this paper in order to answer the questions asked at the beginning of this study. However, there are three tools we used to achieve our purpose. Firstly, we use a simple tool called *AntConc* for many reasons explained in the next section (see section 4.1). Secondly, we use another tool, *Aconcord*, in order to compare its results with the results of the previous tool. Thus, both of the tools complement each other in this project.

III.1. AntConc

Primarily, *AntConc* is a program with a simple graphical interface that allows users to search, count, and view text. It has so many features such as using regular expression, Unicode support and multiple-level sorting. Above all, it is free to use unlike *ConcGram* (Anthony, 2005). In this work we use AntConc in order to answer the main questions of this research. First, we make a comparison between the first sample of "The Mystery of Edwin Drood" which was written by Dickens himself with the second sample that was done by Garfield. As the program is able to make a list of the most frequent words in the text, it is possible to see the most 10 frequent words for each author. Also, the program allows the user to use regular expression. Thus, we can search for the contractions such as "don't", "isn't" and so on.

For the second application, since the Quran has many translations in English, we use this tool to sort the words at the frequency level. Also, we use it to find if the numbers of the words are similar – or even close – to the number of the same words in the original text in Quran. To put it more simply, the words in English are written using different synonyms like "global" and "universal". Simply, we find the frequency of an Arabic word via a tool called *Aconcord* (see section 4.2) and then we find the translation of the same word and count its frequency via *AntConc*.

III.2. Aconcord 0.4.3

As well as *AntConc*, *Aconcord* is a program can do similar tasks with texts. However, Arabic is fully supported by this tool. The tool is free to use, does not need to convert Arabic text into Latin script, and is supported by an Arabic and English interface (see figure 1). Furthermore, it is a multiplatform tool, therefore, it can be used on Linux, windows, and Mac as long as Java is installed on the user's computer (Andrew Roberts Blog. 2016). However, with this tool we found the frequencies of the prophets' names and then we compare them with the translated text after analysed by AntConc.

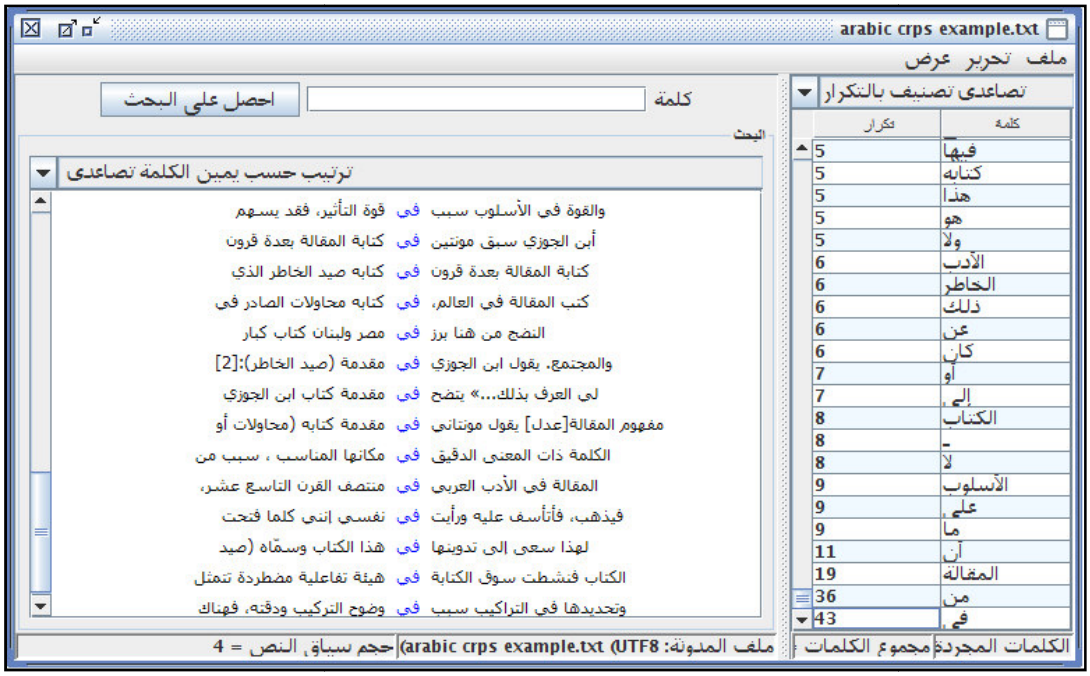


Figure 1: An example of aConCord with a sorted Arabic words in frequency level.

