Evaluating Candidate Answers Based on Derivative Lexical Similarity and Space Padding for the Arabic Language

Samah Ali Al-Azani, C. Namrata Mahender

Evaluating candidate Answers based on derivative lexical similarity and space padding for the Arabic language

Samah Ali Al-azani*, C. Namrata Mahender
Department C.S. and I.T, Dr. Babasaheb Ambedkar Marathawada University, Aurangabad, Maharashtra, India
[1]alazani183@gmail.com
Department C.S. and I.T, Dr. Babasaheb Ambedkar Marathawada University, Aurangabad, Maharashtra, India
[2]nam.mah@gmail.com

Abstract. Character difference represents one of the most common problems that can be occurred when students try to answer questions of fill in the gaps or one-word answer that is needed mostly to one word as the answer. To improve the evolution of the student answer using Hamming distance, we proposed Hamming model tried to solve the drawbacks of the standard Hamming model by applying the stemming approach to achieve derivative lexical similarity and applying the space padding to deal with unequal lengths of the texts.

Keywords: Hamming, lexical similarity, derivatives, questions Answering system

1 Introduction

1.1 Question answering system

Questions Answering system is usually a challenging task, is a software engineering discipline inside the fields of data recovery and characteristic language handling (NLP), which is involved in building frameworks that consequently answer addresses presented by people in a characteristic language. The fundamental purpose of the questionnaire is to provide direct answers to users' questions in the native tongue [1] [2]. This has the incredible position of aiding clients find the answers they want without having to search for large reservoirs of acquaintance. Not at all like web index, for example, Google and yahoo, which permit the client to recover web pages or on the other hand reports that less sensitive to the specified keywords while leaving the function of extracting hyperlinks inactive and finding required verses from clients, QA programs provide the user with the most relevant details in answering their inquiries in the section or at the sentence level [3].

1.2 Challenges of Arabic QAS

Arabic is a Semantic language [4]. It has a population of over 422 million people worldwide. The first language of the Arab and the official language of the United Nations [5]. It is the third most important international language after English and French. The Arabic language has a very rich combination of special features that the computer is difficult to perform [6]. This advantage has created many challenges that researchers have to deal with differently. This section looks at many of these challenges:

1. No capital letters.
2. Lack of linguistic resources.
3. Optional short vowels.
4. Free order senetences.
5. Arabic an inflectional language (word = root +affixes (prefix , infex , suffix).

2 Text Similarity algorithms

A similarity measure is a work that allocates an actual number between 0 and 1 to the whole document. Zero esteem implies that the reports are different completely, whereas one demonstrates that the reports are identical essentially. Vector-based models have been utilized for computing the similarity in files, customarily. The various features presented in the files are given to by vector-based models. Text similarity measurement shows a development role that connected research and all applications in many tasks such as text classification, information retrieval, clustering document, and question answering system, text summarization, detection and correction, machine translation.

2.1 String-based similarity

String similarity is managed operation on strings sequence and structure of characters. A metric that is applied to measure the distance between the text strings is called a string metric. It is applied to matching strings. There are two types of string similarity functions, character-based similarity functions, and term-based similarity functions.

2.1.1 Character-based Similarity

The character-based similarity is additionally known as the sequence-based or edit distance (ED) dimension. proceeds two characters strings and after that figure the edit distance (counting addition, cancellation, and substitution) between these strings., edit distance is broadly utilized for string coordinating estimation to deal with the current data irregularity information inequality [7]. There are some algorithms in this approach illustrative in Fig (1).

2.1.2 Term-based Similarity

This type is known as token-based since it displays each string as a bunch of tokens. the comparability between these strings can be assessed by controlling sets of tokens, such as words. The most thought behind this approach is to perform two string similarity estimation based on common tokens, compare to its token sets [8]. Term based similarity is best utilized on same length token evaluation. Are a few illustrations of these strategies as Cosine similarity [9], Jaccard similarity [10], Manhattan distance [11], and Euclidean distance [12], Dice's coefficient [13].

