JOURNAL OF SOFT COMPUTING AND DATA MINING
e-ISSN: 2716-621X

JSCDM

Vol. 6 No. 1 (2025) 107-126
https://publisher.uthm.edu.my/ojs/index.php/jscdm

LughaNet: Automated Arabic WordNet Construction and Evaluation Using Semantic Question Similarity

Ammar Al-Edhari[1], Mohsen Kahani[1]*

[1] *Department of Computer, Faculty of Engineering, Ferdowsi University of Mashhad, 9177948974 Mashhad, IRAN*

*Corresponding Author: Kahani@um.ac.ir
DOI: https://doi.org/10.30880/jscdm.2025.06.01.008

Article Info

Received: 27 January 2025
Accepted: 14 June 2025
Available online: 30 June 2025

Keywords

WordNet construction, Arabic WordNet, synonyms extraction, Arabic semantic question similarity

Abstract

Several efforts have been undertaken to enhance the Arabic lexicon and address the limitations of Arabic WordNet. However, the development of a comprehensive Arabic WordNet (AWN) in previous work has faced significant challenges, including limited coverage compared to English WordNet, reliance on incomplete bilingual dictionaries, and the inherent complexities of the Arabic language, such as lexical ambiguity and morphological richness. Traditional machine translation methods have proven inadequate in addressing these issues, particularly in low-resource settings where large-scale parallel corpora are scarce. To overcome these limitations, this study introduces LughaNet, an automated Arabic WordNet developed through six key stages: (1) aligning Princeton WordNet (PWN) synsets with Arabic words using bilingual dictionaries and Large language machine translation models ; (2) The most frequent word is selected as the optimal translation, and incorrect translations are refined and eliminated using BERT and cosine similarity; (3) extracting Arabic words from resources such as Wikipedia and the existing Arabic WordNet; (4) applying NLP methods, including Skip-gram with AraVec 2.0 embeddings, to extract synonyms from Arabic Wikipedia; (5) enhancing synonym selection accuracy using a pre-trained BERT model and cosine similarity; and (6) translating PWN glosses and examples into Arabic. This process produced 85,991 synsets, with evaluations indicating 64.23% coverage of dictionary terms and demonstrating LughaNet's effectiveness in Arabic Semantic Question Similarity (ASQS) tasks, achieving an accuracy of 64.11%, a precision of 57.57%, a recall of 79.02%, and an F1 score of 66.61%, surpassing the original Arabic WordNet's F1 score of 56.62%. These results highlight the potential of LughaNet as a valuable resource for Arabic NLP research and applications.

1. Introduction

WordNet, which was promoted in 1985 by Fellbaum [1], can be defined as a thorough linguistic database which can arrange words into groups including verbs, nouns, adverbs and adjectives, in addition to hierarchically sorting these words according to their notions. Words that share definite notions, usually denoted as 'literals,' are sorted into 'synsets' whenever these words have similar meanings. WordNet features a complicated synsets network that is interconnected with different types of connections including semantically and lexically, such as derivational, meronymy, antonymy and hypernymy/hyponymy connections. In several tasks in natural language processing (NLP), WordNet was used widely, including information retrieval [2], semantic similarity and relatedness [3, 4], word sense disambiguation [5], and cross-linguistic applications [6]. The use of WordNet in various applications highlights its important fundamental impact in the NLP area [7].

Capturing the difference in meanings of words in English led scholars to develop the Princeton WordNet (PWN) which featured a detailed framework. Around 207 thousand unparalleled word-sense pairs as well as around 117,000 synsets are included in the latest version 3.1. After the unveiling of PWN, there was progress in developing WordNet in different languages. Many of these WordNets use Princeton WordNet (PWN) as a foundational resource. They translate its synsets and expand them as needed. For instance, EuroWordNet (EWN), developed by Vossen [8], covers seven European languages. While Tufis et al. led a wide WordNet (BalkaNet) in the Balkan region [9]. The Global WordNets Association[1] provides updates on the progress of various WordNets.

Due to the high costs associated with the manual construction of the English WordNet, both in terms of time and expert involvement, researchers increasingly favor automatic approaches for developing WordNets in other languages. These methods typically rely on a pre-existing WordNet from a high-resource language, using it as a backbone to streamline and simplify the development process [10]. The Arabic WordNet [11] employs a method similar to EuroWordNet, where researchers manually extract collections of concepts to enhance alignment with other WordNets. Nevertheless, it was highlighted in several studies the importance and necessity for its construction to be automated. As there is continued development in tools and resources, this challenge is still important. Moreover, there are great challenges when it comes to the Arabic language due to its ingrained intricacy, which includes its abundant inflectional and derivational morphology, making the task very complicated to extract meaningful information essential for building a complete database for linguistics [12].

Several studies reported a great effort in improving and tackling the challenges of using the Arabic language in WordNet. A study reported using the Khaleej-2004 corpus to encompass named structures [13], while another study reported the use of corpus-based techniques, in the attempt to leverage using other tools including Word2Vec and GraPaVec [14]. Additional attention was paid to improving AWN V1, which was achieved by presenting new lemmas, synsets and phrases as samples as well as a native speaker to validate and ensure that the quality of the synset has been enhanced [15]. However, despite all the progress in developing WordNet, it still contains around 11 thousand Arabic synsets, making it very limited. This number is very low as the Arabic language is known for its considerable depth in morphological and semantic aspects. In addition, WordNet still cannot be compared with PWN, which contains 117,000 synsets.

Even with all the progress in NLP, some studies suggested the need to integrate WordNet with other models, such as BERT. The main advantages of WordNet as compared to other models are the ability to offer a structured semantic framework and highlight the connection between words including antonym, synonymy and hyponymy. For example, the "TaxoLLaMA" model explores large language models' (LLMs) capabilities in capturing lexical-semantic knowledge from WordNet, achieving state-of-the-art results across 16 tasks and excelling in zero-shot performance [16]. Another study utilized WordNet alongside BERT to enhance emotion-related word collection for emotion recognition tasks [17]. Additionally, research combining WordNet with BERT demonstrated improved contextual embeddings and model accuracy, affirming WordNet's enduring value in NLP [18].

Given the challenges and ongoing efforts in developing the Arabic WordNet (AWN), the continuous expansion of Arabic language resources, such as Wikipedia and diverse dictionaries, combined with advancements in Natural Language Processing (NLP) technologies—particularly the emergence of large pre-trained models like MBART50 for multilingual machine translation or BERT—presents significant opportunities for creating a more precise and comprehensive Arabic WordNet. These advancements in technology can facilitate the automatic construction of AWN, enhancing both its linguistic depth and breadth.

We address this gap by presenting the key contributions of this work as follows:

- Automated Construction of LughaNet: This study introduces LughaNet, an automated Arabic WordNet, to address key limitations in Arabic linguistic resources, such as limited coverage compared to English WordNet, reliance on incomplete bilingual dictionaries, and the complexities of Arabic, including lexical ambiguity and morphological richness. Traditional machine translation methods have struggled in low-resource settings due to scarce parallel corpora. LughaNet overcomes these challenges through a systematic process: (1) aligning PWN synsets with Arabic words using bilingual dictionaries and machine translation models; (2) refining translations by selecting the most frequent word and eliminating incorrect ones using BERT and cosine similarity; (3) extracting Arabic words from Wikipedia and existing resources; (4) applying NLP techniques like Skip-gram with AraVec 2.0 for synonym extraction; (5) enhancing synonym accuracy with BERT and cosine similarity; and (6) translating PWN glosses and examples into Arabic. LughaNet improves coverage, accuracy, and usability for Arabic NLP tasks, offering a robust solution for automated WordNet construction.
- Evaluation of Coverage and Usability of LughaNet: This study evaluates LughaNet's coverage and usability. The coverage analysis compares terms from an Arabic dictionary with those included in LughaNet. Usability testing applies LughaNet as a knowledge base to capture and utilize the meanings

[1] http://globalwordnet.org

and semantic relationships inherent in Arabic terms, specifically in the context of ASQS within a general domain. To the best of our knowledge, this is the first study to utilize a knowledge base for ASQS, incorporating synonyms and additional semantic relationships.

The remaining parts of this paper are organized as follows: Section two provides an overview of the Arabic WordNet, detailing its structure and identifying key limitations. Section three reviews related work, emphasizing WordNet development across various languages and highlighting gaps specific to the Arabic WordNet. Section four outlines the methodology for constructing LughaNet, leveraging resources such as bilingual dictionaries, pre-trained models, and Arabic Wikipedia. The quantitative analysis of LughaNet is presented in section five in order to evaluate the performance and usability testing in Arabic Semantic Question Similarity (ASQS) tasks. Finally, Section Six presented the conclusion of this work, in addition to proposing future research work.

2. Arabic WordNet

WordNet can be defined as a thorough linguistic database which can arrange English words with the meaning of these words in such a way that is a different format than the traditional dictionary. It can arrange these words into synonymous groups called synsets, including verbs, nouns, adverbs and adjectives. WordNet features a complicated synsets network that is interconnected with different types of connections including semantically and lexically [1]. Arabic WordNet (AWN) was developed following the exact structure in the WordNet for the English language but designed only to serve the Arabic language. It comprises several main units, including links, words, forms and elements. The first main unit is the Elements, which are classified as the conceptual units, including synsets, separate cases and ontology classes. The definite meanings are represented by the word units, while the database is enriched by the form units. The relationship between different synsets is represented by the link units which are classified according to their connection kinds, such as semantic, lexical, or lexico-semantic as well as nouns, verbs, adverbs, and adjectives [11].

Princeton WordNet (PWN) was used as a fundamental model in developing AWN and it has been studied widely [19]. A top-down approach was used, which began with the PWN's core elements being translated and then expanded by comprising additional detailed notions. The first launched version of AWN contained 21,813 words with a total of 9,698 synsets. The sunsets were connected through around 144 thousand links with six different forms of semantic interconnections, including metonymy as well as hyponymy [20]. The next version of AWN contained around 24 thousand words with 296 synsets as well as 22 different forms of connections totalling around 162 thousand interconnections. Various named units, as well as several forms, were incorporated in this version, such as the words' root and irregular plurals. Nevertheless, PWN is still superior to AWN as it contains around 18 thousand synsets even with all the progress in enhancing AWN [21].

3. Related Works

3.1 Efforts in Constructing and Expanding WordNets Across Various Languages

Studies have enhanced WordNet by introducing new information layers to improve its functionality in specific contexts. For instance, Maziarz and Rudnicka [22] focused on expanding WordNet with gloss and polysemy links to better recognize evocation strength, emphasizing associations between word senses beyond standard semantic relations. They developed four versions of the WordNet graph, optimized them using Dijkstra's algorithm, and tested these versions on datasets containing 2,000, 10,000 and 108,000 evocation pairs. Their optimized configuration, labelled as wn+g+polySC, achieved a Pearson correlation of 0.265 for the Sim2 measure, significantly improving the recognition of evocation strength on 107,000 pairs. In another study, Kanika et al. [23] enriched WordNet by adding subject-specific out-of-vocabulary terms using Wikidata. WodrNet was enhanced by adding words around 3.7% to be used for specific fields. This was achieved by using the AI students' classroom notes at Netaji Subhas Institute of Technology to extract and recognise the missing words. This is followed by generating figures from Wikidata aiming to establish connections between the words and the figures and then following with the integration process in WordNet.

