Evolution of Semantic Similarity—A Survey

DHIVYA CHANDRASEKARAN and VIJAY MAGO, Lakehead University

Estimating the semantic similarity between text data is one of the challenging and open research problems in the field of Natural Language Processing (NLP). The versatility of natural language makes it difficult to define rule-based methods for determining semantic similarity measures. To address this issue, various semantic similarity methods have been proposed over the years. This survey article traces the evolution of such methods beginning from traditional NLP techniques such as kernel-based methods to the most recent research work on transformer-based models, categorizing them based on their underlying principles as knowledge-based, corpus-based, deep neural network–based methods, and hybrid methods. Discussing the strengths and weaknesses of each method, this survey provides a comprehensive view of existing systems in place for new researchers to experiment and develop innovative ideas to address the issue of semantic similarity.

CCS Concepts: • **General and reference** → **Surveys and overviews**; • **Information systems** → **Ontologies**; • **Theory of computation** → **Unsupervised learning and clustering**; • **Computing methodologies** → **Lexical semantics**;

Additional Key Words and Phrases: Semantic similarity, linguistics, supervised and unsupervised methods, knowledge-based methods, word embeddings, corpus-based methods

ACM Reference format:
Dhivya Chandrasekaran and Vijay Mago. 2021. Evolution of Semantic Similarity—A Survey. *ACM Comput. Surv.* 54, 2, Article 41 (February 2021), 37 pages.
https://doi.org/10.1145/3440755

1 INTRODUCTION

With the exponential increase in text data generated over time, Natural Language Processing (NLP) has gained significant attention from Artificial Intelligence (AI) experts. Measuring the semantic similarity between various text components such as words, sentences, or documents plays a significant role in a wide range of NLP tasks such as information retrieval [48], text summarization [80], text classification [49], essay evaluation [42], machine translation [134], and question answering [19, 66], among others. In the early days, two text snippets were considered similar if they contain the same words/characters. The techniques such as Bag of Words (BoW), Term Frequency-Inverse Document Frequency (TF-IDF) were used to represent text, as real value vectors to aid calculation of semantic similarity. However, these techniques did not attribute to the fact that words have different meanings and different words can be used to represent a similar concept. For example, consider two sentences: *"John and David studied Maths and Science."* and *"John studied Maths and*

Authors' addresses: D. Chandrasekaran and V. Mago, Lakehead University, 955 Oliver Road, Thunderbay, Ontario, P7B 5E1; email: {dchandra, vmago}@lakeheadu.ca.

0360-0300/2021/02-ART41 $15.00
https://doi.org/10.1145/3440755

David studied Science." Though these two sentences have exactly the same words, they do not convey the same meaning. Similarly, the sentences *"Mary is allergic to dairy products."* and *"Mary is lactose intolerant."* convey the same meaning; however, they do not have the same set of words. These methods captured the lexical feature of the text and were simple to implement, however, they ignored the semantic and syntactic properties of text. To address these drawbacks of the lexical measures various semantic similarity techniques were proposed over the past three decades.

Semantic Textual Similarity (STS) is defined as the measure of semantic equivalence between two blocks of text. Semantic similarity methods usually give a ranking or percentage of similarity between texts, rather than a binary decision as similar or not similar. Semantic similarity is often used synonymously with semantic relatedness. However, semantic relatedness not only accounts for the semantic similarity between texts but also considers a broader perspective analyzing the shared semantic properties of two words. For example, the words *"coffee"* and *"mug"* may be related to one another closely, but they are not considered semantically similar whereas the words *"coffee"* and *"tea"* are semantically similar. Thus, semantic similarity may be considered as one of the aspects of semantic relatedness. The semantic relationship including similarity is measured in terms of semantic distance, which is inversely proportional to the relationship [37].

1.1 Motivation behind the Survey

Most of the survey articles published recently related to semantic similarity, provide in-depth knowledge of one particular semantic similarity technique or a single application of semantic similarity. Lastra-Díaz et al. survey various knowledge-based methods [55] and IC-based methods [53], Camacho-Colladas et al. [20] discuss various vector representation methods of words, Taieb et al. [37], however, describe various semantic relatedness methods and Berna Altinel et al. [8] summarize various semantic similarity methods used for text classification. The motivation behind this survey is to provide a comprehensive account of the various semantic similarity techniques including the most recent advancements using deep neural network–based methods.

This survey traces the evolution of Semantic Similarity Techniques over the past decades, distinguishing them based on the underlying methods used in them. Figure 1 shows the structure of the survey. A detailed account of the widely used datasets available for semantic similarity is provided in Section 2. Sections 3 to 6 provide a detailed description of semantic similarity methods broadly classified as (1) Knowledge-based methods, (2) Corpus-based methods, (3) Deep neural network-based methods, and (4) Hybrid methods. Section 7 analyzes the various aspects and inference of the survey conducted. This survey provides a deep and wide knowledge of existing techniques for new researchers who venture to explore one of the most challenging NLP tasks, Semantic Textual Similarity.

2 DATASETS

In this section, we discuss some of the popular datasets used to evaluate the performance of semantic similarity algorithms. The datasets may include word pairs or sentence pairs with associated standard similarity values. The performance of various semantic similarity algorithms is measured by the correlation of the achieved results with that of the standard measures available in these datasets. Table 1 lists some of the popular datasets used to evaluate the performance of semantic similarity algorithms. The below subsection describes the attributes of the dataset and the methodology used to construct them.

2.1 Semantic Similarity Datasets

The following is a list of widely used semantic similarity datasets arranged chronologically.

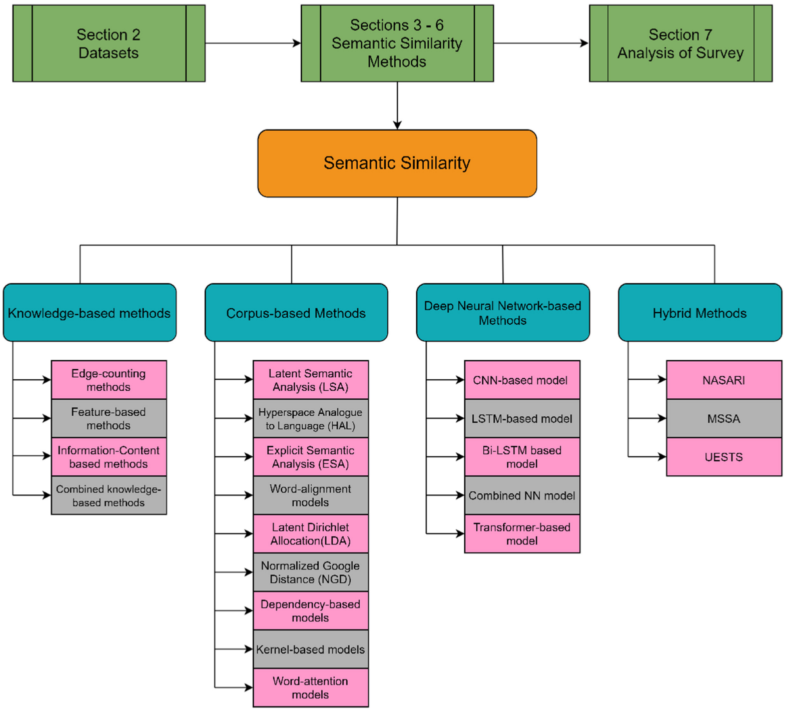

Fig. 1. Survey architecture.

Table 1. Popular Benchmark Datasets for Semantic Similarity

Dataset Name	Word/Sentence pairs	Similarity score range	Year	Reference
R&G	65	0–4	1965	[107]
M&C	30	0–4	1991	[78]
WS353	353	0–10	2002	[30]
LiSent	65	0–4	2007	[63]
SRS	30	0–4	2007	[94]
WS353-Sim	203	0–10	2009	[1]
STS2012	5,250	0–5	2012	[5]
STS2013	2,250	0–5	2013	[6]
WP300	300	0–1	2013	[61]
STS2014	3,750	0–5	2014	[3]
SL7576	7,576	1–5	2014	[116]
SimLex-999	999	0–10	2014	[40]
SICK	10,000	1–5	2014	[69]
STS2015	3,000	0–5	2015	[2]
SimVerb	3,500	0–10	2016	[34]
STS2016	1,186	0–5	2016	[4]
WiC	5,428	NA	2019	[97]

- **Rubenstein and Goodenough (R&G)** [107]: This dataset was created as a result of an experiment conducted among 51 undergraduate students (native English speakers) in two different sessions. The subjects were provided with 65 selected English noun pairs and requested to assign a similarity score for each pair over a scale of 0 to 4, where 0 represents that the words are completely dissimilar and 4 represents that they are highly similar. This dataset is the first and most widely used dataset in semantic similarity tasks [133].
- **Miller and Charles (M&C)** [78]: Miller and Charles repeated the experiment performed by Rubenstein and Goodenough in 1991 with a subset of 30 word pairs from the original 65 word pairs. 38 human subjects ranked the word pairs on a scale from 0 to 4, 4 being the "most similar."
- **WS353** [30]: WS353 contains 353 word pairs with an associated score ranging from 0 to 10. 0 represents the least similarity and 10 represents the highest similarity. The experiment was conducted with a group of 16 human subjects. This dataset measures semantic relatedness rather than semantic similarity. Subsequently, the next dataset was proposed.
- **WS353-Sim** [1]: This dataset is a subset of WS353 containing 203 word pairs from the original 353 word pairs that are more suitable for semantic similarity algorithms specifically.
- **LiSent** [63]: 65 sentence pairs were built using the dictionary definition of 65 word pairs used in the R&G dataset. 32 native English speakers volunteered to provide a similarity range from 0 to 4, 4 being the highest. The mean of the scores given by all the volunteers was taken as the final score.
- **SRS** [94]: Pedersen et al. [94] attempted to build a domain-specific semantic similarity dataset for the biomedical domain. Initially 120 pairs were selected by a physician distributed with 30 pairs over four similarity values. These term pairs were then ranked by 13 medical coders on a scale of 1–10. 30 word pairs from the 120 pairs were selected to increase reliability and these word pairs were annotated by 3 physicians and 9 (out of the 13) medical coders to form the final dataset.
- **SimLex-999** [40]: 999 word pairs were selected from the UFS Dataset [89] of which 900 were similar and 99 were related but not similar. 500 native English speakers, recruited via Amazon Mechanical Turk were asked to rank the similarity between the word pairs over a scale of 0 to 6, 6 being the most similar. The dataset contains 666 noun pairs, 222 verb pairs, and 111 adjective pairs.
- **Sentences Involving Compositional Knowledge (SICK) dataset** [69]: The SICK dataset consists of 10,000 sentence pairs, derived from two existing datasets: the ImageFlickr 8 and MSR-Video descriptions dataset. Each sentence pair is associated with a relatedness score and a text entailment relation. The relatedness score ranges from 1 to 5, and the three entailment relations are "NEUTRAL, ENTAILMENT, and CONTRADICTION." The annotation was done using crowd-sourcing techniques.
- **STS datasets** [2–6, 24]: The STS datasets were built by combining sentence pairs from different sources by the organizers of the SemEVAL shared task. The dataset was annotated using Amazon Mechanical Turk and further verified by the organizers themselves. Table 2 shows the various sources from which the STS dataset was built.

3 KNOWLEDGE-BASED SEMANTIC-SIMILARITY METHODS

Knowledge-based semantic similarity methods calculate semantic similarity between two terms based on the information derived from one or more underlying knowledge sources, such as ontologies/lexical databases, thesauri, dictionaries, and so on. The underlying knowledge-base offers these methods a structured representation of terms or concepts connected by semantic relations, further offering an ambiguity free semantic measure, as the actual meaning of the terms is taken

Table 2. STS English Language Training Dataset (2012–2017) [24]

Year	Dataset	Pairs	Source
2012	MSRPar	1,500	newswire
2012	MSRvid	1,500	videos
2012	OnWN	750	glosses
2012	SMTNews	750	WMT eval.
2012	SMTeuroparl	750	WMT eval.
2013	HDL	750	newswire
2013	FNWN	189	glosses
2013	OnWN	561	glosses
2013	SMT	750	MT eval.
2014	HDL	750	newswire headlines
2014	OnWN	750	glosses
2014	Deft-forum	450	forum posts
2014	Deft-news	300	news summary
2014	Images	750	image descriptions
2014	Tweet-news	750	tweet-news pairs
2015	HDL	750	newswire headlines
2015	Images	750	image descriptions
2015	Ans.-student	750	student answers
2015	Ans.-forum	375	Q & A forum answers
2015	Belief	375	committed belief
2016	HDL	249	newswire headlines
2016	Plagiarism	230	short-answers plag.
2016	post-editing	244	MT postedits
2016	Ans.-Ans	254	Q & A forum answers
2016	Quest.-Quest.	209	Q & A forum questions
2017	Trail	23	Mixed STS 2016

into consideration [123]. In this section, we discuss four lexical databases widely employed in knowledge-based semantic similarity methods and further discuss, in brief, different methodologies adopted by some of the knowledge-based semantic similarity methods.