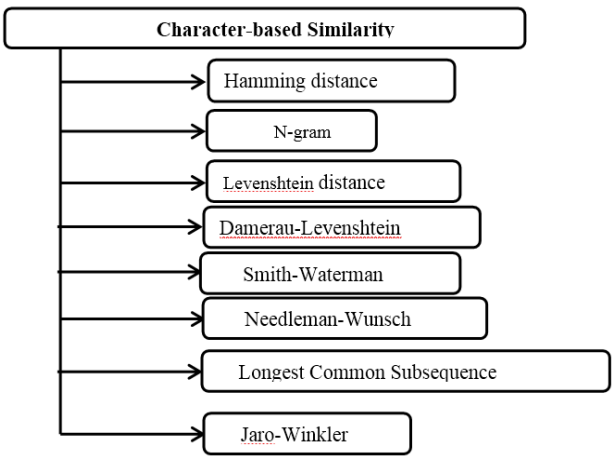

Fig. (1) Examples of Character-based approach.

2.2 Corpus-based Similarity

The corpus-based similarity is applied on a semantic likeness degree, which decides the likeness between the words based on the data picked up from corpora. Text corpus comprises of an expansive and organized set of writings, mostly corpus-based similarity are based on idea established resources, like Wikipedia. There are some types of corpus-based similarity approaches like Semantic Analysis (LSA) [9], Normalized Google Distance (NGD), Explicit Semantic Analysis (ESA) [14].

2.3 Knowledge-based Similarity

A semantic likeness measure that employs data from semantic systems to distinguish the degree of word closeness is known as a knowledge-based likeness measure [15]. Knowledge-based likeness comprises semantic similitude and semantic relatedness. Those concepts have been energetically examined among around the world analysts. Knowledge-based similarity measures divided into the two group's measurements are semantic similarity and semantic relatedness. Measures of semantic similarity have been the focus on words and concepts [16]. The semantic approach employs an express representation of information, for example, the interconnection of realities, the implications of words, and rules to depict conclusions on particular spaces. The pattern of information illustration, by and large, incorporates the rules of eventuality, coherent recommendations, and arranges semantics such as scientific categorization and philosophy. The conspicuousness of word-to-word similitude measurements is due to the asset accessibility and particularly encodes relations between words and ideas (e.g. WordNet), the knowledge-based closeness approach that employments WordNet metaphysics can be categorized into three measures as Fig (2). Semantic relatedness alludes to human judgments of the degree to which a given combination of concepts is related. Semantic relatedness and semantic likeness are two isolated ideas. Semantic relatedness could be a more common idea of the relatedness of concepts, whereas similitude could be an extraordinary case of relatedness that's tied to the resemblance of the concepts.

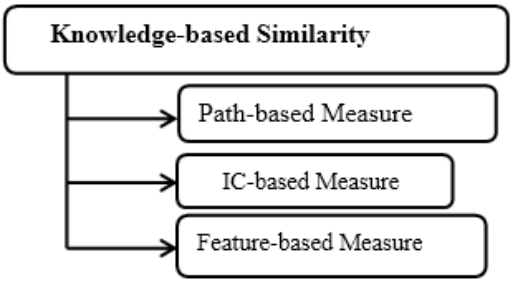

Fig. (2) knowledge-based similarity approach

2.4 Hybrid-based Similarity

The goal of this path is to incorporate the already portrayed approaches, counting string-based, corpus-based, and knowledge-based similitude to reach better results; a metric by receiving their preferences.

Table. (1) Common examples of hybrid Similarity

No	Name of Authors	Examples of hybrid metrics
1.	Monge and Elkan [17]	Assume a recursive matching scheme to compare two long strings.
2.	Wang et al[18]	Employed fuzzy matching between tokens.
3.	Cohen[19]	Apply hybrid metric "soft" TF-IDF similarity uses the Jaro-Winkler.
4.	Lin[20]	Proposed a novel linked data (LD) based on a hybrid semantic similarity measure, called TF-IDF (LD).
5.	Al-Hasan[21]	Proposed a new presumed Ontology-based Semantic Similarity (IOBSS) measure to evaluate semantic similarity.
6.	Atoum and Otoom [22]	Develop a novel hybrid on quantum datasets called text similarity measure (TSM).