Another study reported using sememe information from HowNet to develop a technique for word auto-integration in order to tackle the old semantic information problem [24]. Chinese Open WordNet Dataset was used, which contains around 62 thousand words and 42 thousand synsets, in addition to using Sogou-T Corpus, which comprises 2.7 billion words. To align words between the Chinese Open WordNet and HowNet, the model achieved its best performance at M=100, with a Hit@5 score of 0.244 and a Hit@100 score of 0.488.

Some studies reported attempts to develop Persian WordNet using unsupervised and automated methods for WordNet development [10, 25]. Personalized PageRank algorithms and Expectation-Maximisation were used as unsupervised methods for auto-construction in WordNet [10]. This method required only a bilingual dictionary and a monolingual corpus, achieving a precision of over 93% and a recall of 50%. However, Persian WordNet was improved by adding more comprehensive and accurate verbal units [25]. Persian verbs were linked to Princeton

WordNet using a bilingual dictionary and a feature set for compound verbs. The accuracy was improved by using a supervised classification system in order to leverage FarsNet with other similar techniques. This resulted in more than 27,000 words and 67,000 word-sense pairs, making Persian WordNet the largest of its kind.

The studies by Leenoi et al. [26] and Samson et al. [27] extended the development of WordNet to non-English languages, specifically Thai and Filipino. The development of WordNet continued for non-English languages, particularly Thai and Filipino [26, 27]. A study reported developing Thai WordNet using Translation Similarity between Thai and English, using the Princeton WordNet database as their foundation [26]. Their methodology involved several steps, including direct translation, word compounding, word transferring, and phrase translation, all validated by Thai linguistic experts to ensure accuracy and proper hierarchical representation. This process resulted in 157,054 words, 117,767 synsets, and 207,239 word-sense pairs. Similarly, Samson et al. [27] focused on creating a Filipino WordNet through a two-way approach combining natural language processing and network science. Using the Corpus of Historical Filipino and Philippine English (COHFIE), comprising 5.37 million unique tokens, they fine-tuned a Filipino RoBERTa model for Word Sense Induction. Additionally, they developed a temporal-multiplex network to analyze word co-occurrence and semantics over time, capturing and updating word senses. This method induced existing senses in 30% of the validation data and generated 9,549 semantic sets.

3.2 Efforts in Constructing and Expanding the Arabic WordNet

The work on enhancing AWN has progressed gradually by using various methods and different technologies. In the early stage, a study reported the expansion of AWN by using resources, i.e. bilingual lexical and various rules morphologically. This was achieved by arranging the sunsets using Bayesian Networks (BNs) and classifiers of decision trees [28]. The recall and precision of using this method were obtained to be 0.27 and 0.60, respectively. These two values were even more enhanced using integrating BNs with heuristics to be 0.12 and 0.71, respectively. This work was expanded upon in another study [29] as data extracted from Wikipedia in Arabic version was incorporated and extended in AWN's Named Entity (NE) as well as linked to PWN. The accuracy for around 4000 words in Arabic was 93.3% by using this technique, which is equal to around 2600 synsets in English.

Later on, another study reported using different translation tools including MFS, Yahoo and Google in order to resolve some issues in AWN [30]. Using this method enhanced AWN by adding 1950 irregular plurals, 3142 new verbs, 433339 named entities from the YAGO ontology, 459 new synsets, and 459 new synsets. Parallel corpora and APIs were used to improve AWN in another two studies [31], [32]. More than 3 million translation APIs, parallel Arabic-English sentences and dictionaries were used to add 12,000 new synsets with a precision of 93% [31]. Similarly, Lachichi et al. [32] linked AWN to PWN, verified hypernyms through Wikipedia, and enriched AWN with 3,740 validated synsets, achieving an accuracy of 0.48. These studies demonstrate the significance of leveraging parallel corpora and reliable APIs in enhancing AWN's coverage and accuracy.

Corpus-based approaches have demonstrated promising results in enhancing the Arabic WordNet (AWN). Lebboss et al. [14] adopted a corpus-based method, extracting semantic clusters from a large corpus using tools like GraPaVec and Word2Vec, achieving an F-score of 82.1%. Similarly, Batita and Zrigui [13] utilized the Khaleej-2004 corpus to introduce new semantic relations into AWN, assessing their impact on a Word Sense Disambiguation (WSD) system. The enriched AWN improved WSD performance, increasing precision to 78.6%, recall to 71.1%, and the F1 score to 74.6%.

Recent studies have focused on integrating the Arabic WordNet (AWN) with modern NLP models to enhance its coverage and functionality. For instance, Badaro et al. [33] developed ArSenL 2.0 by mapping AWN to the English WordNet (EWN) using data from AWN, EWN, SAMA, and Machine Translation tables. Their semi-supervised link prediction approach employed tools like NLTK, ALMOR, and similarity measures, achieving F1 scores of 27.6% for nouns, 17.4% for verbs, and 31.0% for adjectives. Similarly, Lam et al. [34] introduced three methods for constructing AWN: Direct Translation, Intermediate Wordnets, and Intermediate Wordnets with a Dictionary, resulting in 76,322 Arabic synsets. Additionally, Souci et al. [35] leveraged Princeton WordNet (PWN) with Transformers, including AraBERT and MT5, to translate English resources into Arabic, achieving a validation accuracy of 75.4% for 1,000 synsets and proposing updates for approximately 68,000 out of 90,127 candidates. These studies highlight the potential of modern NLP techniques in enriching AWN.

Freihat et al. [15] enhanced AWN V1 by updating 5,554 synsets, adding 2,726 new lemmas, 9,322 glosses, 12,204 example sentences, and identifying 236 lexical gaps with 701 phrasets. They also removed 8,751 incorrect lemmas to improve accuracy. Using AWN V1, PWN browsers, and the Al-Mawrid Al-Qareeb dictionary, the team improved synset quality through task generation, refinement, and validation. Manual evaluation by native speakers and a linguistic expert ensured the accuracy and consistency of these updates.

Alsudais [36] expands ImageNet for the Arabic language by mapping 21,841 synsets to Arabic WordNet (AWN), utilizing 14 million images and 9,916 Arabic synsets. Through direct synset matching and hypernym expansion, 99.9% of ImageNet images (14.19 million) are successfully assigned Arabic labels, creating a large-scale Arabic-labeled dataset for computer vision applications. Omer, et al. [37] developed a new Arabic WordNet

J. of Soft Computing and Data Mining Vol. 6 No. 1 (2025) p. 107-126

tree for semantic matching, enhancing text similarity in Arabic NLP by organizing nouns from Al-Baqarah into a hierarchical tree and applying a semantic similarity formula, achieving 85% precision, surpassing existing methods. Mehdioui, et al. [38] extend Arabic WordNet using a morpho-lexical approach, analyzing 82 verb forms through a three-step derivation process while collecting 11,000 tweets per hour and 100 tweets per automatic collection from Twitter using 30 accounts, along with Facebook comments. Their results demonstrate high accuracy in opinion classification, effectively distinguishing between objective and subjective opinions, with applications in NLP and dialect processing.

Table 1 summarizes these studies by mentioning the methods, strengths, and Limitations of each study. A later analysis will identify the research gaps. The next step will be discussed in Section 3.3 (Research Gap) and needs to be addressed in this study. Recently, there has been an ongoing effort to construct and expand Arabic ontology, suggesting that it remains a persistent challenge in both general domains and specialized fields, such as Islamic studies (e.g., Quran and Hadith) and specific applications like sentiment analysis and opinion mining.

Table 1 *Efforts in constructing and expanding the Arabic Wordnet*

Ref.	Method	Coverage	Strengths	Limitations
[28]	Bayesian Inference and heuristics-based approaches	11,270 synsets	- Maximum precision -Combining both methods' benefits.	-Statistical translation models rely on large, high-quality bilingual datasets.
[39]	Semi-automatic extension of AWN using Wikipedia	Add to AWN 1,142 synsets	High accuracy (93.3%)	Issues with polysemy and diacritic restoration for Named Entities
[30]	Semi-automatic extension of AWN	AWN Synsets: +5.2% AWN Word-senses: +98.0% AWN Distinct Lemmas: +29.0%	AWN coverage has increased by 37,000 times.	.Coverage improvement for synset extensions was low at 5.2%, with non-recursive hyponym extraction and few extracted snippets.
[31]	Machine translation and word alignment	add ~12,000 Synsets	increase in coverage with a precision of 93%	Incomplete translations
[32]	Automatic enrichment using machine translation, Wikipedia, and external resources	add ~3,740 Synsets	Reduces manual effort with multilingual resource	Limited recall (28.57%).
[14]	GraPaVec for semantic clustering	Add 5,807 synsets	GraPaVec outperformed other methods in clustering quality	-sensitive to corpus quality -need for better lemmatization and normalization
[13]	Extended Arabic WordNet	Add 8,550 synsets	Improves performance of WSD	limited number of relations considered
[33]	semi-supervised	AWN in 10,456 synsets EWN, linked ,7,183 lemmas mapped to EWN 3.0 synsets	The method improves link prediction accuracy	noisy translations from machine translation
[34]	Direct Translation, Intermediate Wordnets, Intermediate Wordnets + Dictionary	76,322 sysnset	Simple, direct translation; applicable to any language with a bilingual dictionary	bilingual dictionaries Limited in low resource language
[35]	-Machine translation -AraBERT	68,000 correct sysnet out of 90,127	Transformers model used for refinement	Noise during translation
[15]	PWN alignment, bilingual dictionaries	+ 5,554 synsets	Improved synset quality	lexical gaps ongoing isuue
[36]	-Direct Matching(DM) , -Hypernym Expansion(HE) -Multi-Level Hypernym Matching(MLH)	-DM: 1,219 synsets -HE: 10,462 synsets -MLH: 17,438 synsets	99.9% of ImageNet, and improves Arabic computer vision datasets.	Gender Discrimination in the Arabic Language , Classification Issues in the "Person" Tree
[37]	algorithm for Arabic semantic matching	Covers nouns from Al-Baqarah verses	Provides structured Arabic WordNet using Quranic term	Algorithm is manual-based for noun extraction and tree construction
[38]	Morpho-Lexical Based	________________	Novel approach to verb derivation	Limited coverage, focuses only on verbs,

3.3 Efforts in Constructing and Expanding the Arabic Ontology

Studies have focused on developing general-domain ontologies, such as the one presented in [40], which introduces an Ontology Learning (OL) framework for unstructured Arabic text using AraBERT and deep learning models (SVM, MLP, CNN). This study processes 156 Arabic documents and the OSAC corpus (4763 BBC, 5070 CNN, and 22,429 OSAC samples), analyzing 136,630 tokens, 7931 terms, 6528 concepts, and 845 relationships. The model achieved 91% accuracy in information extraction, with 91.2% precision, 89.7% recall, and 91% accuracy using SVM, confirming AraBERT's effectiveness in Arabic ontology learning.