3.1 Lexical Databases

- WordNet [77] is a widely used lexical database for knowledge-based semantic similarity methods that accounts for more than 100,000 English concepts [123]. WordNet can be visualized as a graph, where the nodes represent the meaning of the words (concepts), and the edges define the relationship between the words [133]. WordNet's structure is primarily based on synonyms, where each word has different *synsets* attributed to their different meanings. The similarity between two words depends on the path distance between them [93].
- Wiktionary[1] is an open-source lexical database that encompasses approximately 6.2M words from 4,000 different languages. Each entry has an article page associated with it, and it accounts for a different sense of each entry. Wiktionary does not have a well-established

[1] https://en.wiktionary.org.

taxonomic lexical relationship within the entries, unlike WordNet, which makes it difficult to be used in semantic similarity algorithms [99].

- With the advent of Wikipedia,[2] most techniques for semantic similarity exploit the abundant text data freely available to train the models [74]. Wikipedia has the text data organized as articles. Each article has a title (concept), neighbors, description, and categories. It is used as both structured taxonomic data and/or as a corpus for training corpus-based methods [100]. The complex category structure of Wikipedia is used as a graph to determine the Information Content of concepts, which in turn aids in calculating the semantic similarity [44].
- BabelNet [88] is a lexical resource that combines WordNet with data available on Wikipedia for each *synset*. It is the largest multilingual semantic ontology available with nearly over 13M *synsets* and 380M semantic relations in 271 languages. It includes over 4M *synsets* with at least one associated Wikipedia page for the English language [22].

3.2 Types of Knowledge-based Semantic Similarity Methods

Based on the underlying principle of how the semantic similarity between words is assessed, knowledge-based semantic similarity methods can be further categorized as edge-counting methods, feature-based methods, and information content-based methods.

3.2.1 Edge-counting Methods. The most straightforward edge counting method is to consider the underlying ontology as a graph connecting words taxonomically and count the edges between two terms to measure the similarity between them. The greater the distance between the terms, the less similar they are. This measure called *path* was proposed by Rada et al. [102] where the similarity is inversely proportional to the shortest path length between two terms. In this edge-counting method, the fact that the words deeper down the hierarchy have a more specific meaning, and that they may be more similar to each other even though they have the same distance as two words that represent a more generic concept, was not taken into consideration. Wu and Palmer [131] proposed *wup* measure, where the depth of the words in the ontology was considered an important attribute. The *wup* measure counts the number of edges between each term and their Least Common Subsumer (LCS). LCS is the common ancestor shared by both terms in the given ontology. Consider two terms denoted as t_1, t_2, their LCS denoted as t_{lcs}, and the shortest path length between them denoted as $min\_len(t_1, t_2)$,
path is measured as,

$$sim_{path}(t_1, t_2) = \frac{1}{1 + min\_len(t_1, t_2)} \tag{1}$$

and *wup* is measured as,

$$sim_{wup}(t_1, t_2) = \frac{2depth(t_{lcs})}{depth(t_1) + depth(t_2)}. \tag{2}$$

Li et al. [62] proposed a measure that takes into account both the minimum path distance and depth. *li* is measured as,

$$sim_{li} = e^{-\alpha min\_len(t_1, t_2)} \cdot \frac{e^{\beta depth(t_{lcs})} - e^{-\beta depth(t_{lcs})}}{e^{\beta depth(t_{lcs})} + e^{-\beta depth(t_{lcs})}}. \tag{3}$$

However, the edge-counting methods ignore the fact that the edges in the ontologies need not be of equal length. To overcome this shortcoming of simple edge-counting methods, feature-based semantic similarity methods were proposed.

[2] http://www.wikipedia.org.

3.2.2 Feature-based Methods. The feature-based methods calculate similarity as a function of properties of the words, such as gloss, neighboring concepts, and so on [123]. Gloss is defined as the meaning of a word in a dictionary; a collection of glosses is called a glossary. There are various semantic similarity methods proposed based on the gloss of words. Gloss-based semantic similarity measures exploit the knowledge that words with similar meanings have more common words in their gloss. The semantic similarity is measured as the extent of overlap between the gloss of the words in consideration. The Lesk measure [11] assigns a value of relatedness between two words based on the overlap of words in their gloss and the glosses of the concepts they are related to in an ontology like WordNet [55]. Jiang et al. [45] proposed a feature-based method where semantic similarity is measured using the glosses of concepts present in Wikipedia. Most feature-based methods take into account common and non-common features between two words/terms. The common features contribute to the increase of the similarity value and the non-common features decrease the similarity value. The major limitation of feature-based methods is its dependency on ontologies with semantic features, and most ontologies rarely incorporate any semantic features other than taxonomic relationships [123].

3.2.3 Information Content-based Methods. Information content (IC) of a concept is defined as the information derived from the concept when it appears in context [122]. A high IC value indicates that the word is more specific and clearly describes a concept with less ambiguity, while lower IC values indicate that the words are more abstract in meaning [133]. The specificity of the word is determined using Inverse Document Frequency (IDF), which relies on the principle that the more specific a word is, the less it occurs in a document. IC-based methods measure the similarity between terms using the IC value associated with them. Resnik and Philip [104] proposed a semantic similarity measure called *res* that measures the similarity based on the idea that if two concepts share a common subsumer, then they share more information, since the *IC* value of the LCS is higher. Considering *IC* represents the IC of the given term, *res* is measured as,

$$sim_{res}(t_1, t_2) = IC_{t_{lcs}}. \tag{4}$$

D. Lin [64] proposed an extension of the *res* measure taking into consideration the *IC* value of both the terms that attribute to the individual information or description of the terms and the *IC* value of their LCS that provides the shared commonality between the terms. *lin* is measured as,

$$sim_{lin}(t_1, t_2) = \frac{2IC_{t_{lcs}}}{IC_{t_1} + IC_{t_2}}. \tag{5}$$

Jiang and Conrath [43] calculate a distance measure based on the difference between the sum of the individual *IC* values of the terms and the *IC* value of their LCS using the below equation:

$$dis_{jcn}(t_1, t_2) = IC_{t_1} + IC_{t_2} - 2ICt_{lcs}. \tag{6}$$

The distance measure replaces the shortest path length in Equation (1), and the similarity is inversely proportional to the above distance. Hence *jcn* is measured as,

$$sim_{jcn}(t_1, t_2) = \frac{1}{1 + dis_{jcn}(t_1, t_2)}. \tag{7}$$

IC can be measured using an underlying corpora or from the intrinsic structure of the ontology itself [108] based on the assumption that the ontologies are structured in a meaningful way. Some of the terms may not be included in one ontology, which provides a scope to use multiple ontologies to calculate their relationship [105]. Based on whether the given terms are both present in a single ontology or not, IC-based methods can be classified as mono-ontological methods or multi-ontological methods. When multiple ontologies are involved, the *IC* of the Least Common Subsumer from both the ontologies are accessed to estimate the semantic similarity values. Jiang et al. [44] proposed IC-based semantic similarity measures based on Wikipedia pages, concepts, and neighbors. Wikipedia was both used as a structured taxonomy as well as a corpus to provide *IC* values.

3.2.4 Combined Knowledge-based Methods. Various similarity measures were proposed combining the various knowledge-based methods. Goa et al. [33] proposed a semantic similarity method based on WordNet ontology where three different strategies are used to add weights to the edges and the shortest weighted path is used to measure the semantic similarity. According to the first strategy, the depths of all the terms in WordNet along the path between the two terms in consideration is added as a weight to the shortest path. In the second strategy, only the depth of the LCS of the terms was added as the weight, and in strategy three, the *IC* value of the terms is added as weight. The shortest weighted path length is now calculated and then non-linearly transformed to produce semantic similarity measures. In comparison, it is shown that strategy three achieved a better correlation to the gold standards in comparison with traditional methods and the two other strategies proposed. Zhu and Iglesias [133] proposed another weighted path measure called *wpath* that adds the *IC* value of the Least Common Subsumer as a weight to the shortest path length. *wpath* is calculated as,

$$sim_{wpath}(t_1, t_2) = \frac{1}{1 + min\_len(t_1, t_2) * k^{IC_{t_{lcs}}}}. \tag{8}$$

This method was proposed to be used in various knowledge graphs (KG), such as WordNet [77], DBPedia [17], YAGO [41], and so on, and the parameter k is a hyperparameter that has to be tuned for different KGs and different domains, as different KGs have a different distribution of terms in each domain. Both corpus-based IC and intrinsic IC values were experimented and corpus IC-based *wpath* measure achieved greater correlation in most of the gold standard datasets.

Knowledge-based semantic similarity methods are computationally simple, and the underlying knowledge base acts as a strong backbone for the models, and the most common problem of ambiguity, such as synonyms, idioms, and phrases, is handled efficiently. Knowledge-based methods can easily be extended to calculate sentence to sentence similarity measure by defining rules for aggregation [58]. Lastra-Díaz et al. [54] developed a software Half-Edge Semantic Measures Library (HESML) to implement various ontology-based semantic similarity measures proposed and have shown an increase in performance time and scalability of the models.

However, knowledge-based systems are highly dependent on the underlying source, resulting in the need to update them frequently, which requires time and high computational resources. Although strong ontologies like WordNet exist for the English language, similar resources are not available for other languages that results in the need for the building of strong and structured knowledge bases to implement knowledge-based methods in different languages and across different domains. Various research works were conducted on extending semantic similarity measures in the biomedical domain [94, 118]. McInnes et al. [71] built a domain-specific model called UMLS to measure the similarity between words in the biomedical domain. With nearly 6,500 world languages and numerous domains, this becomes a serious drawback for knowledge-based systems.

4 CORPUS-BASED SEMANTIC-SIMILARITY METHODS

Corpus-based semantic similarity methods measure semantic similarity between terms using the information retrieved from large corpora. The underlying principle called "distributional hypothesis" [36] exploits the idea that "similar words occur together, frequently"; however, the actual meaning of the words is not taken into consideration. While various techniques were used to construct the vector representation of the text data, several semantic distance measures based on the distributional hypothesis were proposed to estimate the similarity between the vectors. A comprehensive survey of various distributional semantic measures was carried out by Mohammad and Hurst [81], and the different measure and their respective formula are provided in Table 4 in Appendix A . However, among all these measures, the cosine similarity gained significance and has been widely used among NLP researchers to date [81]. In this section, we discuss in detail some of the widely used word embeddings built using distributional hypothesis and some of the significant corpus-based semantic similarity methods.

4.1 Word Embeddings

Word embeddings provide vector representations of words wherein these vectors retain the underlying linguistic relationship between the words [111]. These vectors are computed using different approaches, such as neural networks [75], word co-occurrence matrix [95], or representations in terms of the context in which the word appears [59]. Some of the most widely used pretrained word embeddings include:

- ***word2vec*** [75]: Developed from Google News dataset, containing approximately 3M vector representations of words and phrases, *word2vec* is a neural network model used to produce distributed vector representation of words based on an underlying corpus. There are two different models of *word2vec* proposed: the Continuous Bag of Words (CBOW) and the Skip-gram model. The architecture of the network is rather simple and contains an input layer, one hidden layer, and an output layer. The network is fed with a large text corpus as the input, and the output of the model is the vector representations of words. The CBOW model predicts the current word using the neighboring context words, while the Skip-gram model predicts the neighboring context words given a target word. *word2vec* models are efficient in representing the words as vectors that retain the contextual similarity between words. The word vector calculations yielded good results in predicting the semantic similarity [76]. Many researchers extended the *word2vec* model to propose context vectors [73], dictionary vectors [127], sentence vectors [91], and paragraph vectors [56].
- ***GloVe*** [95]: $GloVe$, developed by Stanford University, relies on a global word co-occurrence matrix formed based on the underlying corpus. It estimates similarity based on the principle that words similar to each other occur together. The co-occurrence matrix is populated with occurrence values by doing a single pass over the underlying large corpora. $GloVe$ model was trained using five different corpora, mostly Wikipedia dumps. While forming vectors, words are chosen within a specified context window owing to the fact that words far away have less relevance to the context word in consideration. The $GloVe$ loss function minimizes the least-square distance between the context window co-occurrence values and the global co-occurrence values [55]. $GloVe$ vectors were extended to form contextualized word vectors to differentiate words based on context [70].
- ***fastText*** [18]: Facebook AI researchers developed a word-embedding model that builds word vectors based on Skip-gram models where each word is represented as a collection of character n-grams. $fastText$ learns word embeddings as the average of its character embeddings, thus accounting for the morphological structure of the word, which proves efficient in various languages, such as Finnish and Turkish. Even out-of-the-vocabulary words are assigned word vectors based on their characters or subunits.
- ***Bidirectional Encoder Representations from Transformers (BERT)*** [29]: Devlin et al. [29] proposed a pretrained transformer-based word embeddings that can be fine-tuned by adding a final output layer to accommodate the embeddings to different NLP tasks. BERT uses the transformer architecture proposed by Vaswani et al. [128], which produces

attention-based word vectors using a bi-directional transformer encoder. The BERT framework involves two important processes, namely, "pretraining" and "fine-tuning." The model is pretrained using a corpus of nearly 3,300M words from both the Book corpus and English Wikipedia. Since the model is bidirectional to avoid the possibility of the model knowing the token itself, when training from both directions the pretraining process is carried out in two different ways. In the first task, random words in the corpus are masked and the model is trained to predict these words. In the second task, the model is presented with sentence pairs from the corpus, in which 50% of the sentences are actually consecutive while the remaining are random pairs. The model is trained to predict if the given sentence pair are consecutive or not. In the "fine-tuning" process, the model is trained for the specific downstream NLP task at hand. The model is structured to take as input both single sentences and multiple sentences to accommodate a variety of NLP tasks. To train the model to perform a question answering task, the model is provided with various question-answer pairs, and all the parameters are fine-tuned in accordance with the task. BERT embeddings provided state-of-the-art results in the STS-B data set with a Spearman's correlation of 86.5%, outperforming other BiLSTM models including ELMo [96].