IV. Experimental Setup

The comparison is made here similar to Winnie Cheng's work in the literature review section. However, it is an experiment done on two different parts of "The Mystery of Edwin Drood" written by two authors as we illustrated in the introduction. We used the "sort" and "regular expression" features of *AntConc* for this experiment. The purpose of using regular expression was to find the n-grams and contractions such as "don't" and "isn't" and to find the frequency of characters' names frequency in the Dickens and Garfield sample.

Secondly, in the third section Winnie Cheng illustrates that the software packages have many features that can be implemented on texts like sorting the most frequent words and showing the co-occuring words and so on. However, this paper takes the features of the software packages a step forward by combining two different features from two different software packages, the *AntConc* and *Aconcord*. Simply, the first one is employed to find the most frequent words in the English translation of the Quran made by Abdullah Yusuf Ali, whilst the second one does likewise with the Original text. Therefore, this experiment uncovers the differences of words – or even sentences - used between two versions. For instance, the verse:

"إِنَّا أَنْزَلْنَا إِلَيْكَ الْكِتَابَ بِالْحَقِّ لِتَحْكُمَ بَيْنَ النَّاسِ بِمَا أَرَاكَ اللَّهُ"

is translated in the Sahih International version as "*Indeed, we have sent down to you the Book, in truth. So worship Allah, [being] sincere to Him in religion*". However, there are additions in

phrase level in this sentence. Thus, with the tools we use in this project it is possible to observe an increase - or decrease - in the frequent words used in the translated version.

V. Findings and Discussion

V.1. A Comparison Between Dickens's and Garfield's Work

Firstly, in this paragraph we compared Dickens's words frequencies with Garfield's. On the one hand, we observe the frequency of the contractions, n-grams and the most frequent words in general in Dickens's part. The word "don't" is the most frequent contraction as it appears 13 times, while "of the", "in the", "to the", and "on the" are the most frequent n-grams as they appear in the n-gram graphical window on *AntConc* 92, 72, 40 and 34 times respectively. Also, the most frequent general words in the whole sample of Dickens are "the" 774 times, "and" 483 times, "of" 459 times and "to" 406 times respectively. On the other hand, the most frequent contraction is "don't" which appears 8 times only, while the most frequent n-grams are "of the", "in the", "it is" and "as if" with 58, 41 38 and 33 occurrences respectively. Furthermore, the most frequent words in general are "the", "and", "of" and "to" with 774, 483, 459 and 406 occurrences respectively.

Secondly, we discuss the results, n-grams and contractions, of Garfield and Dickens and compare them to each other. Dickens' sample consists of almost fourteen thousand words. Nevertheless, it appears that Dickens writing was a mixture of contractions and n-grams. For instance, the characters of the novel say "I do not" and "I don't" between themselves in the dialogues. However, it is obvious that Garfield well-imitated Dickens in his writing style as he was able to reach Dickens' writing level.

Finally, we compare the most frequent characters' names in both of the samples. However, it is impossible to find names though thousands of words, 14 thousand for Dickens' sample and almost 12 thousand for Garfield's. Hence, we have to find the character's name first. Simply, we find the important characters' name after Edwin Drood as he is the main characters of the novel. Therefore, according to eNotes (2016) there are eight important characters mentioned in the story such as Jack Jasper. We used regular expression again to find the first name, last name, or both. To put it more simply, the regex code below in was used order to find either the first of the last name of characters.