Another study Kahlawi [40] enhances DBpedia's Arabic text annotation by addressing inaccurate type assignments through a new ontology and an algorithm for type verification. Using SPARQL queries, it collects 407 types, later reducing them to 106 for improved classification. It also introduces ATEA, a Python-based tool for entity annotation, achieving 75%-100% accuracy on the ANERcorp dataset. This research contributes to Arabic NLP by making DBpedia more reliable for structured Arabic data extraction.

Studies focusing on the Islamic domain include Zouaoui and Rezeg [41], who developed AraFamOnto, an Arabic ontology-based system for automating Islamic inheritance calculations. Using 50 real-life family cases, it extracts heirs' details and calculates shares based on Sharia law, identifying 22 heirs with semantic reasoning to determine primary and secondary heirs. In a test case, wives received 6.25% each, sons 19.44%, and daughters 9.72%. Altammami, et al. [42] developed a joint Quran-Hadith ontology, evaluating two Quran ontologies and their compatibility with Bukhari Hadith headings using Protégé, SPARQL, and Camel Tools. Ontology B emerged as the best candidate, covering 47% of Hadith headings, while Ontology A covered 25%, with 61.5% and 56.9% overlap, respectively. Kamran, et al. [43] introduced SemanticHadith, an ontology-driven knowledge graph structuring 34,458 Hadiths from six collections, linking them to 338 Quranic verses and 6735 narrators. Built with Protégé, Apache Jena, SPARQL, OWL API, and LIMES/OpenRefine, it improves search, retrieval, and integration with DBpedia, Wikidata, and Getty vocabularies, enhancing access to Islamic knowledge.

The study focusing on specific applications such as sentiment analysis and opinion mining by Khabour, et al. [44] proposes an ontology-based Arabic sentiment analysis method, utilizing 33,000 reviews across multiple domains and a hotel-specific dataset of 15,000 reviews. It extracts 203 concepts across six ontology levels, leveraging Protégé, Apache Jena, and Python NLP tools to improve feature extraction and sentiment classification, achieving 79.20% accuracy and 78.75% F-measure, outperforming lexicon-based approaches by 4.5% and enhancing domain-specific sentiment detection. Similarly, Boulaalam and El Hannach [45] focus on constructing an Arabic linguistic ontology by integrating morphological, syntactic, and semantic components using AI and deep learning. Their study extracts domain-specific concepts, applies semantic reasoning, and improves text classification and sentiment analysis using NOOJ and Apache Jena. Additionally, Alsemaree, et al. [46] introduce LSAnArTe, a lexicon-based sentiment analysis framework for Arabic, analyzing 10,769 tweets on Saudi coffee brands. Using AraSenTi and Qalasadi, it preprocesses text and classifies sentiment with 93.79% accuracy, surpassing Amazon Comprehend (51.90%). The system effectively handles Arabic dialects and enhances precision and recall in sentiment detection.

3.4 Research Gap

Constructing a comprehensive WordNet remains a challenging and ongoing task, particularly for low-resource languages such as Arabic. Despite efforts to enhance and expand the original WordNet, progress in developing a fully detailed Arabic WordNet has been limited. This is primarily due to insufficient emphasis on automatic construction methods and the inherent challenges of low-resource languages. The key limitations identified in previous studies are as follows:

- **Limited Coverage**: The current coverage of AWN remains significantly smaller than Princeton WordNet (PWN), which contains 117,659 synsets. This highlights the need for more extensive and automated expansion methods. The best AWN coverage reported by [34] is only 76,322 synsets.
- **Dependence on Bilingual Dictionaries**: Many existing approaches rely heavily on bilingual dictionaries, which are often incomplete for low-resource languages.
- **Traditional Machine Translation**: While traditional machine translation (MT) methods have been used, they often struggle with the complexities of Arabic, such as lexical ambiguity and morphological richness [47].
- **Data Scarcity**: low-resource language suffers from a lack of large-scale parallel corpora, which are essential for training high-quality neural machine translation (NMT) models.

To overcome these limitations, the emergence of Large Language Models (LLMs) presents a promising solution for enhancing the automatic construction of Arabic WordNet (AWN). Unlike traditional machine translation methods, which struggle with lexical ambiguity and morphological richness, LLMs can generate more accurate and context-aware translations, even in low-resource settings [47, 48]. By leveraging pre-trained multilingual models, such as M2M-100 and NLLB, AWN can benefit from vast amounts of monolingual data, improving generalization and translation quality, even when parallel corpora (bitexts) are scarce [49]. These models facilitate knowledge transfer from high- to low-resource languages [50] making them particularly effective for synset mapping, expansion, and enrichment. As a result, AWN can achieve greater coverage, depth, and linguistic accuracy, bringing it closer to the comprehensiveness of Princeton WordNet (PWN). However, despite significant improvements in machine translation due to LLMs, computers still struggle to achieve high-quality translations for many tasks [51]. Thus, the proposed method will take this into account to enhance translation quality.

4. Methods for Constructing LughaNet

The construction of LughaNet involved utilizing the following language resources and knowledge bases:

- Princeton WordNet
- (English-Arabic) bilingual dictionaries
- Pre-trained Models for Multilingual Machine Translation
- Arabic Wikipedia
- Existing Arabic WordNet

The construction of LughaNet involves eleven steps (see Figure 1):

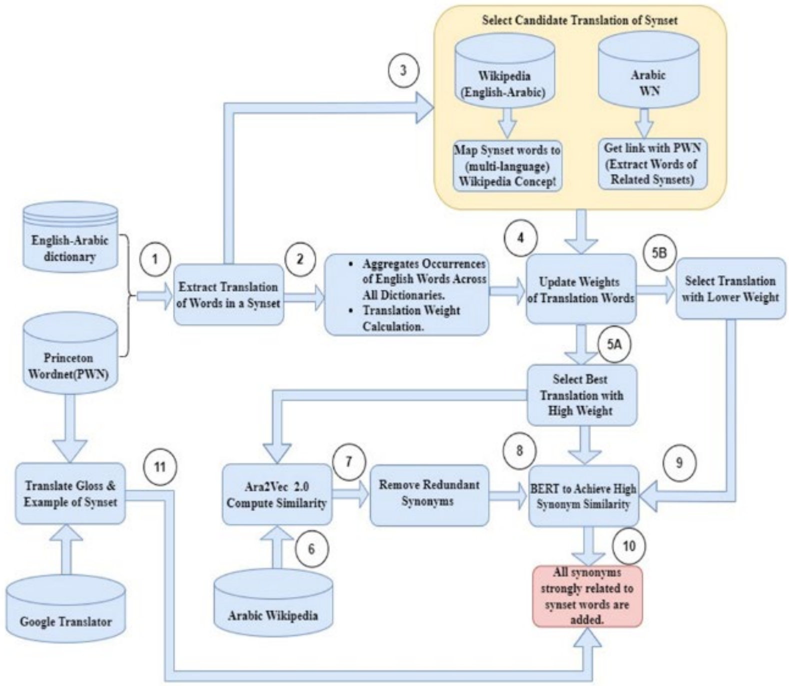

Fig. 1 *The framework for developing LughaNet's synsets*

1) In this step, each word in the synset is translated using two primary approaches:

- Each word in the synset is translated using bilingual dictionaries, including the OmegaWiki database and Wikipedia interlanguage links, which offer pre-mapped terms from English to Arabic.
- Machine translation has significantly improved the handling of low-resource languages through advancements in deep learning and artificial neural networks. Google's introduction of the Transformer model in 2017 represented a major breakthrough, achieving state-of-the-art performance in machine translation tasks at that time. These technologies have greatly enhanced translation quality, increasing interest in addressing challenges associated with low-resource languages. Consequently, modern machine translation systems are now better equipped to manage these languages, enabling solutions for more complex translation tasks [52],[53]. Their performance underscores the promising potential for translating low-resource languages [54]. Arabic, as a low-resource language, has particularly benefited from these advancements and tools [55].

  In this study, we employ four pre-trained models for multilingual machine translation to translate each word in the synset. First, MarianMT is used, a framework incorporating dynamic computation graphs and supporting state-of-the-art neural machine translation (NMT) architectures such as deep RNN and Transformer models [56]. Second, mBART50 is employed, an extension of the mBART model that increased coverage from 25 to 50 languages and introduced multilingual fine-tuning, enhancing translation quality, particularly for low-resource languages [49]. Third, M2M-100, a many-to-many multilingual translation model capable of translating between 9,900 directions across 100 languages, is utilized (see Figure 2) [57]. Fourth, the No Language Left Behind (NLLB) project by Facebook AI is incorporated, aiming to build accurate translation models for 200 languages, with a focus on low-resource languages [59]. Additionally, we employ the Python Google Translation library[2] to complement these models.

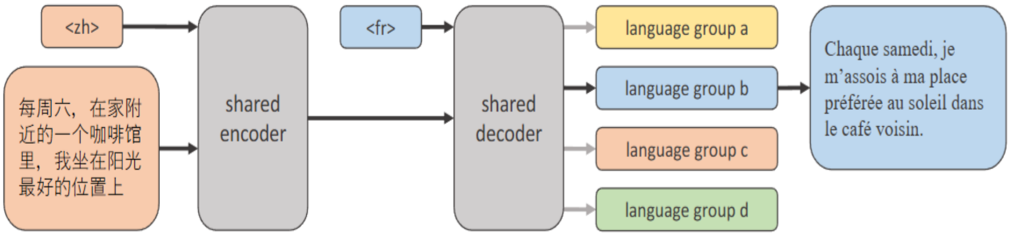

Fig. 2 *M2M-100, translating from Chinese to French*

2) From the previous steps, pairs of words (X, Y) were obtained, where X represents an English word and Y its corresponding Arabic translation. These pairs were merged into a single file to evaluate translation weights based on the frequency of each translation within the word list across all aggregated sources (Step 1). Table 2 illustrates this process using the English word "able" and its various Arabic translations from different sources. For example, "able" is translated as "قادر (qadir)" in sources 1, 2, and 3, resulting in a higher weight of 1.5. In contrast, "يتمكن (yatamakan)" and "يستطيع (yastati)" appear in fewer sources, with weights of 0.5 each. This weighted evaluation determines the most commonly accepted translation, enhancing the accuracy and reliability required for subsequent steps.