Word embeddings are used to measure semantic similarity between texts of different languages by mapping the word embedding of one language over the vector space of another. On training with a limited yet sufficient number of translation pairs, the translation matrix can be computed to enable the overlap of embeddings across languages [35]. One of the major challenges faced when deploying word-embeddings to measure similarity is Meaning Conflation Deficiency. It denotes that word embeddings do not attribute to the different meanings of a word that pollutes the semantic space with noise by bringing irrelevant words closer to each other. For example, the words "finance" and "river" may appear in the same semantic space, since the word "bank" has two different meanings [20]. It is critical to understand that word-embeddings exploit the distributional hypothesis for the construction of vectors and rely on large corpora, hence, they are classified under corpus-based semantic similarity methods. However, deep neural network based–methods and most hybrid semantic similarity methods use word embeddings to convert the text data to high-dimensional vectors, and the efficiency of these embeddings play a significant role in the performance of the semantic similarity methods [60, 79].

4.2 Types of Corpus-based Semantic Similarity Methods

Based on the underlying methods using which the word-vectors are constructed, there are is a wide variety of corpus-based methods, some of which are discussed in this section.

4.2.1 Latent Semantic Analysis (LSA) [51]. LSA is one of the most popular and widely used corpus-based techniques used for measuring semantic similarity. A word co-occurrence matrix is formed where the rows represent the words and columns represent the paragraphs, and the cells are populated with word counts. This matrix is formed with a large underlying corpus, and dimensionality reduction is achieved by a mathematical technique called Singular Value Decomposition (SVD). SVD represents a given matrix as a product of three matrices, where two matrices represent the rows and columns as vectors derived from their eigenvalues and the third matrix is a diagonal matrix that has values that would reproduce the original matrix when multiplied with the other two matrices [52]. SVD reduces the number of columns while retaining the number of rows, thereby preserving the similarity structure among the words. Then each word is represented as a vector using the values in its corresponding rows and semantic similarity is calculated as the cosine value between these vectors. LSA models are generalized by replacing words with texts and columns with different samples and are used to calculate the similarity between sentences, paragraphs, and documents.

4.2.2 Hyperspace Analogue to Language (HAL) [68]. HAL builds a word co-occurrence matrix that has both rows and columns representing the words in the vocabulary and the matrix elements are populated with association strength values. The association strength values are calculated by sliding a "window," the size of which can be varied, over the underlying corpus. The strength of association between the words in the window decreases with the increase in their distance from the focused word. For example, in the sentence "This is a survey of various semantic similarity measures," the words *"survey"* and *"variety"* have greater association value than the words *"survey"* and *"measures."* Word vectors are formed by taking into consideration both the row and column of the given word. Dimensionality reduction is achieved by removing any columns with low entropy values. The semantic similarity is then calculated by measuring the Euclidean or Manhattan distance between the word vectors.

4.2.3 Explicit Semantic Analysis (ESA) [31]. ESA measures semantic similarity based on Wikipedia concepts. The use of Wikipedia ensures that the proposed method can be used over various domains and languages. Since Wikipedia is constantly updated, the method is adaptable to the changes over time. First, each concept in Wikipedia is represented as an attribute vector of the words that occur in it, then an inverted index is formed, where each word is linked to all the concepts it is associated with. The association strength is weighted using the TF-IDF technique, and the concepts weakly associated with the words are removed. Thus, the input text is represented by weighted vectors of concepts called the "interpretation vectors." Semantic similarity is measured by calculating the cosine similarity between these word vectors.

4.2.4 Word-alignment Models [120]. Word-alignment models calculate the semantic similarity of sentences based on their alignment over a large corpus [24, 47, 119]. The second, third, and fifth positions in SemEval tasks 2015 were secured by methods based on word alignment. The unsupervised method that was in the fifth place implemented the word-alignment technique based on Paraphrase Database (PPDB) [32]. The system calculates the semantic similarity between two sentences as a proportion of the aligned context words in the sentences over the total words in both the sentences. The supervised methods that were at the second and third place used *word2vec* to obtain the alignment of the words. In the first method, a sentence vector is formed by computing the "component-wise average" of the words in the sentence, and the cosine similarity between these sentence vectors is used as a measure of semantic similarity. The second supervised method takes into account only those words that have a contextual semantic similarity [120].

4.2.5 Latent Dirichlet Allocation (LDA) [117]. LDA is used to represent a topic or the general idea behind a document as a vector rather than every word in the document. This technique is widely used for topic modeling tasks, and it has the advantage of reduced dimensionality considering that the topics are significantly less than the actual words in a document [117]. One of the novel approaches to determine document-to-document similarity is the use of vector representation of documents and calculates the cosine similarity between the vectors to ascertain the semantic similarity between documents [16].

4.2.6 Normalized Google Distance [25]. NGD measures the similarity between two terms based on the results obtained when the terms are queried using the Google search engine. It is based on the assumption that two words occur together more frequently in web pages if they are more related. Given two terms t_1 and t_2, the following formula is used to calculate the NGD between the two terms:

$$NGD(x, y) = \frac{max\ \{log\ f(t_1), log\ f(t_2)\} - log\ f(t_1, t_2)}{log\ G - min\ \{log\ f(t_1), log\ f(t_2)\}}, \tag{9}$$

where the functions $f(x)$ and $f(y)$ return the number of hits in Google search of the given terms, $f(x, y)$ returns the number of hits in Google search when the terms are searched together, and G represents the total number of pages in the overall Google search. NGD is widely used to measure semantic relatedness rather than semantic similarity, because related terms occur together more frequently in web pages though they may have opposite meaning.

4.2.7 Dependency-based Models [1]. Dependency-based approaches ascertain the meaning of a given word or phrase using the neighbors of the word within a given window. The dependency-based models initially parse the corpus based on its distribution using Inductive Dependency Parsing [90]. For every given word, a "syntactic context template" is built considering both the nodes preceding and succeeding the word in the built parse tree. For example, the phrase *"thinks <term> delicious"* could have a context template as *"pizza, burger, food."* Vector representation of a word is formed by adding each window across the location that has the word in consideration, as its root word, along with the frequency of the window of words appearing in the entire corpus. Once this vector is formed, semantic similarity is calculated using cosine similarity between these vectors. Levy et al. [59] proposed DEPS embedding as a word-embedding model based on dependency-based bag of words. This model was tested with the WS353 dataset where the task was to rank the similar words above the related words. On plotting a recall precision curve, the DEPS curve showed greater affinity towards similarity rankings over BoW methods taken in comparison.

4.2.8 Kernel-based Models [115]. Kernel-based methods were used to find patterns in text data, thus enabling detecting similarity between text snippets. Two major types of kernels were used in text data, namely, the string or sequence kernel [23] and the tree kernel [84]. Moschitti et al. [84] proposed tree kernels in 2007 that contain three different sub-structures in the tree kernel space, namely, a subtree—a tree whose root is not a leaf node along with its children nodes; a subset tree—a tree whose root is not a leaf node but not incorporating all its children nodes and does not break the grammatical rules; and a partial tree—a tree structure closely similar to subset tree but it does not always follow the grammatical rules. Tree kernels are widely used in identifying a structure in input sentences based on constituency or dependency, taking into consideration the grammatical rules of the language. Kernels are used by machine learning algorithms like Support Vector Machines (SVMs) to adapt to text data in various tasks, such as Semantic Role Labelling, Paraphrase Identification [28], Answer Extraction [85], Question-answer classification [86], Relational text categorization [83], Answer Re-ranking in QA tasks [112], and Relational text entailment [87]. Severyn et al. [113] proposed a kernel-based semantic similarity method that represents the text directly as "structural objects" using Syntactic tree kernel [27] and Partial tree kernels [82]. The kernel function then combines the tree structures with semantic feature vectors from two of the best performing models in STS 2012, namely, UKP [12] and Takelab [110], and some additional features including cosine similarity scores based on named entities, part of speech tags, and so on. The authors compare the performance of the model constructed using four different tree structures, namely, shallow tree, constituency tree, dependency tree, phrase-dependency tree, and the above-mentioned feature vectors. They establish that the tree kernel models perform better than all feature vectors combined. The model uses Support Vector Regression to obtain the final similarity score, and it can be useful in various downstream NLP applications, such as question-answering, text-entailment extraction, and so on. Amir et al. [9] proposed another semantic similarity algorithm using kernel functions. They used constituency-based tree kernels where the sentence is broken down into subject, verb, and object based on the assumption most semantic properties of the sentence are attributed to these components. The input sentences are parsed using the Stanford Parser to extract various combinations of subject, verb, and object. The similarity between the various components of the given sentences is calculated using a knowledge base, and different averaging techniques are used to average the similarity values to estimate the overall similarity, and the best among them is chosen based on the root mean squared error value for a particular dataset. In recent research, deep learning methods have been used to replace the traditional machine learning models and efficiently use the structural integrity of kernels in the embedded feature extraction stage [26, 28]. The model that achieved the best results in SemEval-2017 Task 1, proposed by Tian et al. [125], uses kernels to extract features from text data to calculate similarity. The model proposed an ensemble model that used both traditional NLP methods and deep learning methods. Two different features are, namely, the sentence pair matching features, and single sentence features were used to predict the similarity values using regressors, which added nonlinearity to the prediction. In single sentence feature extraction, dependency-based tree kernels are used to extract the dependency features in one given sentence, and in sentence pair matching features, constituency-based parse tree kernels are used to find the common sub-constructs among the three different characterizations of tree kernel spaces. The final similarity score is accessed by averaging the traditional NLP similarity value and the deep learning–based similarity value. The model achieved a Pearson's correlation of 73.16% in the STS dataset.

4.2.9 Word-attention Models [57]. In most of the corpus-based methods, all text components are considered to have equal significance; however, human interpretation of measuring similarity usually depends on keywords in a given context. Word attention models capture the importance of the words from underlying corpora [67] before calculating the semantic similarity. Different techniques such as word frequency, alignment, and word association are used to capture the attention-weights of the text in consideration. Attention Constituency Vector Tree (ACV-Tree) proposed by Le et al. [57] is similar to a parse tree where one word of a sentence is made the root and the remainder of the sentence is broken as a Noun Phrase (NP) and a Verb Phrase (VP). The nodes in the tree store three different attributes of the word into consideration: the word vector determined by an underlying corpus, the attention-weight, and the "modification-relations" of the word. The modification relations can be defined as the adjectives or adverbs that modify the meaning of another word. All three components are linked to form the representation of the word. A tree kernel function is used to determine the similarity between two words based on the equation below:

$$TreeKernel(T_1, T_2) = \sum_{n_1 \in N_{T_1}} \sum_{n_2 \in N_{T_2}} \Delta(n_1, n_2), \tag{10}$$

$$\Delta(n_1, n_2) = \begin{cases} 0, \text{ if } (n_1 \text{ and / or } n_2 \text{ are non-leaf-nodes) and } n_1 \neq n_2, \\ Aw \times SIM(vec_1, vec_2), \text{ if } n_1, n_2 \text{ are leaf nodes}, \\ \mu(\lambda^2 + \sum_{p=1}^{l_m} \delta_p(c_{n_1}, c_{n_2})), \text{ otherwise}, \end{cases} \tag{11}$$

where n_1, n_2 represent the the nodes, $SIM(vec_1, vec_2)$ measures the cosine similarity between the vectors, $\delta_p(.)$ calculates the number of common subsequences of length p, λ, μ denote the decay factors for length of the child sequences and the height of the tree, respectively, c_{n_1}, c_{n_2} refer to the children nodes, and $l_m = min(length(c_{n_1}), length(c_{n_2}))$. The algorithm is tested using the STS benchmark datasets and has shown better performance in 12 out of 19 chosen STS Datasets [57, 101].