"$((Charfirst1|\dots)\{1\}\backslash s(Charlast1)\{0,1\})|(\ Charfirst1)\{0,1\}\backslash s(Charlast1)\{1\}$"

However, $(charfirst1|..)$ represents the first name of each character, while $(Charlast1|..)$ does likewise; however, it represents the last name of each character instead of the first name. The code "*{0,1}*" tells the program to find the required element either once or never. The code "{1}" tells the program that it should find it once in our case the elements are the first and the last names. Thus, the program should find the last name in the first case once or the first name once in the second case. However, the purpose of this code is to find the names only. Then, we clicked on each one of them so we could count their first and last names. The most frequent characters' names in the Dickens sample are Crisparkle as it appears 43 times, then Jasper, and Durdles with 37 and 27 occurrences respectively. The results are surprising as Edwin Drood

himself is not the most mentioned character. The most mentioned characters in Garfield's sample are Jasper, Datchery and Helena Landless respectively (see table 6). Thus, it appears that the most mentioned characters in the samples are Jasper and Crisparkle.

Characters	Dickens	Garfield
Edwin Drood	17	1
Jasper	37	22
Rosa Budd	25	10
Crisparkle	43	10
Helena Landless	20	12
Neville Landless	20	6
Datchery	25	17
Grewgious	15	11
Durdles	27	8

Table 6: The most freqnet names in both samples of the novel.

V.2. Comparison Between the Original Quran and its Translation by Abdullah Yousuf Ali by Using Both AntConc and Aconcord.

Arabic is, until a certain level, a morphologically complex language. Thus, Arabic words are a combination of many parts of speech (Habash, N. and Rambow, O. 2005; Maamouri et al, 2004). In this section we seek to find the frequency of Jesus' and Moses' names in Quran and then compare with the frequencies of the translated text. *Aconcord* shows 129 occurrences for Moses and 15 for Jesus (see figure 2). That is because of the attachment of parts-of-speech that occur with the same name. For instance, the name موسى (Mosa: the Arabic pronunciation of prophet Moses) can occur as by itself in the Quranic text - also in the standard Arabic. However, the same name can occur with a particle attached like in ياموسى. Thus, the program can not find the name in the second case in the example above as it matches the exact characters even with the diacritics like in مِنَ (mɪnə: from us) and مَنٌ (mən: who). That may be a feature to distinguish the semantic meaning of the Arabic words as they are written in the same way but with different diacritics that may – or may not - change the meaning. However, it is a disadvantage in this circumstance. Thus, the only way to find the rest of the names is to search in the whole list or type the names with the attached particles.

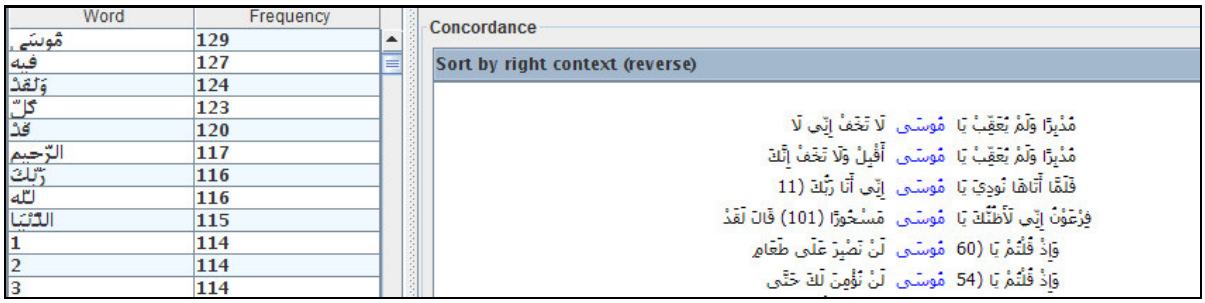
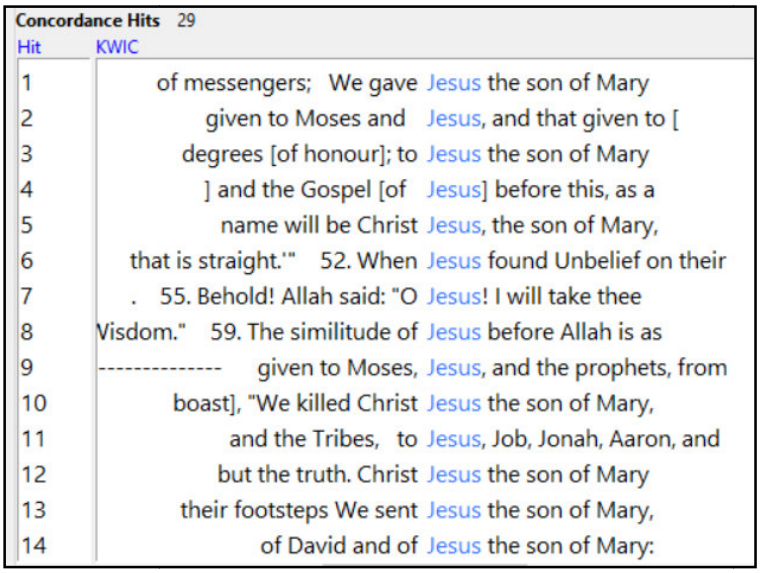