Table 2 *Weighted evaluation of Arabic translations for the English word*

English word	Arabic Sources	Arabic translated	weights
	Source 1	قادر (Qadir)	
	Source 2	قادر (Qadir)	1.5
able.a.01	Source 3	قادر (Qadir)	
	Source 4	يتمكن (Yatamakan)	0.5
	Source 5	يستطيع (Yastati)	0.5

3) In this phase of the process, Arabic words linked to specific Princeton WordNet (PWN) synsets are initially extracted from Wikipedia and Arabic WordNets. Using the Wikipedia Python library[3], Arabic equivalents of these PWN concepts are identified and retrieved, provided that an Arabic Wikipedia article for the concept exists. Additionally, the Arabic WordNet, part of the Open Multilingual WordNet[4,5], contains mappings between Arabic words and their English equivalents. This facilitates the acquisition of Arabic synset words corresponding to English synsets, assuming equivalent synsets exist in the Arabic WordNets. For example, the English synset "able.a.01" corresponds to the Arabic word "قادر (qadir)."
4) Using the words obtained from the previous stage (Step 3), the translations were expanded by adding new terms, and the weights of their connections (established in Step 3) were updated. For instance, the translation of "able" was refined, assigning a higher weight to the Arabic word "قادر (qadir)." As shown in Table 2, the weight for "قادر (qadir)" increased from 1.5 to 2.
5) This step involves a two-part process to identify the optimal translation option based on predefined weights:
   - The first part selects the Arabic word with the highest weight from a set corresponding to the English synset, such as "able." For instance, "قادر (qadir)" is identified as the most suitable translation due to its highest weight. This weighting system is crucial as it reflects the relative precision and relevance of each translation candidate, ensuring that the most accurate translations are prioritized. This

[2] https://pypi.org/project/googletrans/
[3] https://pypi.org/project/wikipedia/
[4] https://omwn.org/omw1.html
[5] https://pypi.org/project/wn/

approach improves the accuracy of extracting related synonyms in the subsequent step (Step 6), focusing on the core meaning.

- The second part groups the remaining Arabic words with lower weights, which will be processed in the subsequent steps.

6) The study by Mikolov et al. [59] introduces two innovative frameworks for word representation in a multi-dimensional vector space: the Continuous Bag of Words (CBOW) and the Skip-gram models. The CBOW model predicts a target word based on surrounding context words, without considering the order of words. In contrast, the Skip-gram model predicts surrounding context words from a given target word. Using approaches such as Wikipedia-Skipgram, the study extracted synonyms and related terms by applying Arabic Wikipedia's distributed word embeddings from AraVec 2.0. AraVec is a pre-trained distributed word representation (word embedding) open-source project, as detailed further by Soliman et al. [60].

   AraVec 2.0[6] provides several pre-trained models, including Wikipedia-Skipgram with different dimensions (100 and 300). For instance, using the Wikipedia-Skipgram model with a 300-dimensional vector, the word "يجلس (yajlis)" ("sits") was used as a target word, and the model predicted related synonyms such as "النوم (al-nawm)" (sleep), "الراحة (al-raha)" (rest), "الجلوس (al-juloos)" (sitting), "يستريح (yastarih)" (relaxes), and "يتكئ (yataki')" (leans), with similarity scores of 0.78, 0.76, 0.74, 0.73, and 0.71, respectively.

7) In the preceding step, noise in the form of redundant synonyms was encountered. For instance, along with the correct translation of "حاسوب (hasub)" as "computer," redundant synonyms such as "الكمبيوتر (al-kompyuter)," "بالكمبيوتر (bil-kompyuter)," and "والكمبيوتر (wal-kompyuter)" were identified. These terms essentially mean "computer" but include different prefixes: "ال" for "the," "بـ" for "with," and "و" for "and." To address this issue and eliminate redundancies, prefixes and suffixes were removed, ensuring that only one synonym was retained for each word.

8) Furthermore, deep learning methodologies were leveraged, particularly the pre-trained BERT model [61], which is based on transformer architectures. Cosine similarity was applied to identify synonyms with high semantic similarity, refine word sense disambiguation, and filter out unrelated synonyms. For example, the Arabic word for 'storm' is "عاصفة (asifa)." However, using AraVec, one of the identified synonyms was "الهوجاء (al-hawja')," which translates to 'fierce' and is not an actual synonym. This inappropriate synonym was subsequently removed following the application of this approach.

9) To acquire more accurate synonyms and exclude irrelevant ones, words with lower weights (as identified in Step 5.B) are filtered out. For this purpose, the BERT model and cosine similarity measures are applied. For instance, in one example, the word "abaxial" ("بعيد عن المحور" (ba'id 'an al-mihwar)) yielded synonyms such as "مباعد للمحور" ("muba'id lil-mihwar" – "away from the axis") and "ظهري" ("zahri" – "dorsal") due to their higher similarity scores. During machine translation, however, "abaxial" was incorrectly translated as "أبوظبي" ("Abu Dhabi"), the capital of the United Arab Emirates. This approach avoids such errors that frequently occur in machine translations.

10) The final word list was refined to include terms from the previous stage that achieved a similarity score of 0.80 or higher with other words.

11) The glosses and examples for each synset were translated using Google Translate. For instance, as shown in Table 3, the English synset numbered 2764245, representing "flare_up.v.01," was translated from its original gloss into Arabic. These translated glosses and examples were subsequently incorporated into LughaNet and aligned with their respective synset numbers.

Table 3 *Gloss and example before and after translation to Arabic*

English synset number		2764245
English synset words		flare_up.v.01
Glosses and examples before translating	Gloss	Ignite quickly and suddenly, especially after having died down.
	Example	the fire flared up and died down once again.
Glosses and examples after translation	Gloss	تشتعل بسرعة وبشكل مفاجئ، خاصة بعد أن تخمد
	Example	اشتعلت النار وخمدت مرة أخرى

After translating and expanding the synsets, the relationships between Princeton WordNet (PWN) synsets were applied to the newly created Arabic WordNet synsets.

6 https://github.com/bakrianoo/aravec/tree/master/AraVec%202.0/

5. Evaluation and Discussion

5.1 Experiment Configuration

In our study, we conducted hardware experiments using a personal computer with the specifications outlined in Table 4. For the software experiments on constructing LughaNet and testing it with a semantic question similarity dataset, we used Python 3.8.3 language program along with various libraries, including Gensim 4.3.2 [62], Transformers 4.38.1 [63], Scikit-learn 1.3.1 [64], PyTorch 1.12.1+cu113 [65], NLTK[7], beautifulsoup[8], and Wikipedia[9].

Table 4 *Personal computer hardware configuration*

Hardware	Specifications
Operating system	Windows 10 Pro
Operating system version	19045.5371
Central processing unit	AMD Ryzen 7 5800H with Radeon Graphics 3.20 GHz
Physical Memory	32 GB
Graphic processing unit	AMD Radeon (TM) Graphics, NVIDIA GeForce RTX 3070 Laptop GPU

5.2 Quantitative and Automatic Coverage Evaluation

In a quantitative assessment comparing LughaNet with the extensively used Princeton WordNet (PWN), as presented in Table 5, LughaNet is reported to have 85,991 synsets, whereas PWN contains 117,659 synsets. LughaNet includes 102,355 unique words, compared to 155,287 in PWN, indicating a smaller vocabulary. However, the average word count per synset in LughaNet is 3.16, higher than the PWN average of 1.759, suggesting a broader range of synonyms and lexical variations for each concept. Moreover, LughaNet has a lower percentage of single-word synsets (33.81%) compared to PWN (54%), reflecting the linguistic richness and diversity inherent in the Arabic language. While LughaNet has fewer synsets compared to PWN, its higher average number of words per synset (3.16 vs. 1.759) indicates richer synonymy, which is valuable for Arabic NLP tasks. This compensates for the smaller size, making LughaNet highly usable despite the lower synset count. Future work can focus on expanding synset coverage to further enhance its utility.

Table 5 *The qualitative assessment of English WordNet and LughaNet*

	English WordNet	LughaNet
Number of synsets	117,659	85,991
Number of words (unique)	155,287	102,355
Number single-word synsets	54%	33.81%
Average number of words in each synset	1.759	3.16

To evaluate LughaNet's coverage, an automatic assessment was conducted using selected Arabic resources from a dictionary. The Hans Wehr Dictionary of Modern Written Arabic [66], originally published in German in 1952 and later translated into English, served as a key resource for Modern Standard Arabic. This dictionary is organized by root words, facilitating the understanding of the structure and derivation of Arabic terms. The fourth edition, published in 1979, contains approximately 26,000 words and is available in multiple formats, including digital versions, making it highly accessible for students and scholars. Preprocessing techniques, such as tokenization, character normalization, and diacritical sign removal, were applied to the dictionary. Redundant terms were removed while preserving their first occurrences.

For the analysis, string-matching was employed to represent and compare terms from this source with LughaNet. Term Frequency-Inverse Document Frequency (TF-IDF) [67] was utilized for vectorization. The

[7] https://www.nltk.org/
[8] https://pypi.org/project/beautifulsoup4/
[9] https://github.com/goldsmith/Wikipedia

J. of Soft Computing and Data Mining Vol. 6 No. 1 (2025) p. 107-126

similarity between these terms and the synsets in LughaNet was measured using cosine similarity, a widely adopted metric in computational linguistics.

The findings, as presented in Table 6, reveal that LughaNet covers 64.23% of the dictionary terms. This level of coverage indicates that LughaNet can be applied to various natural language processing tasks, such as text summarization, machine translation, sentiment analysis, and information retrieval. However, despite its broad coverage, certain dictionary terms remain unrepresented in LughaNet. Addressing this gap offers an opportunity for further enhancement, particularly by integrating these missing terms.

After assessing LughaNet's coverage, the next phase of this research will focus on its practical applications, particularly in semantic question similarity tasks. This phase will involve experiments to evaluate LughaNet's usability, its impact on accuracy, and a comparison with the original Arabic WordNet for semantic question similarity analysis.

Table 6 *Coverage evaluation of LughaNet using an Arabic dictionary*

Arabic dictionary	Number of terms after preprocessing	Percentage of coverage
Hans Wehr searchable dictionary	22,267	64.23%

The coverage results of LughaNet support its applicability in Arabic NLP tasks. LughaNet shares a similar structure to WordNet in terms of relational representation, making it compatible with existing techniques and NLP applications. For instance, semantic similarity measures such as Wu & Palmer (WuP)[68], path-based similarity [69], and Leacock & Chodorow (LCH)[70] can be applied to LughaNet. Specifically:

- **Wu & Palmer (WuP):** Utilizes the depth of concepts in the hierarchy to measure semantic similarity.
- **PATH**: Measures the shortest path between two concepts to determine their relatedness.
- **Leacock & Chodorow (LCH):** Considers the maximum depth of the hierarchy to compute similarity.