3 Proposed Method

Hamming distance is a metric for measure two binary data strings. While match two binary strings of the same length, Hamming distance is the number of bit positions in which the two bits are not the same. In the present work, hamming distance is applied to the data, which is not binary i.e. we have performed distance calculation based on the alphabet, present in the given word if both positions have a similar character the distance is zero else it is predicted to be one.
As this word as an example:

$$\mathbf{dist_{hamming}\ (\text{الزارفة,الزراعة}) = 1}$$

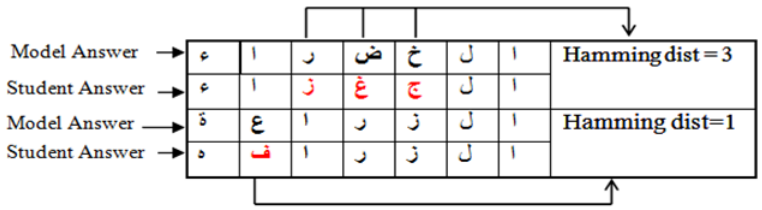

Fig. (3) Hamming Distance examples

The challenge in the hamming distance is it only measures the distance between same length strings. If the lengths of the two strings or comparing strings are different the distance calculation is worthless. So, the standard Hamming model considers that the answer is only correct when both the answers (student answer and model answer) have the same number of characters and there is no any difference (missing, wrong and added characters) in any position. Standard Hamming model also not considers the answer is correct if both the student answer and model answer have lexical similarity and they are derivatives of the same root such as يلعب (play), لعب (played), لاعب (player for male), لاعبة (player for female), ملعب (Playground), لاعبون (players), لعب (playing). Also, there is another issue related to the article of the word ال (the) in the Arabic language, the article ال (the) is always attached with the Arabic word such as اللعب (the playing), the standard Hamming model considers the answer is wrong for the student answer اللعب (the playing) that is not equal with model answer لعب (playing). The proposed Hamming model tried to solve these issues by applying the space padding pre-process to the answer that is smaller than the other answer. padding space makes the text length of both answers is equal. Also, the proposed Hamming model tried to solve the similarity of unequal answer lengths by applying a space padding pre-process Also, the proposed Edit based model tried to solve issue of the lexical similarity for derivatives of the answer by applying another pre-process called stemming pre-process for both the student and model answers.