Figure 2: The name of Moses appears 129 times without attached particle.

However, it appears that Moses's name occurs 176 times in the English translation by Abdullah Yusuf Ali, while in the original text it appears 136 times. That is because in translation techniques there are additions in order to make a text explicit (Molina, L. and Hurtado Albir, A. 2002). Likewise, the frequency of Jesus increased from 25 times in the original text to 29 in the translation text (see figure 3).



Concordance Hits 29

Hit	KWIC
1	of messengers; We gave Jesus the son of Mary
2	given to Moses and Jesus, and that given to [
3	degrees [of honour]; to Jesus the son of Mary
4	] and the Gospel [of Jesus] before this, as a
5	name will be Christ Jesus, the son of Mary,
6	that is straight.'" 52. When Jesus found Unbelief on their
7	. 55. Behold! Allah said: "O Jesus! I will take thee
8	Visdom." 59. The similitude of Jesus before Allah is as
9	---------- given to Moses, Jesus, and the prophets, from
10	boast], "We killed Christ Jesus the son of Mary,
11	and the Tribes, to Jesus, Job, Jonah, Aaron, and
12	but the truth. Christ Jesus the son of Mary
13	their footsteps We sent Jesus the son of Mary,
14	of David and of Jesus the son of Mary:

Figure 3: The occurrences of Jesus in the translated text.

VI. Limitations of Antconc and Aconcord

In this section we present the limitations encountered in this study. Firstly, *AntConc* showed many errors in the texts. The most frequent error encountered was the replacement of apostrophes with \x92. Simply, we had to write \x92 in the regular expression codes used to obtain the required results. However, we were not able to count the exact number of the most frequent contractions as there were many replaced symbols appearing in the list. For instance, x was the most frequent word in the frequency list. However, x might be a substitute for a letter, apostrophe or even a space. The same occurred with the n-grams as x appeared again accompanied with another word. However, we managed to count the most frequent n-grams in a previous section in this study. On the other hand, Aconcord showed a limitation in matching words. Basically, it showed the exact words that were typed on top of each other. However, some of the word did not appear in the program as they were attached to particles. Hence, the program did not count them. Therefore, the only way to find them was to search in the whole given list.

VII. Conclusion

With everything taken into account, this paper discussed Cheng's work in the literature review section. Cheng used software packages, Wordsmith and ConcGram, in her work and discussed them in detail. She also compared three different corpora, BNC, HKFSC and HKEC, with each other and showed the differences with the most frequent words list in each corpus. She also discussed other features such as n-grams and co-occurrences. However, as this paper is related to her work, we discussed the features of AntConc and Aconcord. Antconc can display the most frequent words list, n-grams with as many co-occurrences selected and the use of regular expression. However, some errors were encountered with the program like the substitution of apostrophes. Aconcord is the tool used to compare the Arabic text with the translated text. Furthermore, the experiment succeeded due to its use. Finally, the results for the "Edwin Drood" novel showed that Garfield well-imitated Dickens as he used a limited number of contractions. Also, the most mentioned characters in both works was Jasper. The second part of the study observed a difference in the number of mentions of prophets' names. They both increased in the translated document as this was a result of translation techniques. In certain circumstances translators need to add more details that do not appear in the original text as in the case of "*[of Jesus]"*, line 4 in figure 9, to make the text explicit to its audience. Aconcord encountered a limitation in the searching process as the attachments of Arabic letters and the place of the diacritics do not always match the inputs. Also, the feature of "Regex" is not included.