LughaNet can be used in various Arabic NLP tasks. For Word Sense Disambiguation (WSD), it resolves ambiguous words by identifying contextually appropriate senses. Machine Translation (MT), improves translation quality by handling lexical variations and Arabic's morphological richness. For Information Retrieval, it expands queries with synonyms, increasing search relevance. In Text Summarization, it enables the generation of diverse and accurate summaries. For Question Answering (QA), its graph structure enhances semantic matching, improving QA system performance.

5.3 Arabic Semantic Question Similarity (ASQS)

Estimating semantic similarity between text data is a challenging and ongoing research problem in the field of Natural Language Processing (NLP) [71]. Semantic Textual Similarity (STS) is fundamental to various applications, including question answering, semantic search, conversational systems, and information retrieval [72]. For instance, in information retrieval, keyphrase extraction enhances performance by enabling a semantic understanding of queries and indexed documents, thereby improving the accuracy and relevance of search results [73]. Building on the significance of semantic text similarity, question-answering (QA) systems enhance response precision and contextual relevance, delivering consistent and accurate answers to users [74]. Furthermore, question similarity is crucial in QA systems, improving user experience and reducing response times by reusing previously provided answers [75],[76]. This methodology not only streamlines processes but also reduces redundancy, boosting platform reputation and user satisfaction [77].

A notable challenge in semantic question similarity, as highlighted in [78], lies in the variability of word and character counts in texts. This issue is particularly evident on community-driven platforms, where questions are often concise, offering limited representations and minimal word overlap. Despite these limitations, identifying similarities between short texts remains crucial in Natural Language Processing (NLP) across various contexts. The brevity of such texts frequently complicates the accurate assessment of their similarity [79]. This highlights the importance of employing advanced NLP techniques, particularly on platforms like Quora, where recognizing similarities between brief and diverse queries enhances user experience and operational efficiency.

The evaluation of semantic similarity in texts is well-established in languages like English, with high-performing systems achieving over 80% correlation with human judges due to the abundance of Semantic Textual Similarity (STS) data resources. However, this progress is not consistent across all languages. Arabic, for instance, faces significant challenges due to a scarcity of STS resources, dialectal diversity, and complex morphology [80]. The optional use of diacritics in Arabic further increases ambiguity in semantic interpretation, negatively affecting the accuracy of semantic features and representation in NLP models [81]. For example, the undiacritized sentence "درس محمد الدرس" (drs mHmd Aldrs) has two possible meanings: 'Mohammad studied the lesson' (دَرَسَ مُحَمَّدٌ الدَّرْسَ) or 'Mohammad taught the lesson' (دَرَّسَ مُحَمَّدٌ الدَّرْسَ). The absence of diacritical markings renders the sentence ambiguous, and only the context can clarify the intended meaning [82]. As a result, many researchers prefer studying semantic similarity tasks in languages like English, which offer greater resources and less intrinsic linguistic complexity.

Based on the findings in [71], three primary methods have emerged for identifying semantic similarity. The first is a knowledge-based approach, which leverages lexical resources such as dictionaries, thesauri, and ontologies, particularly WordNet. These tools provide structured data crucial for resolving synonymy and ambiguity, facilitating the extraction of additional semantic context and enhancing text processing accuracy [83],[84]. The second method is corpus-based, utilizing word embeddings from extensive corpora. These embeddings are categorized into traditional (e.g., TF-IDF), static (e.g., Word2Vec, GloVe, FastText), and contextualized (e.g., ELMO, GPT-2, BERT) approaches, as outlined in by [86]. The third method involves deep neural network techniques, including Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTMs). Research on Arabic semantic question similarity indicates that most studies have focused on corpus-based and deep learning approaches, highlighting a gap in the utilization of knowledge-based methods for understanding actual meanings and semantic relations in the public domain [76],[78],[85-88]).

An exception to this trend is the work in [89], which developed an ontology-based system for question answering in the Islamic banking fatwa domain. Although this study includes a subtask to test the similarity of two cases, whether queries or questions, its applicability is limited to the Islamic finance and banking sector. Using frames for questions is uncommon, as fatwa questions are typically framed as narratives rather than structured queries. Another study by [90] attempted to utilize knowledge-based approaches but faced limitations in fully leveraging structured relationships within Wikidata and WordNet. The model primarily relied on synonym matching and embeddings, which hindered its ability to capture more complex graph-based relationships between concepts. For instance, terms from Wikidata such as محاضر (educator), معيد (teaching assistant), أستاذ جامعي (university teacher), معلم خاص (private tutor), and معلم (tutor) are not treated as synonyms but as instances of the broader category 'مهنة (profession). This shortcoming highlights the model's inability to utilize relational structures effectively, resulting in a less nuanced understanding of semantic connections beyond simple synonymy.

While previous studies [91-93] show that knowledge-based approaches play an important role in measuring semantic similarity, their use to integrate actual meanings and semantic relationships of keywords in the public domain remains relatively unexplored, particularly in the context of Arabic Semantic Question Similarity. Moreover, these studies have largely focused on synonymy, with limited attention given to other semantic relationships, such as hyponymy and meronymy, which can enhance the understanding of semantic connections between terms. This highlights a notable research gap, as tools and approaches leveraging these methods to measure semantic similarity between general public domain queries in Arabic have been minimally explored.

5.4 Experimentally Test the Usability of Lughanet In The ASQS

This task involves comparing the experimental results of the gold standard Arabic WordNet with the newly constructed Arabic WordNet to evaluate performance improvements. The experimental procedure utilized 15,712 annotated question pairs from the Mawdoo3 Q2Q dataset, part of the NSURL-2019 challenge 8 on semantic question similarity in Arabic. Each pair in this dataset is labelled as either semantically similar (1) or not semantically similar (0) [94]. The evaluation metrics include accuracy, precision, recall, and the F1 measure [95].

To prepare the dataset for subsequent processing and enhance accuracy while minimizing data noise, several Arabic preprocessing steps were applied. These steps included tokenization, which segmented the text into individual word units identified as tokens; stop word removal, which excluded frequently used words that add little analytical value, such as "الى" (to), "من" (from), and "على" (on); text cleaning, which removed unnecessary elements like hashtags, emojis, and hyperlinks; punctuation removal, which eliminated dashes, punctuation marks, and other non-alphabetic characters to streamline the text; and normalization, which standardized the Arabic script by consolidating various character forms. For instance, "آ" ,"أ" ,"إ", and " ا" were unified to "ا", while "ة" ,"ئ" ,"ؤ" ,"ى", and "گ" were standardized to "ه" ,"ء" ,"ء" ,"ي", and "ك", respectively.

In this study, the comparison technique between LughaNet and the original Arabic WordNet utilizes the shortest path similarity measure. This measure is accessed via the function synset1.path_similarity(synset2), which evaluates the similarity between two concepts based on the shortest path connecting them within the "is-a" (hypernym/hyponym) taxonomy, as shown in Equation 1.

$$\text{Similarity}(s_1, s_2) = \frac{1}{1 + Short\ path\ lenght(s_1, S_2)} \tag{1}$$

The evaluation metrics will include accuracy, precision, recall [96], and the F1 measure [97] The following equations illustrate the calculation of these metrics:

- **Accuracy**: It is a widely used metric for evaluating classification algorithms, defined as the ratio of correctly classified instances to the total number of observations (as shown in Equation 2). However, it may be misleading when the class distribution is imbalanced.

$$\text{Accuracy}=\frac{\text{True Positive}+\text{True Negative}}{\text{True Positive}+\text{True Negative}+\text{False Positive}+\text{False Negative}} \tag{2}$$

- **Precision**: It represents the proportion of relevant data points within a selected subset. In other words, it measures how many instances predicted as positive are truly positive. As shown in Equation 3, precision is calculated as the ratio of true positives to the total number of predicted positives (true positives + false positives).

$$\text{Precision}=\frac{\text{True positive}}{\text{True positive}+\text{False Positive}} \tag{3}$$

- **Recall**: It measures the proportion of actual positive instances correctly identified by the algorithm. As shown in Equation 4, it is computed as the ratio of true positives to the sum of true positives and false negatives.

$$\text{Recall}=\frac{\text{True postive}}{\text{True Positive}+\text{False Negative}} \tag{4}$$

- **F1 Score**: Also known as the F-score or F-measure, the F1 Score is a metric that assesses an algorithm's performance by considering both precision and recall. It is calculated using equation(5).

$$\text{F1-score}=2*\frac{\text{Precision} * \text{Recall}}{\text{Precision}+\text{Recall}} \tag{5}$$

After the preprocessing, the analysis advanced to establish similarity between pairs of questions. This process involved evaluating each word within a question to assess its similarity using both the original Arabic WordNet and the enhanced version, LughaNet. The results, as shown in Table 7, demonstrate that LughaNet achieves a notable improvement in the semantic question similarity task. Specifically, LughaNet attained an accuracy rate of 64.11%, outperforming the original WordNet's accuracy of 58.15%. This highlights LughaNet's enhanced capability to classify question pairs as either similar or not.

Additionally, LughaNet outperforms the original Arabic WordNet in precision, achieving a score of 57.57% compared to 48.85%. This suggests that LughaNet is better at reducing false positives. In terms of recall, LughaNet also surpasses the original, attaining 79.02% over 67.35%, reflecting its enhanced capability to correctly identify relevant question pairs—an essential factor in minimizing information loss. Furthermore, LughaNet achieves a better balance between precision and recall, as evidenced by its higher F1 score of 66.61%, compared to the original WordNet's 56.62%.

In conclusion, LughaNet demonstrates potential as a useful tool for semantic question similarity tasks, showing improvements in specific performance metrics. Various natural language processing (NLP) applications could benefit from LughaNet's ability to discern and interpret semantic context within the Arabic language. However, this study has certain limitations. The sense representation between LughaNet and the dataset was based on part of speech. While LughaNet contains several multi-word expressions (34.74%), such as full-time.a.01: "دوام كامل", only single-word representations, like empty.a.01: "فارغ", were utilized in this study. This approach was adopted to simplify the comparison and emphasize the differences between LughaNet and the original WordNet.

Table 7 *Comparative performance metrics between original ARW and LughaNet*

Metric	Original Arabic WordNet	LughaNet (New Arabic WordNet)
Accuracy	58.15%	64.11%
Precision	48.85%	57.57%
Recall	67.35%	79.02%
F1 Score	56.62%	66.61%

5.5 Misclassification Analysis

Table 8 examines cases of misclassification based on semantic similarity, presenting ten instances—five where the true label is 1 and five where it is 0—along with their predicted probabilities. The system selects candidates from the synsets in LughaNet using part-of-speech (POS) tagging, a basic technique in word sense disambiguation. However, this method has limitations, as it does not always capture contextual meaning, leading to errors in synset selection. For example, in Case 1, Question 1 (Capitalism defect, ʿayb raʾsmāliyya – عيب رأسمالية), the word ʿayb (عيب, defect) is classified as a noun and is associated with 16 different synsets, including imperfection, abnormality, fault, and defect. In contrast, in Question 2 (salbī raʾsmāliyya – سلبي رأسمالية, negative capitalism), salbī (سلبي, negative) is classified as an adjective and mapped to a different synset containing passive, minus, negative, and unfavorable. The semantic similarity between the two questions is 0.42, which falls below the task threshold of 0.5, further illustrating the challenge of POS-based synset selection, as it may misclassify semantically similar words by overlooking context. Nevertheless, since the objective is to compare the Arabic WordNet and LughaNet rather than enhance semantic similarity, this approach remains suitable for evaluating both knowledge bases.