Unlike knowledge-based systems, corpus-based systems are language- and domain-independent [8]. Since they are dependent on statistical measures, the methods can be easily adapted across various languages using an effective corpus. With the growth of the internet, building corpora of most languages or domains has become rather easy. Simple web-crawling techniques can be used to build large corpora [13]. However, the corpus-based methods do not take into consideration the actual meaning of the words. The other challenge faced by corpus-based methods is the need to process the large corpora built, which is a rather time-consuming and resource-dependent task. Since the performance of the algorithms largely depends on the underlying corpus, building an efficient corpus is paramount. Though efforts are made by researchers to build a clean and efficient corpus like the C4 corpus built by web crawling and five steps to clean the corpus [103], an "ideal corpus" is still not defined by researchers.

5 DEEP NEURAL NETWORK–BASED METHODS

Semantic similarity methods have exploited the recent developments in neural networks to enhance performance. The most widely used techniques include Convolutional Neural Networks (CNN), Long Short Term Memory (LSTM), Bidirectional Long Short Term Memory (Bi-LSTM), and Recursive Tree LSTM. Deep neural network models are built based on two fundamental operations: convolution and pooling. The convolution operation in text data may be defined as the sum of the element-wise product of a sentence vector and a weight matrix. Convolution operations are used for feature extraction. Pooling operations are used to eliminate features that have a negative impact and only consider those feature values that have a considerable impact on the task at hand. There are different types of pooling operations, and the most widely used is Max pooling, where only the maximum value in the given filter space is selected. This section describes some of the methods that deploy deep neural networks to estimate semantic similarity between text snippets. Although the methods described below exploit word embeddings built using large corpora, deep neural networks are used to estimate the similarity between the word embeddings, hence they are classified separately from corpus-based methods.

5.1 Types of Deep Neural Network–based Semantic Similarity Methods:

- Wang et al. [130] proposed a model to estimate semantic similarity between two sentences based on lexical decomposition and composition. The model uses *word2vec* pretrained embeddings to form a vector representation of the sentences s_1 and s_2. A similarity matrix M of dimension i x j is built where i and j are the number of words in sentence 1 (S_1) and sentence 2 (S_2), respectively. The cells of the matrix are populated with the cosine similarity between the words in the indices of the matrix. Three different functions are used to construct semantic matching vectors $\vec{s_1}$ and $\vec{s_2}$, the global, local, and max function. The global function constructs the semantic matching vector of S_1 by taking the weighted sum of the vectors, of all the words in S_2, the local function, takes into consideration only word vectors within a given window size, and the max function takes only the vectors of the words that have the maximum similarity. The second phase of the algorithm uses three different decomposition functions—rigid, linear, and orthogonal—to estimate the similarity component and the dissimilarity component between the sentence vectors and the semantic matching vectors. Both the similarity component and the dissimilarity component vectors are passed through a two-channel convolution layer followed by a single max-pooling layer. The similarity is then calculated using a sigmoid layer that estimates the similarity value within the range of 0 and 1. The model was tested using the QASent dataset [129] and the WikiQA dataset [72]. The two measures used to estimate the performance are mean average precision (MAP) and mean reciprocal rank (MRR). The model achieves the best MAP in the QASent dataset and the best MAP and MRR in the WikiQA dataset. Yang Shao [114] proposed a semantic similarity algorithm that exploits the recent development in neural networks using $GloVe$ word embeddings. Given two sentences, the model predicts a probability distribution over set semantic similarity values. The pre-processing steps involve the removal of punctuation, tokenization, and using *GloVe* vectors to replace words with word embeddings. The length of the input is set to 30 words, which is achieved by removal or padding as deemed necessary. Some special hand-crafted features, such as flag values indicating if the words or numbers occurred in both the sentences and POS tagging one hot encoded values, were added to the *GloVe* vectors. The vectors are then fed to a CNN with 300 filters and one max-pooling layer that is used to form the sentence vectors. ReLU activation function is used in the convolution layer. The semantic difference between the vectors is calculated by the element-wise absolute difference and the element-wise multiplication of the two sentence vectors generated. The vectors are further passed through two fully connected layers, which predicts the probability distribution of the semantic similarity values. The model performance was evaluated using the SemEval datasets where the model was ranked third in SemEval 2017 dataset track.

- The LSTM networks are a special kind of Recurrent Neural Network (RNN). While processing text data, it is essential for the networks to remember previous words, to capture the context, and RNNs have the capacity to do so. However, not all the previous content has significance over the next word/phrase, hence RNNs suffer the drawback of long-term dependency. LSTMs are designed to overcome this problem. LSTMs have gates that enable the network to choose the content it has to remember. For example, consider the text snippet, *"Mary is from Finland. She is fluent in Finnish. She loves to travel."* While we reach the second sentence of the text snippet, it is essential to remember the words *"Mary"* and *"Finland."* However, on reaching the third sentence, the network may forget the word *"Finland."* The architecture of LSTMs allows this. Many researchers use the LSTM architecture to measure semantic similarity between blocks of text. Tien et al. [126] uses a network combined with LSTM and CNN to form a sentence embedding from pretrained word embeddings followed by an LSTM architecture to predict their similarity. Tai et al. [124] proposed an LSTM architecture to estimate the semantic similarity between two given sentences. Initially, the sentences are converted to sentence representations using Tree-LSTM over the parse tree of the sentences. These sentence representations are then fed to a neural network that calculates the absolute distance between the vectors and the angle between the vectors. The experiment was conducted using the SICK dataset, and the similarity measure varies with the range 1 to 5. The hidden layer consisted of 50 neurons and the final softmax layer classifies the sentences over the given range. The Tree-LSTM model achieved better Pearson's and Spearman's correlation in the gold standard datasets than the other neural network models in comparison.
- He and Lin [39] proposed a hybrid architecture using Bi-LSTM and CNN to estimate the semantic similarity of the model. Bi-LSTMs have two LSTMs that run parallel, one from the beginning of the sentence and one from the end, thus capturing the entire context. In their model, He and Lin use Bi-LSTM for context modelling. A pairwise word interaction model is built that calculates a comparison unit between the vectors derived from the hidden states of the two LSTMs using the below formula:

  $$CoU(\vec{h_1}, \vec{h_2}) = \{cos(\vec{h_1}, \vec{h_2}), euc(\vec{h_1}, \vec{h_2}), manh((\vec{h_1}, \vec{h_2})\}, \tag{12}$$

  where $\vec{h_1}$ and $\vec{h_2}$ represent the vectors from the hidden state of the LSTMs and the functions $cos()$, $euc()$, $manh()$ calculate the Cosine distance, Euclidean distance, and Manhattan distance, respectively. This model is similar to other recent neural network–based word attention models [7, 10]. However, attention weights are not added; rather, the distances are added as weights. The word interaction model is followed by a similarity focus layer where weights are added to the word interactions (calculated in the previous layers) based on their importance in determining the similarity. These re-weighted vectors are fed to the final convolution network. The network is composed of alternating spatial convolution layers and spatial max pooling layers, and ReLU activation function is used and at the network ends with two fully connected layers followed by a LogSoftmax layer to obtain a non-linear solution. This model outperforms the previously mentioned Tree-LSTM model on the SICK dataset.

- Lopez-Gazpio et al. [67] proposed an extension to the existing Decomposable Attention Model (DAM) proposed by Parikh et al. [92], which was originally used for Natural Language Inference (NLI). NLI is used to categorize a given text block to a particular relation, such as entailment, neutral, or contradiction. The DAM model used feed-forward neural networks in three consecutive layers: the attention layer, comparison layer, and aggregation layer. Given two sentences, the attention layer produces two attention vectors for each sentence by finding the overlap between them. The comparison layer concatenates the attention vectors with the sentence vectors to form a single representative vector for each sentence. The final aggregation layer flattens the vectors and calculates the probability distribution over the given values. Lopez-Gazpio et al. [67] used word n-grams to capture attention in the first layer instead of individual words. $n - grams$ may be defined as a sequence of n words that are contiguous with the given word, and n-grams are used to capture the context in various NLP tasks. To accommodate n-grams, an RNN is added to the attention layer. Variations were proposed by replacing RNN with LSTM, and CNN. The model was used for semantic similarity calculations by replacing the final classes of entailment relationships with semantic similarity ranges from 0 to 5. The models achieved better performance in capturing the semantic similarity in the SICK dataset and the STS benchmark dataset when compared to DAM and other models, such as Sent2vec [91] and BiLSTM, among others.
- **Transformer-based models:** Vaswani et al. [128] proposed a transformer model that relies on attention mechanisms to capture the semantic properties of words in the embeddings. The transformer has two parts: "encoder" and "decoder." The encoder consists of layers of multi-head attention mechanisms followed by a fully connected feed-forward neural network. The decoder is similar to the encoder with one additional layer of multi-head attention that captures the attention weights in the output of the encoder. Although this model was proposed for the machine translation task, Devlin et al. [29] used the transformer model to generate BERT word embeddings. Sun et al. [121] proposed a multi-tasking framework using transformers called ERNIE 2.0. In this framework, the model is continuously pretrained, i.e., when a new task is presented, the model is fine-tuned to accommodate the new task while retaining the previously gained knowledge. The model outperformed BERT. XLNet, proposed by Yang et al. [132], used an autoregression model as opposed to the autoencoder model and outperformed BERT and ERNIE 2.0. A number of variations of BERT models were proposed based on the corpus used to train the model and by optimizing the computational resources. Lan et al. [50] proposed ALBERT, with two techniques to reduce the computational complexity of BERT, namely, "factorized embedding parameterization" and "cross-layer parameter sharing." ALBERT outperformed all the above three models. Other variations of BERT models that use transformers include TinyBERT [46], RoBERTa [65, 109], and a domain-specific variation trained on a scientific corpus with a focus on the BioMedical domain the SciBERT [15]. Raffel et al. [103] proposed a transformer model with a well-defined corpus called "Colossal Clean Crawled Corpus," or C4, to train the model named T5-11B. Unlike BERT, they adopt a "text-to-text framework," where the input sequence is attached with a token to identify the NLP task to be performed, thus eliminating the two stages pretraining and fine-tuning. They propose five different versions of their

Table 3. Pearson's Correlation of Various Transformer-based Models on STS Benchmark Dataset

Model Name	Title	Year	Pearson's Correlation
T5-11B	Exploring the Limits of Transfer Learning with a Unified Text-to-text Transformer	2019	0.925
XLNet	XLNet: Generalized Autoregressive Pretraining for Language Understanding	2019	0.925
ALBERT	ALBERT: A Lite BERT for Self-supervised Learning of Language Representations	2019	0.925
RoBERTa	RoBERTa: A Robustly Optimized BERT Pretraining Approach	2019	0.922
ERNIE 2.0	ERNIE 2.0: A Continual Pretraining Framework for Language Understanding	2019	0.912
DistilBERT	DistilBERT, a distilled version of BERT: smaller, faster, cheaper, and lighter	2019	0.907
TinyBERT	TinyBERT: Distilling BERT for Natural Language Understanding	2019	0.799

model based on the number of trainable parameters each model has, namely, (1) T5-Small, (2) T5-Base, (3) T5-Large, (4) T5-3B, and (5) T511B, and they have 60M, 220M, 770M, 3B, and 11B parameters, respectively. This model outperformed all other transformer-based models and achieved state-of-the-art results. As a result of their study, they confirm that the performance of the models increases with increased data and computational power, and the performance can be further improved if larger models are built, and it is important to note that to replicate their best model, five GPUs are required, among other resources. A compilation of the various transformer-based models and their Pearson's correlation on the STS-B dataset is provided below in Table 3.

Deep neural network–based methods outperform most of the traditional methods, and the recent success of transformer-based models has served as a breakthrough in semantic similarity research. However, implementation of deep-learning models requires large computational resources. Though variations of the models to minimize the computational resources are being proposed, we see that the performance of the model takes a hit as well; for example, TinyBERT [46]. And the performance of the models is largely increased by the use of a bigger corpus, which again poses the challenge of building an ideal corpus. Most deep-learning models are "black-box" models, and it is difficult to ascertain the features based on which the performance is achieved, hence it becomes difficult to be interpreted, unlike in the case of corpus-based methods that have a strong mathematical foundation. Various fields, such as finance, insurance, and so on, that deal with sensitive data may be reluctant to deploy deep neural network–based methods due to their lack of interpretability.

6 HYBRID METHODS

Based on all the previously discussed methods, we see that each has its advantages and disadvantages. The knowledge-based methods exploit the underlying ontologies to disambiguate synonyms, while corpus-based methods are versatile, as they can be used across languages. Deep neural network–based systems, though computationally expensive, provide better results. However, many researchers have found ways to exploit the best of each method and build hybrid models to measure semantic similarity. In this section, we describe the methodologies used in some of the widely used hybrid models.