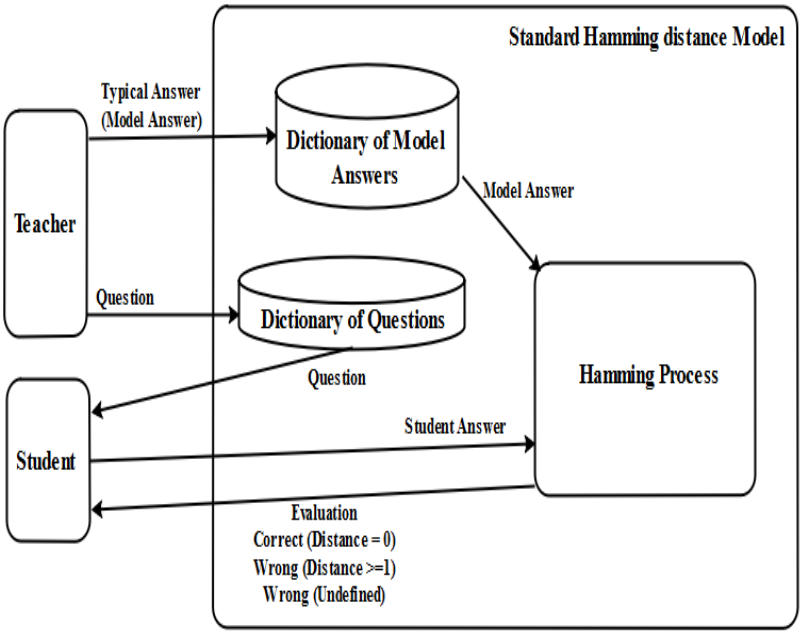

Fig. (4) Standard Hamming Distance

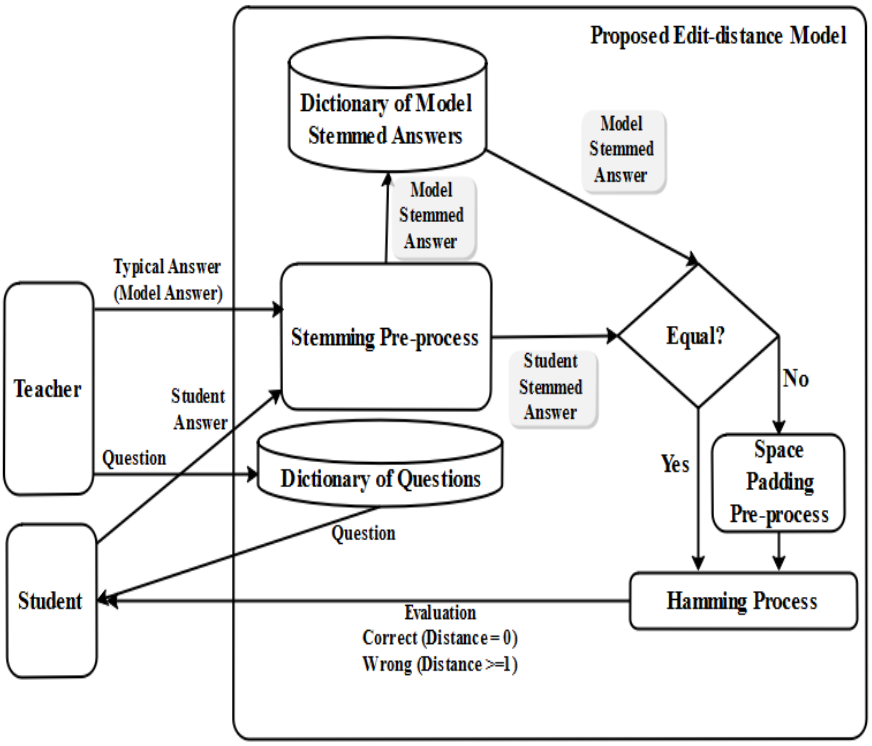

Fig. (5) Proposed Edit Distance.

3.1 Data collection

To collect the required data for the proposed Hamming model, we designed some questions in Arabic language and stored in a ***dictionary of questions*** such as ماهي الحرفة التي كان يعمل بها اجدادنا القدماء في اليمن ؟ (What is the craft that our ancient ancestors used to work in Yemen?). We supposed the typical answer (model answer) for this question is الزراعة (farming), and we had a number of students answer the question (approximately 60 students), 20 students answered الزراعة (farming), 10 students answered مزرعة (farm), 7 students answered مزارعون (farmers), 3 students answered

مزارع (farmer) , 10 students answered رياضة (sport) and 10 students answered زرا (the word is missing letters), and so on for other questions.

3.2 Pre-processing stage

The proposed model used two pre-processes: ***stemming pre-process*** and ***space padding pre-***process. The stemming pre-process takes both the answers for the student and the model to remove the article ال (the) from the answer, and to return all derivatives of the word (the answer) to the root of the word. The derivatives of the answer have lexical similarity with the same meaning. The ***stemmed model answer*** will be stored in the ***dictionary of the model answers***. The proposed model will use the root of the word to make exactly lexical similarity between the student answer and typical answer. The space padding is applied to the small length of the answer to become equal in length to the other answer. To understand the padding approach, we will investigate cases of standard Hamming distance model as follows:

There are three cases for the standard Hamming distance:

- ***First case (normal case):*** when the lengths of both texts are equal. For example, text1 has 7 characters, and text2 has also 7 characters. So, the Hamming distance will have 2 parameters as follows:

  Hamming (text1, text2)

  Hamming operation will take the first parameter (text1) to compute its length (Here 7), and makes 7 comparisons between the characters of the two texts. The comparisons here are *possible* and *sufficient* to get *correct defined Hamming distance* value between 0 to 7 because the text1 string is equal in length to the text2 string. So, there is no need to a space padding where both the texts have the same length.

- ***Second case (issue case):*** when the lengths of both texts are un-equal. For example, text1 has 5 characters, and text2 has 7 characters. So, the Hamming operation will make 5 comparisons between the characters of the two texts. The comparisons here are *possible* but *insufficient* to get correct defined Hamming distance, where *wrong defined Hamming distance* value between 0 to 5 is produced, because the available comparisons (5 comparisons) are lesser than the text2 length (7 characters). So, there is need to two space paddings for text1 to be both the texts have the same length (7 characters).

- ***Third case (second issue):*** Also, when the lengths of both texts are un-equal. For example, text1 has 7 characters, and text2 has 5 characters. So, the Hamming operation will make 7 comparisons between the characters of the two texts. The comparisons here are *impossible*, so *undefined Hamming distance* is occurred due to a *string index out of range*, where the required comparisons (7 comparisons) are greater than the text2 length (5 characters). So, there is need to two space paddings for text2 to be both the texts have the same length (7 characters).