Table 8 *Samples of misclassification predictions*

Cases	Question 1	Question 2	True Label	Predicted Probability
Case 1	عيب راسماليه Capitalism defect	سلبي راسماليه negative capital	1	0.42
Case2	طريقه عمل ماسك بشره method to make a skin mask	خطوه لازم عمل قناع وجه Step to make a face mask	1	0.32
Case 3	قام جهاز قضاءي مملكه اردني The Jordanian judiciary has established	ساهم جهاز قضاءي أراد Jordanian judiciary contributed	1	0.45
Case 4	ولد بكر زاي Born Bakr Zay	اي مدينه ولد راز Which city was Raz born in	1	0.38
Case 5	خديجه بنت خويلد Khadija bint Khuwaylid	زوجه نبي Prophet's wife	1	0.07
Case 6	سبب نقص فيتامين ب 12 Cause of vitamin B12 deficiency	سبب نقص فيتامين ب 2 Cause of vitamin B2 deficiency	0	0.81
Case 7	طريقه تحضير ورقه عنب How to prepare grape leaves	طريق تحضير ورقه عنب دجاج How to prepare chicken vine leaves	0	0.86
Case 8	قصد مدير مبيع Sales manager intended	قصد اداره مبيع Intended to manage sales	0	0.70
Case 9	ضرر تناول بقدونس كميه كبير جدا Harm of eating a very large amount of parsley	ضرر تناول نعناع كميه كبير Harmful of eating large amounts of mint	0	0.69
Case 10	طريق تحضير كيك بسكويت زبد How to make butter biscuit cake	طريق تحضير بسكويت زبد How to make butter biscuits	0	1.0

In Case 2 (ṭarīqat ʿamal māsk bishra – طريقة عمل ماسك بشرة, the method to make a skin mask vs. khuṭwa lāzim ʿamal qināʿ wajh – خطوة لازم عمل قناع وجه, the step must make a face mask), the misclassification arises due to minimal lexical overlap, as the two questions share only the word ʿamal (عمل, "make"), leading to a low predicted probability of 0.32 despite the true label being 1. Similarly, in Case 3 (qāma jihāz qaḍāʾī mamlaka ʾurduniyya – قام جهاز قضائي مملكة أردنية, The Jordanian judiciary has established vs. sāhama jihāz qaḍāʾī arād – ساهم جهاز قضائي أراد, The Jordanian judiciary contributed), the issue stems from preprocessing errors, where stemming incorrectly transformed ʾurdunī (أردني, Jordanian) into arād (أراد), an invalid word in Arabic. This demonstrates a common problem in Arabic NLP, where stemming can generate out-of-vocabulary (OOV) words that do not exist in the language, leading to misclassification.

In Case 4 (wulida bakr zāy – ولد بكر زاي, Born Bakr Zay vs. ʾayy madīna wulida rāz – أي مدينة ولد راز, Which city was Raz born in?), a similar stemming issue appears, as the correct word zāy (زاي) in Question 1 was mistakenly processed as rāz (راز) in Question 2, causing semantic distortion. This challenge relates to issues in Arabic NLP, which is considered a low-resource language with limited tools, where morphological complexities are not always handled accurately. In Case 5 (khadīja bint khuwaylid – خديجة بنت خويلد, Khadija bint Khuwaylid vs. zawjat nabī – زوجة نبي, Prophet's wife), the misclassification results from Named Entity Recognition (NER) limitations. While Question 1 explicitly mentions a named entity (khadīja bint khuwaylid – خديجة بنت خويلد), LughaNet's reliance on the

English WordNet leads to the categorization of words into linguistic classes (nouns, verbs, adjectives, adverbs) without adequately addressing named entities.

In Case 6 (sabab nuqṣ fītamīn B12 – سبب نقص فيتامين ب 12, Cause of vitamin B12 deficiency vs. sabab nuqṣ fītamīn B2 – سبب نقص فيتامين ب 2, Cause of vitamin B2 deficiency), the misclassification occurs because the method does not account for multi-word expressions such as vitamin B12 and vitamin B2. As a result, the system assigns a high similarity score of 0.81 due to significant lexical overlap in words like sabab (سبب, cause), fītamīn (فيتامين, vitamin), and nuqṣ (نقص, deficiency), while being unable to distinguish between B12 and B2 because these terms are unavailable in LughaNet as a single sense. This limitation arises from treating words individually rather than recognizing multi-word terms as a single semantic unit, leading to incorrect classification.

In Case 7, the misclassification occurs due to high lexical overlap between the two questions, with dijāj (دجاج, chicken) being the only distinguishing word in Question 2, leading to a high similarity score of 0.86. Similarly, in Case 8, mudīr (مدير, manager) in Question 1 and ʾidāra (إدارة, management) in Question 2 are semantically related, yet the system assigns a similarity score of 0.70 without fully capturing their contextual difference. In Case 9, five words are identical between the two questions, with the only difference being baqdūnis (بقدونس, parsley) in Question 1 and naʿnāʿ (نعناع, mint) in Question 2, resulting in a similarity score of 0.69. Finally, in Case 10, nearly complete lexical overlap leads to a similarity score of 1.0, with the only difference being the additional word kīk (كيك, cake) in Question 1.

6. Conclusion

In conclusion, this study developed LughaNet, an automated Arabic WordNet, by aligning synsets from Princeton WordNet (PWN) with Arabic terms using an English-Arabic bilingual dictionary and pre-trained models for multilingual machine translation. Translations were refined by selecting the most frequent words and eliminating incorrect ones using BERT and cosine similarity. Arabic terms were extracted from multiple resources, including Wikipedia and existing Arabic WordNets, while techniques such as Skip-gram, based on pre-trained AraVec 2.0 embeddings from Wikipedia, were employed to extract synonyms. Synonym selection accuracy was further improved using a pre-trained BERT model and cosine similarity measures, resulting in LughaNet comprising 85,991 synsets. Additionally, the translation of synset glosses and examples into Arabic enriched LughaNet's content. The evaluation demonstrated LughaNet's improved coverage and usability, achieving 64.23% dictionary term coverage. For semantic question similarity tasks, LughaNet attained an F1 score of 66.61%, highlighting its utility in capturing semantic relationships.

This study has certain limitations. The sense representation between LughaNet and the dataset was selected based on part of speech. While LughaNet includes many multi-word expressions (34.74%), this study relied solely on single-word representations. Additionally, the processing tools exhibited limitations, sometimes generating incorrect words, highlighting the need for improvements in tools such as stemming.

Future improvements could involve translating missing synsets from Princeton WordNet, updating LughaNet to provide an API, and using crowdsourcing to expand lexical coverage and enhance synsets. Moreover, while this study used part-of-speech tagging to select synsets, future research could employ advanced techniques such as word embeddings for sense representation selection, combined with parsing and chunking tools to better identify multi-word expressions, ultimately enhancing LughaNet's performance.

Acknowledgement

The authors sincerely thank Ferdowsi University of Mashhad for their invaluable support and resources throughout this research. All content was carefully reviewed and revised by the authors to ensure accuracy and alignment with the research objectives.

Conflict of Interest

The authors declare that there is no conflict of interest regarding the paper's publication.

Code and Knowledge Base Availability

The code for this study is publicly available on GitHub at https://github.com/ammardhnoor/LughaNet. Additionally, the LughaNet knowledge base can be accessed and used free of charge. For requests to use it, please do not hesitate to contact the first author via email at ammardhnoor1@gmail.com.

Author Contribution

All authors made an equal contribution to the development and planning of the study.

References

[1] Fellbaum, C. *WordNet: An electronic lexical database*. (1998). MIT Press. https://doi.org/10.7551/mitpress/7287.001.0001

[2] Hussain, W. A. G. A. (2021). Applying similarity measures to improve query expansion. *Iraqi Journal of Science,* 2053-2063. https://doi.org/10.24996/IJS.2021.62.6.31

[3] Hussain, M. J., Bai, H., Wasti, S. H., Huang, G., & Jiang, Y. (2023). Evaluating semantic similarity and relatedness between concepts by combining taxonomic and non-taxonomic semantic features of WordNet and Wikipedia. *Information Sciences*, 625, 673-699. https://doi.org/10.1016/j.ins.2023.01.007

[4] AlMousa, M., Benlamri, R., & Khoury, R. (2021). Exploiting non-taxonomic relations for measuring semantic similarity and relatedness in WordNet. *Knowledge-Based Systems,* 212, 106565. https://doi.org/10.1016/j.knosys.2020.106565

[5] AlMousa, M., Benlamri, R., & Khoury, R. (2022). A novel word sense disambiguation approach using WordNet knowledge graph. *Computer Speech & Language*, 74, 101337. https://doi.org/10.48550/arXiv.2101.02875

[6] Geller, E. Gajek, M. Reibach, A. & Łapa, Z. (2021). Applicability of wordnet architecture in lexical borrowing studies. *International Journal of Lexicography,* 34(1), 92-111. https://doi.org/10.1093/ijl/ecaa013

[7] Dziob, A. & Naskręt, T. Wordnet–a Basic Resource for Natural Language Processing: The Case of plWordNet. (2020). *Advances in Computational Collective Intelligence,* ICCCI 2020, Communications in Computer and Information Science, 1287, 689-700. https://doi.org/10.1007/978-3-030-63119-2_56

[8] Vossen, P. (2004). Eurowordnet: a multilingual database of autonomous and language-specific wordnets connected via an inter-lingualindex. *International Journal of Lexicography*, 17(2), 161–173. https://doi.org/10.1093/ijl/17.2.161

[9] Tufis,D., Cristea, D., & Stamou, S. (2004). BalkaNet: Aims, methods, results and perspectives. a general overview. *Romanian Journal of Information science and technology*, 7(1-2), 9-43.

[10] Berangi, P., Mousavi, Z., Faili, H., & Shakery, A. (2021). WordNet construction for under-resourced languages using personalized PageRank. *Digital Scholarship in the Humanities,* 36(3), 565-580. https://doi.org/10.1093/llc/fqaa036

[11] Fellbaum, C. et al. (2006). Introducing the Arabic wordnet project. *Third Global Wordnet Conference*, Jeju Island, Korea.