6.1 Types of Hybrid Semantic Similarity Methods

- Novel Approach to a Semantically Aware Representation of Items (NASARI) [21]: Camacho Collados et al. [21] proposed an approach the *NASARI* where the knowledge source BabelNet is used to build a corpus based on which vector representation for concepts (words or group of words) are formed. Initially, the Wikipedia pages associated with a given concept, in this case, the *synset* of BabelNet, and all the outgoing links from the given page are used to form a sub-corpus for the specific concept. The sub-corpus is further expanded with the Wikipedia pages of the hypernyms and hyponyms of the concept in the BabelNet network. The entire Wikipedia is considered as the reference corpus. Two different types of vector representation were proposed. In the first method, weighted vectors were formed using lexical specificity. Lexical specificity is a statistical method of identifying the most representative words for a given text based on the hypergeometric distribution (sampling without replacement). Let "T and t" denote the total content words in the reference corpus RC and sub-corpus SC, respectively, and "F and f" denote the frequency of the given word in the reference corpus RC and sub-corpus SC, respectively, then lexical specificity can be represented by the below equation:

$$spec(T, t, F, f) = -log_{10}P(X \geq f). \tag{13}$$

X represents a random variable that follows a hypergeometric relation with the parameters T, t, and F, and $P(X \geq f)$ is defined as,

$$P(X \geq f) = \sum_{i=f}^{F} P(X = i). \tag{14}$$

$P(X = i)$ is the probability of a given term appearing exactly i times in the given sub-corpus in hypergeometric distribution with T, t, and F. The second method forms a cluster of words in the sub-corpus that share a common hypernym in the WordNet taxonomy that is embedded in BabelNet. The specificity is then measured based on the frequency of the hypernym and all its hyponyms in the taxonomy, even those that did not occur in the given sub-corpus. This clustering technique forms a unified representation of the words that preserve the semantic properties. The specificity values are added as weights in both methods to rank the terms in a given text. The first method of vector representation was called $NASARI_{lexical}$, and the second method was called $NASARI_{unified}$. The similarity between these vectors is calculated using the measure called Weighted Overlap [98] as,

$$WO(v_1, v_2) = \sqrt{\frac{\sum_{d \in O}(rank(d, \vec{v_1}) + rank(d, \vec{v_2}))^{-1}}{\sum_{i=1}^{|O|(2i)^{-1}}}}, \tag{15}$$

where O denotes the overlapping terms in each vector, and $rank(d, \vec{v_i})$ represents the rank of the term d in the vector v_i.

Camacho Collados et al. [22] proposed an extension to their previous work and proposed a third vector representation by mapping the lexical vector to the semantic space of word embeddings produced by complex word embedding techniques like *word2vec*. This representation was called as $NASARI_{embedded}$. The similarity is measured as the cosine similarity between these vectors. All three methods were tested across the gold standard datasets M&C, WS-Sim, and SimLex-999. $NASARI_{lexical}$ achieved higher Pearson's and Spearman's correlation on average over the three datasets in comparison with other methods, such as ESA, *word2vec*, and *lin*.

- **Most Suitable Sense Annotation (MSSA)** [106]: Ruas et al. proposed three different methodologies to form word-sense embeddings. Given a corpus, the word-sense disambiguation step is performed using one of the three proposed methods: Most Suitable Sense Annotation (MSSA), Most Suitable Sense Annotation N Refined (MSSA-NR), and Most Suitable Sense Annotation Dijkstra (MSSA-D). Given a corpus, each word in the corpus is associated with a *synset* in the WordNet ontology and *"gloss-average-vector"* is calculated for each *synset*. The gloss-average-vector is formed using the vector representation of the words in the gloss of each *synset*. MSSA calculates the gloss-average-vector using a small window of words and returns the *synset* of the word that has the highest gloss-average-vector value. MSSA-D, however, considers the entire document from the first word to the last word and then determines the associated *synset*. These two systems use Google News vectors[3] to form the synset-embeddings. MSSA-NR is an iterative model, where the first pass produces the synset-embeddings, which are fed back in the second pass as a replacement to gloss-average-vectors to produce more refined synset-embeddings. These synset-embeddings are then fed to a *word2vec* CBOW model to produce multi-sense word embeddings that are used to calculate the semantic similarity. This combination of MSSA variations and *word2vec* produced solid results in gold standard datasets, such as R&G, M&C, WS353-Sim, and SimLex-999 [106].
- **Unsupervised Ensemble Semantic Textual Similarity Methods (UESTS)** [38]: Hassan et al. proposed an ensemble semantic similarity method based on an underlying unsupervised word-aligner. The model calculates the semantic similarity as the weighted sum of four different semantic similarity measures between sentences S_1 and S_2 using the equation below:

$$\begin{aligned} sim_{USETS}(S_1, S_2) = \alpha * sim_{WAL}(S_1, S_2) + \beta * sim_{SC}(S_1, S_2) \\ + \gamma * sim_{embed}(S_1, S_2) + \theta * sim_{ED}(S_1, S_2). \end{aligned} \tag{16}$$

  $sim_{WAL}(S_1, S_2)$ calculates similarity using a synset-based word aligner. The similarity between text is measured based on the number of shared neighbors each term has in the BableNet taxonomy. $sim_{SC}(S_1, S_2)$ measures similarity using soft cardinality measure between the terms in comparison. The soft cardinality function treats each word as a set and the similarity between them as an intersection between the sets. $sim_{embed}(S_1, S_2)$ forms word vector representations using the word embeddings proposed by Baroni et al. [14]. Then similarity is measured as the cosine value between the two vectors. $sim_{ED}(S_1, S_2)$ is a measure of dissimilarity between two given sentences. The edit distance is defined as the minimum number of edits it takes to convert one sentence to another. The edits may involve insertion, deletion, or substitution. $sim_{ED}(S_1, S_2)$ uses word-sense edit distance where word senses are taken into consideration instead of actual words themselves. The hyperparameters α, β, γ, and θ were tuned to values between 0 and 0.5 for different STS benchmark datasets. The ensemble model outperformed the STS benchmark unsupervised models in the 2017 SemEval series on various STS benchmark datasets.

Hybrid methods exploit both the structural efficiency offered by knowledge-based methods and the versatility of corpus-based methods. Many studies have been conducted to build multi-sense embeddings to incorporate the actual meaning of words into word vectors. Iacobacci et al. formed word embeddings called "Sensembed" by using BabelNet to form a sense annotated corpus and

[3] https://code.google.com/archive/p/word2vec/.

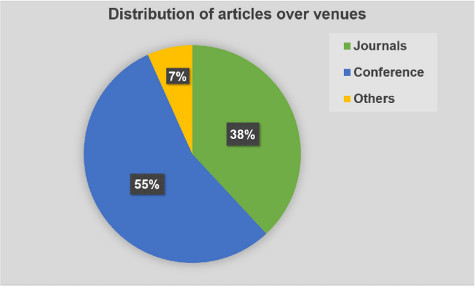

Fig. 2. Distribution of articles over venues.

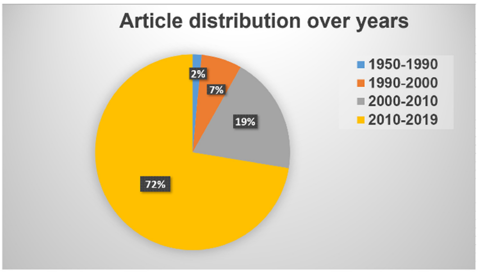

Fig. 3. Distribution of articles over years.

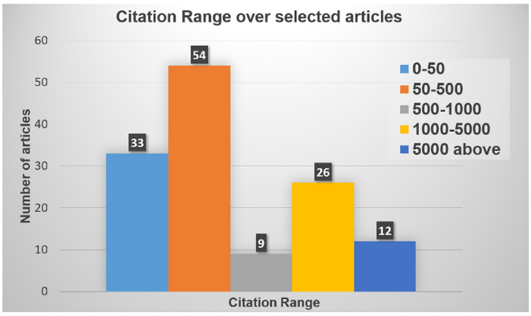

Fig. 4. Distribution of citation range over the articles.

then using *word2vec* to build word vectors, thus having different vectors for different senses of the words. As we can see, hybrid models compensate for the shortcomings of one method by incorporating other methods. Hence, the performance of hybrid methods is comparatively high. The first five places of SemEval 2017 semantic similarity tasks were awarded to ensemble models, which clearly shows the shift in research towards hybrid models [24].

7 ANALYSIS OF SURVEY

This section discusses the method used to build this survey article and provides an overview of the various research articles taken into consideration.

7.1 Search Strategy

The articles considered for this survey were obtained using the Google Scholar search engine and the keywords used include *"semantic similarity, word embedding, knowledge-based methods, corpus-based methods, deep neural network-based semantic similarity, LSTM, text processing, and semantic similarity datasets."* The results of the search were fine-tuned using various parameters, such as the Journal Ranking, Google Scholar Index, number of citations, year of publication, and so on. Only articles published in journals with Scimago Journal ranking of Quartile 1 and conferences that have a Google metrics H-index above 50 were considered. Exceptions were made for some articles that have a higher impact and relevance. The table of references sorted by the year of publication is included in Appendix B as Table 5. The table records (1) Title, (2) Year of Publication, (3) Author Names, (4) Venue, (5) SJR Quartile (for journals), (6) H-Index, and (7) Number of Citations (as of 02.04.2020). Some of the statistical results of the chosen articles are shown in the figures below. These figures highlight the quality of the articles chosen that, in turn, highlights the quality of the survey. Figure 2 shows the distribution of the referenced articles over conferences, journals,

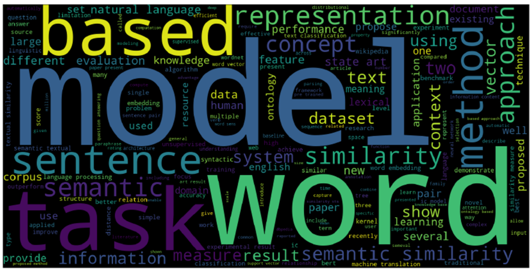

Fig. 5. World Cloud representing the collection of words from the abstracts of the papers used in the survey.

and others. 55% of the articles are from conferences, and 38% of the articles are from journals. The remaining 7% of the articles are from arXiv and books. However, they have rather a high impact in relation to the topic of the survey. Figure 3 highlights the distribution of the selected articles over the years. Nearly 72% of the chosen articles areworks carried out after 2010, and the remaining 28% of the articles represent the traditional methods adopted during the early stages of the evolution of semantic similarity. Figure 4 represents the citation range of the articles. 54% of the articles have 50 to 500 citations, 26% have 1,000–5,000 citations, and 12% of the articles have more than 5,000 citations. We see that 33% of the articles have citations below 50; however, all these articles are published after the year 2017, which accounts for the fewer citations.

7.2 Word-cloud Generation:

We implemented a simple Python code to generate a word cloud using the abstracts from all the articles used in this survey. The abstracts from all the 118 articles were used to build a dataset that was then used in the Python code. The extracted abstracts are initially pre-processed by converting the text to lower case, removing the punctuation, and removing the most commonly used English stop words available in the nltk[4] library. Then the word cloud is built using the *wordcloud* python library. The word cloud thus built is shown in Figure 5. From the word cloud, we infer that though different keywords were used in our search for articles, the general focus of the selected articles is semantic similarity. In a word cloud, the size of the words is proportional to the frequency of use of these words. The word *"word"* is considerably bigger than the word *"sentence,"* showing that most of the research works focus on word-to-word similarity rather than sentence-to-sentence similarity. We could also infer that the words *"vector"* and *"representation"* have been used more frequently than the words *"information," "context,"* and *"concept,"* indicating the influence of corpus-based methods over knowledge-based methods. With the given word cloud, we showcase the focus of the survey graphically.

8 CONCLUSION

Measuring semantic similarity between two text snippets has been one of the most challenging tasks in the field of Natural Language Processing. Various methodologies have been proposed over the years to measure semantic similarity, and this survey discusses the evolution, advantages, and disadvantages of these methods. Knowledge-based methods take into consideration the actual meaning of text; however, they are not adaptable across different domains and languages. Corpus-based methods have a statistical background and can be implemented across languages, but they do not take into consideration the actual meaning of the text. Deep neural network–based methods show better performance, but they require high computational resources and lack interpretability. Hybrid methods are formed to take advantage of the benefits from different

[4] http://www.nltk.org/.

methods, compensating for the shortcomings of each other. It is clear from the survey that each method has its advantages and disadvantages, and it is difficult to choose one best model; however, most recent hybrid methods have shown promising results over other independent models. While the focus of recent research is shifted towards building more semantically aware word embeddings, and the transformer models have shown promising results, the need for determining a balance between computational efficiency and performance is still a work in progress. Research gaps can also be seen in areas such as building domain-specific word embeddings, addressing the need for an ideal corpus. This survey would serve as a good foundation for researchers who intend to find new methods to measure semantic similarity.