References

Anthony, L (2005). AntConc: design and development of a freeware corpus analysis toolkit for the technical writing classroom [online]. In Professional Communication Conference, 2005. IPCC 2005. Proceedings. International (pp. 729-737). IEEE. [Accessed 29 April 2016]. Available at: *http://ieeexplore.ieee.org.ezproxy.wlv.ac.uk/stamp/stamp.jsp?tp=&arnumber=1494244*

Cheng, W., Greaves, C. and Warren, M. (2006) From n-gram to skipgram to concgram. International journal of corpus linguistics [online], **11**(4), pp.411-433. [Accessed 5 May 2016]. Available at: *http://www.ingentaconnect.com.ezproxy.wlv.ac.uk/content/jbp/ijcl/2006/00000011/00000004/art00003*

Cheng, W. (2012) Exploring Corpus Linguistics Language in Action. London: Routledge.

eNotes (2016) The Mystery of Edwin Drood Characters [online]. [Accessed 2 May 2016]. Available at: *http://www.enotes.com/topics/mystery-edwin-drood/characters#characters-characters-discussed*

eNotes (2016) The Mystery of Edwin Drood Essay - Critical Essays [online]. [Accessed 25 April 2016]. Available at: *http://www.enotes.com/topics/mystery-edwin-drood/critical-essays#critical-essays-critical-evaluation*

Farooq, U (2010) Dr. Zakir Naik recommends English translation of Qur'an [online]. [Accessed 1 May 2016]. Available at: *https://www.youtube.com/watch?v=C-1rcq6IlOo&t=0m30s*

Freelance Writing (2016) The Writing Style of Agatha Christie [online]. [Accessed 25 April 2016]. Available at: *http://www.freelancewriting.com/articles/WS-the-writing-style-of-agatha-christie.php*

Greaves, C. (2009). ConcGram 1.0: A phraseological search engine [online]. Amsterdam: John Benjamins Publishing Company. [Accessed 5 May 2016]. Available at: *https://benjamins.com/#catalog/software/cls.1/main*

Habash, N. and Rambow, O. (2005) Arabic tokenization, part-of-speech tagging and morphological disambiguation in one fell swoop. In Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics [online] (pp. 573-580). Association for Computational Linguistics. [Accessed 5 May 2016]. Available at: *http://delivery.acm.org/10.1145/1220000/1219911/p573-habash.pdf?ip=31.205.26.157&id=1219911&acc=OPEN&key=4D4702B0C3E38B35%2E4D4702B0C3E38B35%2E4D4702B0C3E38B35%2E6D218144511F3437&CFID=780809505&CFTOKEN=49852218&__acm__=1462431755_2d85a7acb639349d51661b17fffd3e72*

Maamouri, M., Bies, A., Buckwalter, T. and Mekki, W. (2004) The penn arabic treebank: Building a large-scale annotated arabic corpus [online]. [Accessed 7 March 2016]. Available at: *https://www.researchgate.net/profile/Mohamed_Maamouri/publication/228693973_The_penn_arabic_treebank_Building_a_largescale_annotated_arabic_corpus/links/0046351802c78190c5000000.pdf.*

Molina, L. and Hurtado Albir, A. (2002) Translation techniques revisited: A dynamic and functionalist approach. Meta: Journal des traducteursMeta:/Translators' Journa [online]l, **47**(4), pp.498-512. [Accessed 24 August 2015]. Available at: *http://www.erudit.org/revue/meta/2002/v47/n4/008033ar.pdf*

Roberts, A (2016) aConCorde [blog entry]. [Accessed 5 May 2016]. Available at: *http://www.andy-roberts.net/coding/aconcorde*

Sparknotes (2015) Oliver Twist [online]. [Accessed 25 April 2016]. Available at: *http://www.sparknotes.com/lit/oliver/context.html*