[12] Ahmed, I. A., AL-Aswadi, F. N., & Noaman, K. M. (2022). Arabic Knowledge Graph Construction: A close look in the present and into the future. *Journal of King Saud University-Computer and Information Sciences*, 34(9), 6505-6523. https://doi.org/10.1016/j.jksuci.2022.04.007

[13] Batita, M. A., & Zrigui, M. (2019). The extended Arabic WordNet: A case study and an evaluation using a word sense disambiguation system. *Proceedings of the 10th Global Wordnet Conference*, 46-53.

[14] Lebboss, G., Bernard, G., Aliane, N., & Hajjar, M. (2017). Towards the Enrichment of Arabic WordNet with Big Corpora. *IJCCI*, 101-109. https://doi.org/10.5220/0006505701010109

[15] Freihat, A. Khalilia, A., Bella, H. G., & Giunchiglia, F. (2024). Advancing the Arabic WordNet: Elevating Content Quality. arXiv preprint arXiv. 20215. https://doi.org/10.48550/arXiv.2403.20215

[16] Moskvoretskii, V., Neminova, E., Lobanova, A., Panchenko, A., & Nikishina, I. (2024). TaxoLLaMA: WordNet-based Model for Solving Multiple Lexical Sematic Tasks. arXiv preprint arXiv, 09207. https://doi.org/10.48550/arXiv.2403.09207

[17] Zhu, Z., & Mao, K. (2023). Knowledge-based BERT word embedding fine-tuning for emotion recognition. *Neurocomputing,* 552, 126488. https://doi.org/10.1016/j.neucom.2023.126488

[18] Zervakis, G., Vincent, E., Couceiro, M., & Schoenauer, M. (2021). On refining bert contextualized embeddings using semantic lexicons. *ECML PKDD 2021-Machine Learning with Symbolic Methods and Knowledge Graphs co-located with European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, Spain.

[19] ElKateb, S., et al., (2006). Building a WordNet for Arabic. *Proceedings of the Fifth International Conference on Language Resources and Evaluation*, Italy, 29-34.

[20] Cavalli-Sforza, V., Saddiki, H., Bouzoubaa, K., Abouenour, L., Maamouri, M., & Goshey, E. (2013). Bootstrapping a wordnet for an arabic dialect from other wordnets and dictionary resources. *International Conference on Computer Systems and Applications (AICCSA)*, Morocco, 1-8. https://doi.org/10.1109/AICCSA.2013.6616508

[21] Batita M. A., & Zrigui, M. (2017). The enrichment of arabic wordnet antonym relations. Computational *Linguistics and Intelligent Text Processing: 18th International Conference*, Hungary, 342-353. https://doi.org/10.1007/978-3-319-77113-7_27

[22] Maziarz M., & Rudnicka, E. (2020). Expanding wordnet with gloss and polysemy links for evocation strength recognition. Cognitive Studies, 20. https://doi.org/10.11649/cs.2325

[23] Kanika, C. S., Chakraborty, P., Aggarwal, A., Madan, M., Gupta, G. (2022). Enriching WordNet with Subject Specific Out of Vocabulary Terms Using Existing Ontology. Data Engineering for Smart Systems. *Lecture Notes in Networks and Systems*, Singapore, 238. https://doi.org/10.1007/978-981-16-2641-8_19

[24] Li, S., Li, B., Yao, H., Zhou, S., Zhu, J., & Zeng, Z. (2021). Completing wordnets with sememe knowledge. *Electronics*, 11(1), 79. https://doi.org/10.3390/electronics11010079

[25] Mousavi, Z., & Faili, H. (2021). Developing the persian wordnet of verbs using supervised learning. *Transactions on Asian and Low-Resource Language Information Processing*, 20(4), 1-18. https://doi.org/10.1145/3450969

[26] Leenoi, D., Alongkornchai, A., Takhom, A., Boonkwan, P., & Sunnithi, T. (2022). A Construction of Thai WordNet through Translation Equivalence. *17th International Joint Symposium on Artificial Intelligence and Natural Language Processing*, IEEE, 1-4. ttps://doi.org/doi: 10.1109/iSAI-NLP56921.2022.9960263

[27] Samson, B. P., et al., (2023). Towards the Creation of the Filipino Wordnet: A Two-Way Approach. *International Conference on Asian Language Processing (IALP)*, IEEE, 320-325. https://doi.org/10.1109/ialp61005.2023.10336981

[28] Rodríguez, H., Farwell, D., Ferreres, J., Bertran, M., Alkhalifa, M., & M. Martí, A. (2008), Arabic WordNet: Semi-automatic Extensions using Bayesian Inference. *Proceedings of the Sixth International Conference on Language Resources and Evaluation*, Morocco.

[29] Alkhalifa, M., & Rodríguez, H. (2009). Automatically extending NE coverage of Arabic WordNet using Wikipedia. Proc. Of the 3rd International Conference on Arabic Language Processing CITALA2009, Morocco, 23-30.

[30] Abouenour, L., Bouzoubaa, K., & Rosso, P. (2013). On the evaluation and improvement of Arabic WordNet coverage and usability. *Language resources and evaluation*, 47, 891-917. https://doi.org/10.1007/s10579-013-9237-0

[31] Hadj Ameur, M. S., A. Khadir, C., and Guessoum, A. (2017). An automatic approach for wordnet enrichment applied to arabic wordnet. *International Conference on Arabic Language Processing*, Springer, 3-18. https://doi.org/10.1007/978-3-319-73500-9_1

[32] Lachichi, C., Bendiaf, C., Berkani, L., & Guessoum, A. (2018). An Arabic WordNet enrichment approach using machine translation and external linguistic resources. *2nd International Conference on Natural Language and Speech Processing (ICNLSP)*, IEEE, 1-6. https://doi.org/10.1109/ICNLSP.2018.8374385

[33] Badaro, G., Hajj, H., & Habash, N. (2020). A link prediction approach for accurately mapping a large-scale Arabic lexical resource to English WordNet. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*, 19(6), 1-38. https://doi.org/10.1145/3404854

[34] Lam, K. N., Tarouti, F. A., & Kalita, J. (2022). Automatically constructing Wordnet synsets. *arXiv preprint arXiv*, 03870. https://doi.org/10.48550/arXiv.2208.03870

[35] Souci, M. D. E., Cherifi, Y., Berkani, L., Ameur, M. S. H., & Guessoum, A. (2023). Enrichment of Arabic WordNet Using Machine Translation and Transformers. *Proceedings of the 6th International Conference on Natural Language and Speech Processing (ICNLSP 2023)*, 333-340.

[36] A. Alsudais. (2022). Extending ImageNet to Arabic using Arabic WordNet. *Multimedia Tools and Applications*, 81(6), 8835-8852. https://doi.org/10.1007/s11042-022-11981-6

[37] Omer, Z., Al Daoud, E., Samara, G., & Al-Lahham, Y. (2024). An Algorithm for Arabic Semantic Matching Using New WordNet Tree. *Frontiers of Human Centricity in the Artificial Intelligence-Driven Society 5.0: Springer*, 1331-1341. https://doi.org/10.1007/978-3-031-73545-5_124

[38] Mehdioui, O., Rhazi, A., & Refouh, K. (2024). Morpho-Lexical Based Approach for Arabic WorldNet Extension. *International Conference on Arabic Language Processing: Springer*, 90-96. https://doi.org/10.1007/978-3-031-79164-2_8

[39] Taye, M. M., Abulail, R., & Al-Oudat, M. (2023). An Ontology Learning Framework for unstructured Arabic Text. *7th International Symposium on Innovative Approaches in Smart Technologies (ISAS): IEEE*, 1-12. https://doi.org/10.1109/ISAS60782.2023.10391548

[40] Kahlawi, A. (2023). An ontology-driven DBpedia quality enhancement to support Entity Annotation for Arabic Text, *International Journal of Advanced Computer Science and Applications*, 14(3). http://dx.doi.org/10.14569/IJACSA.2023.0140301

[41] Zouaoui, S., & Rezeg, K. (2021). Islamic inheritance calculation system based on Arabic ontology (AraFamOnto). *Journal of King Saud University-Computer and Information Sciences*, 33(1), 68-76. https://doi.org/10.1016/j.jksuci.2018.11.015

[42] Altammami, S., Atwell, E., & Alsalka, A. (2021). Towards a joint ontology of quran and hadith. *International Journal on Islamic Applications in Computer Science And Technology*, 9(2), 01-12.

[43] Kamran, A. B., Abro, B., & Basharat, A. (2023). SemanticHadith: An ontology-driven knowledge graph for the hadith corpus. *Journal of Web Semantics*, 78, 100797. https://doi.org/10.1016/j.websem.2023.100797

[44] Khabour, S. M., Al-Radaideh, Q. A., & Mustafa, D. (2022). A new ontology-based method for Arabic sentiment analysis. *Big Data and Cognitive Computing*, 6(2), 48. https://doi.org/10.3390/bdcc6020048

[45] Boulaalam, A., & El Hannach, N. (2023). Arabic Ontology: Linguistic Engineering Foundations. *Journal of e-learning Research*, 2(1), 29-49. https://doi.org/10.33422/jelr.v2i1.378

[46] Alsemaree, O., Alam, A. S., Gill, S. S., & Uhlig, S. (2024). An analysis of customer perception using lexicon-based sentiment analysis of Arabic Texts framework. *Heliyon*, 10(11). https://doi.org/10.1016/j.heliyon.2024.e30320

[47] Aldawsari, H. (2025). Evaluating the Performance of Large Language Models on Arabic Lexical Ambiguities: A Comparative Study with Traditional Machine Translation Systems. *World Journal of English Language*, 15, 354. https://doi.org/doi: 10.5430/wjel.v15n3p354

[48] Zhang, J. et al. (2024). Neural machine translation for low-resource languages from a chinese-centric perspective: A survey. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 23(6), 1-60. https://doi.org/10.1145/36652

[49] Tang, Y., et al., (2020). Multilingual translation with extensible multilingual pretraining and finetuning. *arXiv preprint arXiv*, 00401. https://doi.org/10.48550/arXiv.2008.00401

[50] Tars, M., Purason, T., & Tättar, A. (2022). Teaching unseen low-resource languages to large translation models. *Proceedings of the Seventh Conference on Machine Translation (WMT)*, 375-380.