  The space padding pre-process is not needed if both the answers have exactly the same root characters, but it is necessary if both the answers have different root characters.

3.3 Processing Stage

The main process of the proposed process is the ***Hamming process*** that takes both the answers of the student and the model as ***stemmed answers*** without space padding in case of the same root resulted from derivatives of the typical answer, and with space padding in case of different roots resulted from different answers to perform the Hamming operation for strings. Therefore, there is no possibility that the two answers are different in length; also, in most cases the proposed model

returns 0 value as a distance for derivative words. The proposed model gives equal or greater than 1 as a distance if there is missing or/and wrong in characters.

3.4 Algorithm of the proposed Hamming model:

Step 1: Give the question to the student from questions dictionary.
Step 2: Take the student and model answers and perform Stemming process on them.

```
Set st = ISRIStemmer()
Perform stdent_answer = st.stem(input("Answer: "))
Perform Stemming operation manually for model answer
and stored in model answer dictionary.
```

Step 3: perform the space padding on the stemmed answers

```
if len(stdent_answer)>len(value):
    m=len(stdent_answer)-len(value)
    value=value+(" "*m)
if len(stdent_answer)<len(value):
    m=len(value)-len(stdent_answer)
    stdent_answer=stdent_answer+(" "*m)
```

Step 4: Perform Hamming operation on the stemmed padded answers

```
r=hammingDist(stdent_answer,value)
```

Step 5: Hamming distance returns 0 value for correct answer and >=1 for wrong answer

4 Result and Discussion

The result of the proposed edit-based model that is applied on some questions with 60 students is shown in the table below. The standard Hamming model achieved 33.3% correct answer with 0 correct defined distance value, and achieved 58.3% wrong answer with greater than or equa1 to 1 wrong defined distance value, and achieved 8.3% wrong answer with undefined distance value. But The proposed edit-based model achieved 75% correct answer with 0 correct defined distance value, and achieved 25% wrong answer with greater than or equa1 to 1 correct defined distance value.

Table (2). Comparison between standard Hamming method and proposed Hamming method.

(question) ماهي الحرفة التي كان يعمل بها اجدادنا القدماء في اليمن ؟
model answer: الزراعة (7 characters) (farm)

Number of Students	Student answer	Distance		Evaluation	
		standard Hamming model	proposed Edit based model	standard Hamming model	proposed Edit based model
20	الزراعة (7 characters)	0	0	Correct	Correct
10	مزرعة (5 characters)	5	0	Wrong	Correct
7	مزارعون (7 characters)	6	0	Wrong	Correct
3	مزارع (5 characters)	4	0	Wrong	Correct
5	رياضة (5 characters)	5	3	Wrong	Wrong
10	زرا (3 characters)	3	1	Wrong	Wrong
5	المزروعات (8 characters)	Undefined	0	Wrong	Correct

Fig (5). distance of the standard Hamming model

Fig (6). distance of the proposed edit based model

5 Conclusion and Future Work

The result of the proposed edit-based model that is applied on some questions with 60 students it's in details on table. (2). The standard Hamming model achieved 33.3% correct answer with 0 correct defined distance value, and achieved 58.3% wrong answer with greater than or equa1 to 1 wrong defined distance value, and achieved 8.3% wrong answer with undefined distance value. But The proposed edit-based model achieved 75% correct answer with 0 correct defined distance value, and achieved 25% wrong answer with greater than or equa1 to 1 correct defined distance value.