APPENDICES

A SEMANTIC DISTANCE MEASURES AND THEIR FORMULAE

Table 4. Table of Semantic Measures and their Formulae (Adapted from Mohammad and Hurst [81])

SNo	Semantic distance measure	Formula
1	α - skew divergence (ASD)	$\displaystyle\sum_{w \in C(w_1) \cup C(w_2)} P(w\|w_1) log \frac{P(w\|w_1)}{\alpha P(w\|w_2) + (1-\alpha) P(w\|w_1)}$
2	Cosine similarity	$\dfrac{\sum_{w \in C(w_1) \cup C(w_2)} P(w \mid w_1) \times P(w \mid w_2)}{\sqrt{\sum_{w \in C(w_1)} P(w \mid w_1)^2} \times \sqrt{\sum_{w \in C(w_2)} P(w\|w_2)^2}}$
3	Co-occurrence Retrieval Models (CRM)	$\gamma \left[\dfrac{2 \times P \times R}{P + R} \right] + (1-\gamma) \left[\beta[P] + (1-\beta)[R] \right]$
4	Dice coefficient	$\dfrac{2 \times \sum_{w \in C(w_1) \cup C(w_2)} \min(P(w\|w_1), P(w\|w_2))}{\sum_{w \in C(w_1)} P(w\|w_1) + \sum_{w \in C(w_2)} P(w\|w_2)}$
5	Manhattan Distance or L1 norm	$\displaystyle\sum_{w \in C(w_1) \cup C(w_2)} \left\| P(w\|w_1) - P(w\|w_2) \right\|$
6	Division measure	$\displaystyle\sum_{w \in C(w_1) \cup C(w_2)} \left\| \log \frac{P(w\|w_1)}{P(w\|w_2)} \right\|$
7	Hindle	$\sum_{w \in C(w)} \begin{cases} \min(I(w, w_1), I(w, w_2)), \\ \qquad \text{if both} I(w, w_1) \text{and} I(w, w_2) > 0 \\ \|\max(I(w, w_1), I(w, w_2))\|, \\ \qquad \text{if both} I(w, w_1) \text{and} I(w, w_2) < 0 \\ 0, \text{otherwise} \end{cases}$
8	Jaccard	$\dfrac{\sum_{w \in C(w_1) \cup C(w_2)} \min(P(w\|w_1), P(w\|w_2))}{\sum_{w \in C(w_1) \cup C(w_2)} \max(P(w\|w_1), P(w\|w_2))}$
9	Jensen-Shannon divergence (JSD)	$\sum_{w \in C(w_1) \cup C(w_2)} \Big(P(w\|w_1) \log \frac{P(w\|w_1)}{\frac{1}{2}(P(w\|w_1) + P(w\|w_2))} + P(w\|w_2) \log \frac{P(w\|w_2)}{\frac{1}{2}(P(w\|w_1) + P(w\|w_2))} \Big)$

(Continued)

Table 4. Continued

SNo	Semantic distance measure	Formula
10	Kullback-Leibler divergence - common occurrence	$\sum_{w \in C(w_1) \cup C(w_2)} P(w\|w_1) \log \frac{P(w\|w_1)}{P(w\|w_2)}$
11	Kullback-Leibler divergence - absolute	$\sum_{w \in C(w_1) \cup C(w_2)} P(w\|w_1) \left\| \log \frac{P(w\|w_1)}{P(w\|w_2)} \right\|$
12	Kullback-Leibler divergence - average	$\frac{1}{2} \sum_{w \in C(w_1) \cup C(w_2)} (P(w\|w_1) - P(w\|w_2)) \log \frac{P(w\|w_1)}{P(w\|w_2)}$
13	Kullback-Leibler divergence - maximum	$\max(KLD(w_1, w_2), KLD(w_2, w_1))$
14	Euclidean Distance or L2 norm	$\sqrt{\sum_{w \in C(w_1) \cup C(w_2)} (P(w\|w_1) - P(w\|w_2))^2}$
15	Lin	$\frac{\sum_{(r,w) \in T(w_1) \cap T(w_2)} (I(w_1, r, w) + I(w_2, r, w))}{\sum (r, w') \in T(w_1) I(w_1, r, w') + \sum_{(r,w'') \in T(w_2)} I(w_2, r, w'')}$
16	Product measure	$\sum_{w \in C(w_1) \cup C(w_2)} \frac{P(w\|w_1) \times P(w\|w_2)}{(\frac{1}{2}(P(w\|w_1) + P(w\|w_2)))^2}$

B TABLE OF REFERENCES

Table 5. Table of References Used in the Analysis of the Survey

Citation	Title	Year	Authors	Venue	SJR Quartile	H-Index	Citations as on 02.04.2020
[1]	A study on similarity and relatedness using distributional and WordNet-based approaches	2009	Agirre, Eneko and Alfonseca, Enrique and Hall, Keith and Kravalova, Jana and Pasca, Marius and Soroa, Aitor	Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics		61	809
[2]	SemEval-2015 task 2: Semantic textual similarity, English, Spanish and pilot on interpretability	2015	Agirre, Eneko and Banea, Carmen and Cardie, Claire and Cer, Daniel and Diab, Mona and Gonzalez-Agirre, Aitor and Guo, Weiwei and Lopez-Gazpio, Inigo and Maritxalar, Montse and Mihalcea, Rada, and others	Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)		49	242
[3]	SemEval-2014 task 10: Multilingual semantic textual similarity	2014	Agirre, Eneko and Banea, Carmen and Cardie, Claire and Cer, Daniel and Diab, Mona and Gonzalez-Agirre, Aitor and Guo, Weiwei and Mihalcea, Rada and Rigau, German and Wiebe, Janyce	Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)		49	220
[4]	SemEval-2016 task 1: Semantic textual similarity, monolingual and cross-lingual evaluation	2016	Agirre, Eneko and Banea, Carmen and Cer, Daniel and Diab, Mona and Gonzalez Agirre, Aitor and Mihalcea, Rada and Rigau Claramunt, German and Wiebe, Janyce	Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016)		49	200
[5]	SemEval-2012 task 6: A pilot on semantic textual similarity	2012	Agirre, Eneko and Cer, Daniel and Diab, Mona and Gonzalez-Agirre, Aitor	Proceedings of the 6th International Workshop on Semantic Evaluation (SemEval 2012)		49	498

(Continued)

Table 5. Continued

Citation	Title	Year	Authors	Venue	SJR Quartile	H-Index	Citations as on 02.04.2020
[6]	SEM 2013 shared task: Semantic textual similarity	2013	Agirre, Eneko and Cer, Daniel and Diab, Mona and Gonzalez-Agirre, Aitor and Guo, Weiwei	Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity		49	268
[7]	A Neural Attention Model for Abstractive Sentence	2015	Alexander M. Rush, Sumit Chopra and Jason Weston	EMNLP		88	1350
[8]	Semantic text classification: A survey of past and recent advances	2018	Altinel, Berna and Ganiz, Murat Can	Information Processing & Management	Q1	88	29
[9]	Sentence similarity based on semantic kernels for intelligent text retrieval	2017	Amir, Samir and Tanasescu, Adrian and Zighed, Djamel A	Journal of Intelligent Information Systems	Q2	52	8
[10]	Neural machine translation by jointly learning to align and translate	2015	Dzmitry Bahdanau and Kyunghyun Cho and Yoshua Bengio	International Conference on Learning Representations		150	10967
[11]	Extended gloss overlaps as a measure of semantic relatedness	2003	Banerjee, Satanjeev and Pedersen, Ted	IJCAI		109	953
[12]	UKP: Computing semantic textual similarity by combining multiple content similarity measures	2012	Bär, Daniel and Biemann, Chris and Gurevych, Iryna and Zesch, Torsten	Proceedings of the 6th International Workshop on Semantic Evaluation (SemEval 2012)		50	227
[13]	The WaCky wide web: a collection of very large linguistically processed web-crawled corpora	2009	Baroni, Marco and Bernardini, Silvia and Ferraresi, Adriano and Zanchetta, Eros	Language resources and evaluation		40	1130
[14]	Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors	2014	Baroni, Marco and Dinu, Georgiana and Kruszewski, GermÃ¡n	Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)		106	1166
[15]	SciBERT: A Pretrained Language Model for Scientific Text	2019	Beltagy, Iz and Lo, Kyle and Cohan, Arman	EMNLP		88	74
[16]	Computing inter-document similarity with Context Semantic Analysis	2019	Fabio Benedetti and Domenico Beneventano and Sonia Bergamaschi and Giovanni Simonini	Information Systems	Q1	76	24
[17]	DBpedia-A crystallization point for the Web of Data	2009	Bizer, Christian and Lehmann, Jens and Kobilarov, Georgi and Auer, Sören and Becker, Christian and Cyganiak, Richard and Hellmann, Sebastian	Journal of Web Semantics		28	2331
[18]	Enriching word vectors with subword information	2017	Bojanowski, Piotr and Grave, Edouard and Joulin, Armand and Mikolov, Tomas	Transactions of the Association for Computational Linguistics		47	2935
[19]	Question Answering with Subgraph Embeddings	2014	Bordes, Antoine and Chopra, Sumit and Weston, Jason	EMNLP		88	433
[20]	From Word to Sense Embeddings: A Survey on Vector Representations of Meaning	2018	Camacho-Collados, Jose and Pilehvar, Mohammad Taher	Journal of Artificial Intelligence Research	Q1	103	69
[21]	Nasari: A novel approach to a semantically aware representation of items	2015	Camacho-Collados, José and Pilehvar, Mohammad Taher and Navigli, Roberto	Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies		61	74
[22]	Nasari: Integrating explicit knowledge and corpus statistics for a multilingual representation of concepts and entities	2016	José Camacho-Collados and Mohammad Taher Pilehvar and Roberto Navigli	Artificial Intelligence	Q1	135	117

(Continued)

Table 5. Continued

Citation	Title	Year	Authors	Venue	SJR Quartile	H-Index	Citations as on 02.04.2020
[23]	Word-sequence kernels	2003	Cancedda, Nicola and Gaussier, Eric and Goutte, Cyril and Renders, Jean-Michel	Journal of Machine Learning Research	Q1	188	291
[24]	SemEval-2017 task 1: Semantic textual similarity-multilingual and cross-lingual focused evaluation	2017	Cer, Daniel and Diab, Mona and Agirre, Eneko and Lopez-Gazpio, Inigo and Specia, Lucia	Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)		49	227
[25]	The Google similarity distance	2007	Cilibrasi, Rudi L. and Vitanyi, Paul MB	IEEE Transactions on Knowledge and Data Engineering	Q1	148	2042
[26]	Convolution kernels for natural language	2002	Collins, Michael and Duffy, Nigel	Advances in Neural Information Processing Systems	Q1	169	1118
[27]	New ranking algorithms for parsing and tagging: Kernels over discrete structures, and the voted perceptron	2002	Collins, Michael and Duffy, Nigel	Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics		135	671
[28]	Deep learning in semantic kernel spaces	2017	Croce, Danilo and Filice, Simone and Castellucci, Giuseppe and Basili, Roberto	Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)		106	15
[29]	BERT: Pretraining of Deep Bidirectional Transformers for Language Understanding	2019	Devlin, Jacob and Chang, Ming-Wei and Lee, Kenton and Toutanova, Kristina	Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)		61	7390
[30]	Placing search in context: The concept revisited	2001	Finkelstein, Lev and Gabrilovich, Evgeniy and Matias, Yossi and Rivlin, Ehud and Solan, Zach and Wolfman, Gadi and Ruppin, Eytan	Proceedings of the 10th International Conference on World Wide Web		70	1768
[31]	Computing semantic relatedness using Wikipedia-based explicit semantic analysis.	2007	Gabrilovich, Evgeniy and Markovitch, Shaul and others	IJCAI		109	2514
[32]	PPDB: The paraphrase database	2013	Ganitkevitch, Juri and Van Durme, Benjamin and Callison-Burch, Chris	Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies		61	493
[33]	A WordNet-based semantic similarity measurement combining edge-counting and information content theory	2015	Jian-Bo Gao and Bao-Wen Zhang and Xiao-Hua Chen	Engineering Applications of Aritifical Intelligence	Q1	86	74
[34]	SimVerb-3500: A Large-scale Evaluation Set of Verb Similarity	2016	Gerz, Daniela and Vulić, Ivan and Hill, Felix and Reichart, Roi and Korhonen, Anna	EMNLP		88	113
[35]	A resource-light method for cross-lingual semantic textual similarity	2018	Goran Glavaš and Marc Franco-Salvador and Simone P. Ponzetto and Paolo Rosso	Knowledge-based Systems	Q1	94	13
[36]	Scaling distributional similarity to large corpora	2006	Gorman, James and Curran, James R.	44th Annual Meeting of the Association for Computational Linguistics		135	54
[37]	A survey of semantic relatedness evaluation datasets and procedures	2019	Hadj Taieb, Mohamed Ali and Zesch, Torsten and Ben Aouicha, Mohamed	Artificial Intelligence Review	Q1	63	–
[38]	UESTS: An Unsupervised Ensemble Semantic Textual Similarity Method	2019	Hassan, Basma and Abdelrahman, Samir E and Bahgat, Reem and Farag, Ibrahim	IEEE Access	Q1	56	1
[39]	Pairwise Word Interaction Modeling with Deep Neural Networks for Semantic Similarity Measurement	2016	He, Hua and Lin, Jimmy	Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies		61	140