[51] Gómez, S., Domingo, M., & Casacuberta, F. (2024). Interactive Machine Translation with Large Language Models in Low Resources Languages. *Proc. IberSPEECH 2024*, 66-70. https://doi.org/10.21437/IberSPEECH.2024-14

[52] Bentivogli, L., Bisazza, A., Cettolo, M., & Federico, M. (2018). Neural versus phrase-based mt quality: An in-depth analysis on english–german and english–french. *Computer speech & language,* 49, 52-70. https://doi.org/10.1016/j.csl.2017.11.004

[53] Hu, S., et al., (2023). Neural Machine Translation by Fusing Key Information of Text. *CMC Comput. Mater. Contin*, 74, 2803-2815. https://doi.org/10.32604/cmc.2023.032732

[54] Lee, E.-S. A., et al., (2022). Pre-trained multilingual sequence-to-sequence models: A hope for low-resource language translation? *arXiv preprint arXiv*, 08850. https://doi.org/10.48550/arXiv.2203.08850

[55] Baniata, L. H., Ampomah, I. K., & Park, S. (2021). A transformer-based neural machine translation model for Arabic dialects that utilizes subword units. *Sensors*, 21(19), 6509. https://doi.org/10.3390/s21196509

[56] Junczys-Dowmunt, M. et al., (2018). Marian: Fast neural machine translation in C++. *arXiv preprint arXiv*, 00344. https://doi.org/10.48550/arXiv.1804.00344

[57] Fan, A., et al. (2021). Beyond english-centric multilingual machine translation. *Journal of Machine Learning Research,* 22(107), 1-48. https://doi.org/10.48550/arXiv.2010.11125

[58] Costa-jussà, M. R., et al., (2022). No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv*, 04672. https://doi.org/10.48550/arXiv.2207.04672

[59] Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv*, 3781. https://doi.org/10.48550/arXiv.1301.3781

[60] Soliman, A. B., Eissa, K., & El-Beltagy, S. R. (2017). Aravec: A set of arabic word embedding models for use in arabic nlp. *Procedia Computer Science*, 117, 256-265. https://doi.org/10.1016/j.procs.2017.10.117

[61] Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv,* 04805. https://doi.org/10.48550/arXiv.1810.04805

[62] Řehůřek, R., & Sojka, P. (2011). Gensim—statistical semantics in python. *Retrieved from genism. org, 2025*.

[63] Wolf, T. (2020). Transformers: State-of-the-Art Natural Language Processing. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. https://doi.org/10.18653/v1/2020.emnlp-demos.6

[64] Pedregosa, F., et al. (2011). Scikit-learn: Machine learning in Python. *The Journal of machine Learning research*, 12, 2825-2830.

[65] Paszke, A. et al. (2019). Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32. *https://doi.org/10.48550/arXiv.1912.01703*

[66] Abu-Haidar, J. (1984). Hans Wehr: A dictionary of modern written Arabic (Arabic-English). (considerably enlarged and amended by the author.) Edited by J. Milton-Cowan. xvii, 1301 pp. Wiesbaden: Otto Harrassowitz, 1979. DM 198. *Bulletin of the School of Oriental and African Studies*, 46(2), 351-353.

[67] Salton, G., & Buckley, C. (1087). *Term weighting approaches in automatic text retrieval*. Cornell University.

[68] Wu, Z., & Palmer, M. (1994). Verb semantics and lexical selection. 32nd Annual Meeting of the Association for Computational Linguistics, 133-138. https://doi.org/10.48550/arXiv.cmp-lg/9406033

[69] Pedersen, T., Patwardhan, S., & Michelizzi, J. (2004). WordNet: Similarity-Measuring the Relatedness of Concepts. *Proceedings of the AAAI Conference on Artificial Intelligence*, 4, 25-29.

[70] Fellbaum, C., & Miller, G. (1998). Combining local context and WordNet similarity for word sense identification. *MIT Press*, 265-283.

[71] Chandrasekaran, D., & Mago, V. (2021). Evolution of semantic similarity—a survey. *ACM Computing Surveys (CSUR)*, 54(2), 1-37. https://doi.org/10.1145/3440755

[72] Wang, J., & Dong, Y. (2020). Measurement of text similarity: a survey. *Information*, 11(9), 421. https://doi.org/10.3390/info11090421

[73] Wafaa, A.-H. (2024). Extracting Key-phrase Embedding using Deep Average Network and Maximal Marginal Relevance to Enhance Information Retrieval. *Journal of University of Babylon*, 32(2). https://doi.org/10.29196/jubpas.v32i2.5268

[74] Li, M., Shen, X., Sun, Y., Zhang, W., Nan, J., & Gao, D. (2023). Using Semantic Text Similarity calculation for question matching in a rheumatoid arthritis question-answering system. *Quantitative Imaging in Medicine and Surgery*, 13(4), 2183. https://doi.org/10.21037/qims-22-749

[75] Fuad, A., & Al-Yahya, M., (2022). Recent developments in Arabic conversational AI: a literature review. *IEEE Access,* 10, 23842-23859. https://doi.org/10.1109/ACCESS.2022.3155521

[76] Hamza, A., Ouatik, S. E. A., Zidani, K. A., & En-Nahnahi, N. (2022). Arabic duplicate questions detection based on contextual representation, class label matching, and structured self attention. *Journal of King Saud University-Computer and Information Sciences*, 34(6), 3758-3765. https://doi.org/10.1016/j.jksuci.2020.11.032

[77] Al-Asa'd, M., Al-Khdour, N., Younes, M. B., Khwaileh, E., Hammad, M., & Mohammad, A.-S. (2019). Question to Question Similarity Analysis using Morphological, Syntactic, Semantic, and Lexical Features. *IEEE/ACS 16th International Conference on Computer Systems and Applications (AICCSA)*, IEEE, 1-6. https://doi.org/10.1109/AICCSA47632.2019.9035248

[78] Othman, N., Faiz, R., & Smaïli, K. (2022). Learning English and Arabic question similarity with Siamese neural networks in community question answering services. *Data & Knowledge Engineering*, 138, 101962. https://doi.org/10.1016/j.datak.2021.101962

[79] Han, M., Zhang, X., Yuan, X., Jiang, J., Yun, W., & Gao, C. (2021). A survey on the techniques, applications, and performance of short text semantic similarity. *Concurrency and Computation: Practice and Experience*, 33(5), e5971. https://doi.org/10.1002/cpe.5971

[80] Al Sulaiman, M., Moussa, A. M., Abdou, S., Elgibreen, H., Faisal, M., & Rashwan, M. (2022). Semantic textual similarity for modern standard and dialectal Arabic using transfer learning. *PloS one*, 17(8), e0272991. https://doi.org/10.1371/journal.pone.0272991

[81] Alshammari, H., & Elleithy, K. (2024). Toward Robust Arabic AI-Generated Text Detection: Tackling Diacritics Challenges. *Information*, 15(7), 419. https://doi.org/10.3390/info15070419

[82] Azmi, A. M., Alnefaie, R. M., & Aboalsamh, H. A. (2021). Light diacritic restoration to disambiguate homographs in modern Arabic texts. *Transactions on Asian and Low-Resource Language Information Processing*, 21(3), 1-14. https://doi.org/10.1145/3486675

[83] Hasan, A. M., Noor, N. M., Rassem, T. H., Hasan, A. M., & Hammood, W. A. (2020). A Combined Weighting for the Feature-Based Method on Topological Parameters in Semantic Taxonomy Using Social Media. *IOP Conference Series: Materials Science and Engineering*, 769(1), 012002. https://doi.org/10.1088/1757-899X/769/1/012002

[84] Hasan, A. M., Rassem, T. H., Noor, N. M., & Hasan, A. M. (2021). A Review of Recent trends: text mining of taxonomy using WordNet 3.1 for the solution and problems of ambiguity in social media. *Intelligent Computing and Innovation on Data Science: Proceedings of ICTIDS*, 137-152. https://doi.org/10.1007/978-981-15-3284-9_15

[85] Selva Birunda, S., & Kanniga Devi, R. (2020). A review on word embedding techniques for text classification. *Innovative Data Communication Technologies and Application: Proceedings of ICIDCA*, 267-281. https://doi.org/10.1007/978-981-15-9651-3_23

[86] Hammad, M., Al-Smadi, M., Baker, Q. B., & Sa'ad, A. (2021). Using deep learning models for learning semantic text similarity of Arabic questions. *International Journal of Electrical and Computer Engineering*, 11(4), 3519. http://doi.org/10.11591/ijece.v11i4.pp3519-3528

[87] Balla, H. A., Llorens Salvador, M., & Delany, S. J. (2022). Arabic Medical Community Question Answering Using ON-LSTM and CNN. *14th international conference on machine learning and computing (ICMLC)*, 298-307. https://doi.org/10.1145/3529836.3529913

[88] Daoud, M. (2022). Topical and non-topical approaches to measure similarity between Arabic questions. *Big Data and Cognitive Computing*, 6(3), 87. https://doi.org/10.3390/bdcc6030087

[89] Keltoum, B., Laboudi, Z., Nouaouria, N., & Zegour, D. E. (2021). An ontology driven question answering system for fatawa retrieval. *Indonesian Journal of Electrical Engineering and Computer Science*, 23(2), 980-992. http://doi.org/10.11591/ijeecs.v23.i2.pp980-992

[90] Boudaa, S., Haddadi, A. E., & Boudaa, T. (2024). *Semantic Similarity Between Arabic Questions Using Support Vector Machines and Hungarian Method*. Modern Artificial Intelligence and Data Science: Tools, Techniques and Systems: Springer, 213-222. https://doi.org/10.1007/978-3-031-65038-3_17

[91] Li, F., Liao, L., Zhang, L., Zhu, X., Zhang, B., & Wang, Z. (2020). An efficient approach for measuring semantic similarity combining WordNet and Wikipedia. *IEEE Access,* 8, 184318-184338. https://doi.org/10.1109/ACCESS.2020.3025611

[92] Zhao, F., Zhu, Z., & Han, P. (2021). A novel model for semantic similarity measurement based on wordnet and word embedding. *Journal of Intelligent & Fuzzy Systems*, 40(5), 9831-9842. https://doi.org/10.3233/JIFS-202337

[93] Hameed, N. H., Alimi, A. M., & Sadiq, A. T. (2022). Short Text Semantic Similarity Measurement Approach Based on Semantic Network. *Baghdad Science Journal*, 19(6), 1581-1581. https://orcid.org/0000-0002-4552-3927

[94] Seelawi, H., Mustafa, A., Al-Bataineh, H., Farhan, W., & Al-Natsheh, H. T. (2019). NSURL-2019 task 8: Semantic question similarity in arabic. *Proceedings of the First International Workshop on NLP Solutions for Under Resourced Languages (NSURL 2019) co-located with ICNLSP 2019-Short Papers*, 1-8. https://doi.org/10.48550/arXiv.1909.09691

[95] Nandwani, P., & Verma, R. (2021). A review on sentiment analysis and emotion detection from text. *Social network analysis and mining*, 11(1), 81. https://doi.org/10.1007/s13278-021-00776-6

[96] Powers, D. M. (2020). Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. *Journal of Machine Learning Technologies*, 2(1), 37-63. https://doi.org/10.48550/arXiv.2010.16061

[97] Sasaki, Y. (2007). The truth of the F-measure. *Teach tutor mater*, 1(5), 1-5.