6 Reference

1. Bassam .H , Salem .A, steven .L and Martha .EM, "Experimenting with a Question Answering System for the Arabic Language", Computers & the Humanities, Vol. 38, pp. 397-415, Nov 2004.
2. K. Arai and A. Handayani "Question Answering System for an Effective Collaborative Learning", International Journal of Advanced Computer Science and Applications, Vol. 3, pp.60-64.
3. A. Allam and M. Haggag, "The Question Answering Systems: A Survey," International Journal of Research and Reviews in Information Sciences (IJRRIS), Vol. 2, No. 3, September 2012.
4. K. Ray, Santosh, and Shaalan, Khaled. "A Review and Future Perspectives of Arabic Question Answering Systems" IEEE Transactions on Knowledge and Data Engineering 28 (2016): 3169 - 3190.
5. Bakari, W., Bellot, P., & Neji, M. (2015). Literature Review of Arabic Question-Answering: Modeling, Generation, Experimentation and Performance Analysis. In Flexible Query Answering Systems 2015 (pp. 321-334).
6. Amit Mishra, Sanjay Kumar Jain," A survey on question answering systems with classification ", Journal of King Saud University – Computer and Information Sciences (2016) 28, 345–361
7. L. Gravano et al., "Approximate string joins in a database (almost) for free," in VLDB, 2001, vol. 1, pp. 491–500, available at http://www.vldb.org/conf/2001/P491.pdf.
8. M. Yu, G. Li, D. Deng, and J. Feng, "String similarity search and join a survey," *Front. Comput. Sci.*, vol. 10, no. 3, pp. 399–417, Jun. 2016, doi: https://doi.org/10.1007/s11704-015-5900-5.
9. A. Bhattacharya, "On a measure of divergence of two multinomial populations," *Sankhya. v7*, pp. 401–406.
10. P. Jaccard, "Étude comparative de la distribution florale dans une portion des Alpes et des Jura," *Bull Soc Vaudoise Sci Nat*, vol. 37, pp. 547–579, 1901.
11. E. F. Krause, *Taxicab geometry: An adventure in non-Euclidean geometry*. Courier Corporation, 1975.
12. J. H. Friedman, "On Bias, Variance, 0/1—Loss, and the Curse-of-Dimensionality," *Data Min. Knowl. Discov.*, vol. 1, no. 1, pp. 55–77, 1997, doi: https://doi.org/10.1023/A:1009778005914.
13. T. K. Landauer and S. T. Dumais, "A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge.," *Psychol. Rev.*, vol. 104, no. 2, pp. 211–240, 1997, doi: https://doi.org/10.1037/0033-295X.104.2.211.
14. E. Gabrilovich and S. Markovitch, "Computing semantic relatedness using Wikipedia-based explicit semantic analysis.," in *IJcAI*, 2007, vol. 7, pp. 1606–1611, available at http://www.aaai.org/Papers/IJCAI/2007/IJCAI07-259.pdf.
15. R. Mihalcea, C. Corley, C. Strapparava, and others, "Corpus-based and knowledge-based measures of text semantic similarity," in *AAAI*, 2006, vol. 6, pp. 775–780, available at: http://www.aaai.org/Papers/AAAI/2006/AAAI06-123.pdf.

16. A. Budanitsky and G. Hirst, “Evaluating WordNet-based Measures of Lexical Semantic Relatedness,” *Comput. Linguist.*, vol. 32, no. 1, pp. 13–47, Mar. 2006, doi: https://doi.org/10.1162/coli.2006.32.1.13.
17. A. E. Monge, C. Elkan, and others, “The Field Matching Problem: Algorithms and Applications.,” in *KDD*, 1996, pp. 267–270, available at : http://www.aaai.org/Papers/KDD/1996/KDD96-044.pdf.
18. A. E. Monge, C. Elkan, and others, “The Field Matching Problem: Algorithms and Applications.,” in *KDD*, 1996, pp. 267–270, available at : http://www.aaai.org/Papers/KDD/1996/KDD96-044.pdf.
19. J. Wang, G. Li, and J. Fe, “Fast-join: An efficient method for fuzzy token matching based string similarity join,” in *2011 IEEE 27th International Conference on Data Engineering*, 2011, pp. 458–469, doi: https://doi.org/10.1109/ICDE.2011.5767865.
20. C. Lin, D. Liu, W. Pang, and Z. Wang, “Sherlock: A Semi-automatic Framework for Quiz Generation Using a Hybrid Semantic Similarity Measure,” *Cognit. Comput.*, vol. 7, no. 6, pp. 667–679, Dec. 2015, doi: https://doi.org/10.1007/s12559-015-9347-7.
21. M. Al-Hassan, H. Lu, and J. Lu, “A semantic enhanced hybrid recommendation approach: A case study of e-Government tourism service recommendation system,” *Decis. Support Syst.*, vol. 72, pp. 97–109, Apr. 2015, doi: https://doi.org/10.1016/j.dss.2015.02.001.
22. I. Atoum and A. Otoom, “Efficient Hybrid Semantic Text Similarity using Wordnet and a Corpus,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 7, no. 9, pp. 124–130, 2016, doi: 10.14569/IJACSA.2016.070917, available at: http://thesai.org/Publications/ViewPaper?Volume=7&Issue=9&Code=ijacsa&SerialNo=17.