(Continued)

Table 5. Continued

Citation	Title	Year	Authors	Venue	SJR Quartile	H-Index	Citations as on 02.04.2020
[40]	Simlex-999: Evaluating semantic models with (genuine) similarity estimation	2015	Hill, Felix and Reichart, Roi and Korhonen, Anna	Computational Linguistics	Q2	85	728
[41]	YAGO2: A spatially and temporally enhanced knowledge base from Wikipedia	2013	Hoffart, Johannes and Suchanek, Fabian M and Berberich, Klaus and Weikum, Gerhard	Artificial Intelligence	Q1	135	1064
[42]	Syntactic, Semantic and Sentiment Analysis: The Joint Effect on Automated Essay Evaluation	2019	Janda, Harneet Kaur and Pawar, Atish and Du, Shan and Mago, Vijay	IEEE Access	Q1	56	–
[43]	Semantic similarity based on corpus statistics and lexical taxonomy	1997	Jiang, Jay J. and Conrath, David W.	COLING		41	3682
[44]	Wikipedia-based information content and semantic similarity computation	2017	Yuncheng Jiang and Wen Bai and Xiaopei Zhang and Jiaojiao Hu	Information Processing & Management	Q1	88	43
[45]	Feature-based approaches to semantic similarity assessment of concepts using Wikipedia	2015	Jiang, Yuncheng and Zhang, Xiaopei and Tang, Yong and Nie, Ruihua	Information Processing & Management	Q1	88	55
[46]	TinyBERT: Distilling BERT for natural language understanding	2019	Jiao, Xiaoqi and Yin, Yichun and Shang, Lifeng and Jiang, Xin and Chen, Xiao and Li, Linlin and Wang, Fang and Liu, Qun	arXiv			56
[47]	Building a monolingual parallel corpus for text simplification using sentence similarity based on alignment between word embeddings	2016	Kajiwara, Tomoyuki and Komachi, Mamoru	COLING		49	39
[48]	Bridging the gap: Incorporating a semantic similarity measure for effectively mapping PubMed queries to documents	2017	Kim, Sun and Fiorini, Nicolas and Wilbur, W John and Lu, Zhiyong	Journal of Biomedical Informatics	Q1	83	14
[49]	Convolutional Neural Networks for Sentence Classification	2014	Kim, Yoon	EMNLP		88	6790
[50]	ALBERT: A Lite BERT for Self-supervised Learning of Language Representations	2019	Lan, Zhenzhong and Chen, Mingda and Goodman, Sebastian and Gimpel, Kevin and Sharma, Piyush and Soricut, Radu	International Conference on Learning Representations		150	270
[51]	A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge.	1997	Landauer, Thomas K. and Dumais, Susan T.	Psychological Review	Q1	192	6963
[52]	An introduction to latent semantic analysis	1998	Landauer, Thomas K. and Foltz, Peter W. and Laham, Darrell	Discourse Processes	Q1	50	5752
[53]	A new family of information content models with an experimental survey on WordNet	2015	Lastra-Díaz, Juan J. and García-Serrano, Ana	Knowledge-based Systems	Q1	94	12
[54]	HESML: A scalable ontology-based semantic similarity measures library with a set of reproducible experiments and a replication dataset	2017	Lastra-Díaz, Juan J. and García-Serrano, Ana and Batet, Montserrat and Fernández, Miriam and Chirigati, Fernando	Information Systems	Q1	76	27
[55]	A reproducible survey on word embeddings and ontology-based methods forword similarity: Linear combinations outperform the state of the art	2019	Juan J. Lastra-Díaz and Josu Goikoetxea and Mohamed Ali Hadj Taieb and Ana García-Serrano and Mohamed Ben Aouicha and Eneko Agirre	Engineering Applications of Aritificial Intelligence	Q1	86	7

(Continued)

Table 5. Continued

Citation	Title	Year	Authors	Venue	SJR Quartile	H-Index	Citations as on 02.04.2020
[56]	Distributed representations of sentences and documents	2014	Le, Quoc and Mikolov, Tomas	International Conference on Machine Learning		135	5406
[57]	ACV-tree: A new method for sentence similarity modeling	2018	Le, Yuquan and Wang, Zhi-Jie and Quan, Zhe and He, Jiawei and Yao, Bin	IJCAI		109	4
[58]	A novel sentence similarity measure for semantic-based expert systems	2011	Lee, Ming Che	Expert Systems with Applications	Q1	162	47
[59]	Dependency-based word embeddings	2014	Levy, Omer and Goldberg, Yoav	Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics		106	860
[60]	Neural word embedding as implicit matrix factorization	2014	Levy, Omer and Goldberg, Yoav	Book			1480
[61]	Computing term similarity by large probabilistic isA knowledge	2013	Li, Peipei and Wang, Haixun and Zhu, Kenny Q. and Wang, Zhongyuan and Wu, Xindong	Proceedings of the 22nd ACM International Conference on Information & Knowledge Management		48	56
[62]	An approach for measuring semantic similarity between words using multiple information sources	2003	Li, Yuhua and Bandar, Zuhair A. and McLean, David	IEEE Transactions on Knowledge and Data Engineering	Q1	148	1315
[63]	Sentence similarity based on semantic nets and corpus statistics	2006	Li, Yuhua and McLean, David and Bandar, Zuhair A. and O'shea, James D. and Crockett, Keeley	IEEE Transactions on Knowledge and Data Engineering	Q1	148	849
[64]	An information-theoretic definition of similarity	1998	Lin	ICML		135	5263
[65]	RoBERTa: A robustly optimized bert pretraining approach	2019	Liu, Yinhan and Ott, Myle and Goyal, Naman and Du, Jingfei and Joshi, Mandar and Chen, Danqi and Levy, Omer and Lewis, Mike and Zettlemoyer, Luke and Stoyanov, Veselin	arXiv			229
[66]	Interpretable semantic textual similarity: Finding and explaining differences between sentences	2017	I. Lopez-Gazpio, M. Maritxalar, A. Gonzalez-Agirre, G. Rigau, L. Uria, E. Agirre,	Knowledge-based Systems	Q1	94	16
[67]	Word n-gram attention models for sentence similarity and inference	2019	I. Lopez-Gazpio and M. Maritxalar and M. Lapata and E. Agirre	Expert Systems with Applications	Q1	162	2
[68]	Producing high-dimensional semantic spaces from lexical co-occurrence	1996	Lund, Kevin and Burgess, Curt	Behavior Research Methods	Q1	114	1869
[69]	A SICK cure for the evaluation of compositional distributional semantic models	2014	Marelli, M. and Menini, S. and Baroni, M. and Bentivogli, L. and Bernardi, R. and Zamparelli, R.	International Conference on Language Resources and Evaluation (LREC)		45	464
[70]	Learned in translation: Contextualized word vectors	2017	McCann, Bryan and Bradbury, James and Xiong, Caiming and Socher, Richard	NIPS		169	376
[71]	UMLS: Similarity: Measuring the relatedness and similarity of biomedical concept	2013	McInnes, Bridget T. and Liu, Ying and Pedersen, Ted and Melton, Genevieve B. and Pakhomov, Serguei V.	Human Language Technologies: The 2013 Annual Conference of the North American Chapter of the Association for Computational Linguistics		61	14
[72]	WIKIQA: A challenge dataset for open-domain question answering	2018	Meek, Wen-tau Yih Christopher	EMNLP		88	351
[73]	context2vec: Learning generic context embedding with bidirectional LSTM	2016	Melamud, Oren and Goldberger, Jacob and Dagan, Ido	Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning		34	198

(Continued)

Table 5. Continued

Citation	Title	Year	Authors	Venue	SJR Quartile	H-Index	Citations as on 02.04.2020
[74]	Wikify! Linking documents to encyclopedic knowledge	2007	Mihalcea, Rada and Csomai, Andras	Proceedings of the 16th ACM Conference on Information and Knowledge Management		48	1120
[75]	Efficient estimation of word representations in vector space	2013	Mikolov, Tomas and Chen, Kai and Corrado, Greg and Dean, Jeffrey	arXiv			14807
[76]	Linguistic regularities in continuous space word representations	2013	Mikolov, Tomas and Yih, Wen-tau and Zweig, Geoffrey	Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies		61	2663
[77]	WordNet: a lexical database for English	1995	Miller, George A.	Communications of the ACM	Q1	189	13223
[78]	Contextual correlates of semantic similarity	1991	Miller, George A. and Charles, Walter G.	Language and Cognitive Processes			1727
[79]	Learning word embeddings efficiently with noise-contrastive estimation	2013	Mnih, Andriy and Kavukcuoglu, Koray	Advances in Neural Information Processing Systems	Q1	169	495
[80]	SRL-ESA-TextSum: A text summarization approach based on semantic role labeling and explicit semantic analysis	2019	Mohamed, Muhidin and Oussalah, Mourad	Information Processing & Management	Q1	88	2
[81]	Distributional measures of semantic distance: A survey	2012	Mohammad, Saif M. and Hirst, Graeme	arXiv			51
[82]	Efficient convolution kernels for dependency and constituent syntactic trees	2006	Moschitti, Alessandro	European Conference on Machine Learning		31	493
[83]	Kernel methods, syntax, and semantics for relational text categorization	2008	Moschitti, Alessandro	Proceedings of the 17th ACM Conference on Information and Knowledge Management		54	105
[84]	Tree kernels for semantic role labeling	2008	Moschitti, Alessandro and Pighin, Daniele and Basili, Roberto	Computational Linguistics	Q1	92	180
[85]	Kernels on linguistic structures for answer extraction	2008	Moschitti, Alessandro and Quarteroni, Silvia	Proceedings of ACL-08: HLT, Short Papers		90	34
[86]	Exploiting syntactic and shallow semantic kernels for question answer classification	2007	Moschitti, Alessandro and Quarteroni, Silvia and Basili, Roberto and Manandhar, Suresh	Annual Meeting of the Association of Computational Linguistics		135	229
[87]	Fast and effective kernels for relational learning from texts	2008	Moschitti, Alessandro and Zanzotto, Fabio Massimo	International Conference on Machine Learning		135	56
[88]	BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network	2012	Navigli, Roberto and Ponzetto, Simone Paolo	Artificial Intelligence	Q1	135	1110
[89]	The University of South Florida free association, rhyme, and word fragment norms	2004	Nelson, Douglas L. and McEvoy, Cathy L. and Schreiber, Thomas A.	Behavior Research Methods, Instruments, & Computers	Q1	114	2162
[90]	Inductive Dependency Parsing	2006	J. Nivre	Book			313
[91]	Unsupervised learning of sentence embeddings using compositional n-gram features	2018	Pagliardini, Matteo and Gupta, Prakhar and Jaggi, Martin	North American Chapter of the Association for Computational Linguistics: Human Language Technologies		61	233
[92]	A decomposable attention model for natural language inference	2016	Parikh, Ankur and Tackstrom, Oscar and Das, Dipanjan and Uszkoreit, Jakob	EMNLP		88	550
[93]	Challenging the boundaries of unsupervised learning for semantic similarity	2019	A. Pawar and V. Mago,	IEEE Access	Q1	56	11

(Continued)

Table 5. Continued

Citation	Title	Year	Authors	Venue	SJR Quartile	H-Index	Citations as on 02.04.2020
[94]	Measures of semantic similarity and relatedness in the biomedical domain	2007	Pedersen, Ted and Pakhomov, Serguei V. S. and Patwardhan, Siddharth and Chute, Christopher G.	Journal of Biomedical Informatics	Q1	83	555
[95]	GloVe: Global vectors for word representation	2014	Pennington, Jeffrey and Socher, Richard and Manning, Christopher D	EMNLP		88	12376
[96]	Deep contextualized word representations	2018	Peters, Matthew E. and Neumann, Mark and Iyyer, Mohit and Gardner, Matt and Clark, Christopher and Lee, Kenton and Zettlemoyer, Luke	Proceedings of NAACL-HLT		61	3842
[97]	WiC: The Word-in-Context dataset for evaluating context-sensitive meaning representations	2019	Pilehvar, Mohammad Taher and Camacho-Collados, Jose	Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)		61	11
[98]	Align, disambiguate and walk: A unified approach for measuring semantic similarity	2013	Pilehvar, Mohammad Taher and Jurgens, David and Navigli, Roberto	Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)		106	184
[99]	From senses to texts: An all-in-one graph-based approach for measuring semantic similarity	2015	Mohammad Taher Pilehvar and Roberto Navigli	Artificial Intelligence	Q1	135	66
[100]	Computing semantic similarity based on novel models of semantic representation using Wikipedia	2018	Rong Qu and Yongyi Fang and Wen Bai and Yuncheng Jiang	Information Processing & Management	Q1	88	11
[101]	An efficient framework for sentence similarity modeling	2019	Z. Quan and Z. Wang and Y. Le and B. Yao and K. Li and J. Yin	IEEE/ACM Transactions on Audio, Speech and Language Processing	Q1	55	4
[102]	Development and application of a metric on semantic nets	1989	Rada, Roy and Mili, Hafedh and Bicknell, Ellen and Blettner, Maria	IEEE Transactions on Systems, Man, and Cybernetics	Q1	111	2347
[103]	Exploring the limits of transfer learning with a unified text-to-text transformer	2020	Raffel, Colin and Shazeer, Noam and Roberts, Adam and Lee, Katherine and Narang, Sharan and Matena, Michael and Zhou, Yanqi and Li, Wei and Liu, Peter J.	arXiv			192
[104]	Using information content to evaluate semantic similarity in a taxonomy	1995	Resnik, Philip	IJCAI		109	4300
[105]	Determining semantic similarity among entity classes from different ontologies	2003	Rodríguez, M. Andrea and Egenhofer, Max J.	EMNLP		88	1183
[106]	Multi-sense embeddings through a word sense disambiguation process	2019	Terry Ruas and William Grosky and Akiko Aizawa	Expert Systems with Applications	Q1	162	4
[107]	Contextual correlates of synonymy	1965	Rubenstein, Herbert and Goodenough, John	Communications of the ACM	Q1	189	1336
[108]	Ontology-based information content computation	2011	Sánchez, David and Batet, Montserrat and Isern, David	Knowledge-based Systems	Q1	94	251
[109]	DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter	2019	Sanh, Victor and Debut, Lysandre and Chaumond, Julien and Wolf, Thomas	arXiv			112
[110]	Takelab: Systems for measuring semantic text similarity	2012	Šarić, Frane and Glavaš, Goran and Karan, Mladen and Šnajder, Jan and Bašić, Bojana Dalbelo	Proceedings of the 6th International Workshop on Semantic Evaluation (SemEval 2012)		49	224
[111]	Evaluation methods for unsupervised word embeddings	2015	Schnabel, Tobias and Labutov, Igor and Mimno, David and Joachims, Thorsten	EMNLP		88	334

(Continued)

Table 5. Continued

Citation	Title	Year	Authors	Venue	SJR Quartile	H-Index	Citations as on 02.04.2020
[112]	Structural relationships for large-scale learning of answer re-ranking	2012	Severyn, Aliaksei and Moschitti, Alessandro	ACM SIGIR Conference on Research and Development in Information Retrieval		57	85
[113]	Learning semantic textual similarity with structural representations	2013	Severyn, Aliaksei and Nicosia, Massimo and Moschitti, Alessandro	Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)		135	40
[114]	HCTI at SemEval-2017 Task 1: Use Convolutional Neural Network to evaluate semantic textual similarity	2017	Shao, Yang	Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)		49	32
[115]	Kernel methods for pattern analysis	2004	Shawe-Taylor, John and Cristianini, Nello and others	Book			7721
[116]	Learning grounded meaning representations with autoencoders	2014	Silberer, Carina and Lapata, Mirella	Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)		61	127
[117]	Knowledge-enhanced document embeddings for text classification	2018	Roberta A. Sinoara and Jose Camacho-Collados and Rafael G. Rossi and Roberto Navigli and Solange O. Rezende	Knowledge-based Systems	Q1	94	25
[118]	BIOSSES: a semantic sentence similarity estimation system for the biomedical domain	2017	Soğancıoğlu, Gizem and Öztürk, Hakime and Özgür, Arzucan	Bioinformatics	Q1	335	34
[119]	DLS@ CU: sentence similarity from word alignment	2014	Sultan, Md Arafat and Bethard, Steven and Sumner, Tamara	Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)		49	112
[120]	Dls@ cu: Sentence similarity from word alignment and semantic vector composition	2015	Sultan, Md Arafat and Bethard, Steven and Sumner, Tamara	Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)		49	105
[121]	ERNIE 2.0: A Continual Pretraining framework for language understanding.	2020	Sun, Yu and Wang, Shuohuan and Li, Yu-Kun and Feng, Shikun and Tian, Hao and Wu, Hua and Wang, Haifeng	AAAI		95	53
[122]	A semantic similarity method based on information content exploiting multiple ontologies	2013	David Sánchez and Montserrat Batet	Expert Systems with Applications	Q1	162	82
[123]	Ontology-based semantic similarity: A new feature-based approach	2012	David Sánchez and Montserrat Batet and David Isern and Aida Valls	Expert Systems with Applications	Q1	162	361
[124]	Improved semantic representations from tree-structured long short-term memory networks	2015	Tai, Kai Sheng and Socher, Richard and Manning, Christopher D.	Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)		106	1676
[125]	Ecnu at SemEval-2017 task 1: Leverage kernel-based traditional nlp features and neural networks to build a universal model for multilingual and cross-lingual semantic textual similarity	2017	Tian, Junfeng and Zhou, Zhiheng and Lan, Man and Wu, Yuanbin	Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)		49	34
[126]	Sentence modeling via multiple word embeddings and multi-level comparison for semantic textual similarity	2019	Nguyen Huy Tien and Nguyen Minh Le and Yamasaki Tomohiro and Izuha Tatsuya	Information Processing & Management	Q1	88	7
[127]	Dict2vec: Learning word embeddings using lexical dictionaries	2017	Tissier, Julien and Gravier, Christophe and Habrard, Amaury	EMNLP		112	51

(Continued)

Table 5. Continued

Citation	Title	Year	Authors	Venue	SJR Quartile	H-Index	Citations as on 02.04.2020
[128]	Attention is all you need	2017	Vaswani, Ashish and Shazeer, Noam and Parmar, Niki and Uszkoreit, Jakob and Jones, Llion and Gomez, Aidan N. and Kaiser, Lukasz and Polosukhin, Illia	NIPS		169	9994
[129]	What is the Jeopardy model? A quasi-synchronous grammar for QA	2007	Wang, Mengqiu and Smith, Noah A. and Mitamura, Teruko	EMNLP		88	337
[130]	Sentence similarity learning by lexical decomposition and composition	2016	Wang, Zhiguo and Mi, Haitao and Ittycheriah, Abraham	COLING		41	119
[131]	Verbs semantics and lexical selection	1994	Wu, Zhibiao and Palmer, Martha	Proceedings of the 32nd Annual Meeting on Association for Computational Linguistics		106	3895
[132]	XlNet: Generalized autoregressive pretraining for language understanding	2019	Yang, Zhilin and Dai, Zihang and Yang, Yiming and Carbonell, Jaime and Salakhutdinov, Russ R. and Le, Quoc V.	Advances in Neural Information Processing Systems	Q1	169	865
[133]	Computing semantic similarity of concepts in knowledge graphs	2017	G. Zhu and C. A. Iglesias	IEEE Transactions on Knowledge and Data Engineering	Q1	148	88
[134]	Bilingual word embeddings for phrase-based machine translation	2013	Zou, Will Y. and Socher, Richard and Cer, Daniel and Manning, Christopher D.	EMNLP		88	468

ACKNOWLEDGMENTS

The authors would like to extend our gratitude to the research team in the DaTALab at Lakehead University for their support; in particular, Abhijit Rao, Mohiuddin Qudar, Punardeep Sikka, and Andrew Heppner for their feedback and revisions on this publication. We would also like to thank Lakehead University, CASES, and the Ontario Council for Articulation and Transfer (ONCAT), without their support this research would not have been possible.

REFERENCES

[1] Eneko Agirre, Enrique Alfonseca, Keith Hall, Jana Kravalova, Marius Pasca, and Aitor Soroa. 2009. A study on similarity and relatedness using distributional and WordNet-based approaches. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Citeseer, 19.

[2] Eneko Agirre, Carmen Banea, Claire Cardie, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Weiwei Guo, Inigo Lopez-Gazpio, Montse Maritxalar, Rada Mihalcea, et al. 2015. Semeval-2015 task 2: Semantic textual similarity, English, Spanish, and pilot on interpretability. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval’15)*. 252–263.

[3] Eneko Agirre, Carmen Banea, Claire Cardie, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Weiwei Guo, Rada Mihalcea, German Rigau, and Janyce Wiebe. 2014. Semeval-2014 task 10: Multilingual semantic textual similarity. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval’14)*. 81–91.

[4] Eneko Agirre, Carmen Banea, Daniel Cer, Mona Diab, Aitor Gonzalez Agirre, Rada Mihalcea, German Rigau Claramunt, and Janyce Wiebe. 2016. Semeval-2016 task 1: Semantic textual similarity, monolingual and cross-lingual evaluation. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval’16)*. ACL (Association for Computational Linguistics), 497–511.

[5] Eneko Agirre, Daniel Cer, Mona Diab, and Aitor Gonzalez-Agirre. 2012. Semeval-2012 task 6: A pilot on semantic textual similarity. In * *SEM 2012: The First Joint Conference on Lexical and Computational Semantics–Volume 1: Proceedings of the Main Conference and the Shared Task, and Volume 2: Proceedings of the 6th International Workshop on Semantic Evaluation (SemEval’12)*. 385–393.

[6] Eneko Agirre, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, and Weiwei Guo. 2013. * SEM 2013 shared task: Semantic textual similarity. In *Proceedings of the 2nd Joint Conference on Lexical and Computational Semantics (* SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity.* 32–43.

[7] Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing.* 5, 3 (2015), 379–389.

[8] Berna Altinel and Murat Can Ganiz. 2018. Semantic text classification: A survey of past and recent advances. *Inf. Proc. Manag.* 54, 6 (2018), 1129–1153. DOI: https://doi.org/10.1016/j.ipm.2018.08.001

[9] Samir Amir, Adrian Tanasescu, and Djamel A. Zighed. 2017. Sentence similarity based on semantic kernels for intelligent text retrieval. *J. Intell. Inf. Syst.* 48, 3 (2017), 675–689.

[10] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR'15).*

[11] Satanjeev Banerjee and Ted Pedersen. 2003. Extended gloss overlaps as a measure of semantic relatedness. In *Proceedings of the International Joint Conference on Artificial Intelligence*, Vol. 3. 805–810.

[12] Daniel Bär, Chris Biemann, Iryna Gurevych, and Torsten Zesch. 2012. UKP: Computing semantic textual similarity by combining multiple content similarity measures. In ** SEM 2012: The First Joint Conference on Lexical and Computational Semantics–Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the 6th International Workshop on Semantic Evaluation (SemEval'12).* 435–440.

[13] Marco Baroni, Silvia Bernardini, Adriano Ferraresi, and Eros Zanchetta. 2009. The wacky wide web: A collection of very large linguistically processed web-crawled corpora. *Lang. Res. Eval.* 43, 3 (2009), 209–226.

[14] Marco Baroni, Georgiana Dinu, and Germán Kruszewski. 2014. Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proceedings of the 52nd Meeting of the Association for Computational Linguistics (Volume 1: Long Papers).* 238–247.

[15] Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP'19).* 3606–3611.

[16] Fabio Benedetti, Domenico Beneventano, Sonia Bergamaschi, and Giovanni Simonini. 2019. Computing inter-document similarity with context semantic analysis. *Inf. Syst.* 80 (2019), 136–147. DOI: https://doi.org/10.1016/j.is.2018.02.009

[17] Christian Bizer, Jens Lehmann, Georgi Kobilarov, Sören Auer, Christian Becker, Richard Cyganiak, and Sebastian Hellmann. 2009. DBpedia-a crystallization point for the web of data. *J. Web Seman.* 7, 3 (2009), 154–165.

[18] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Trans. Assoc. Comput. Ling.* 5 (2017), 135–146.

[19] Antoine Bordes, Sumit Chopra, and Jason Weston. 2014. Question answering with subgraph embeddings. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP'14).* 615–620.

[20] Jose Camacho-Collados and Mohammad Taher Pilehvar. 2018. From word to sense embeddings: A survey on vector representations of meaning. *J. Artif. Intell. Res.* 63 (2018), 743–788.

[21] José Camacho-Collados, Mohammad Taher Pilehvar, and Roberto Navigli. 2015. Nasari: A novel approach to a semantically aware representation of items. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies.* 567–577.

[22] José Camacho-Collados, Mohammad Taher Pilehvar, and Roberto Navigli. 2016. Nasari: Integrating explicit knowledge and corpus statistics for a multilingual representation of concepts and entities. *Artif. Intell.* 240 (2016), 36–64. DOI: https://doi.org/10.1016/j.artint.2016.07.005

[23] Nicola Cancedda, Eric Gaussier, Cyril Goutte, and Jean-Michel Renders. 2003. Word-sequence kernels. *J. Mach. Learn. Res.* 3, Feb. (2003), 1059–1082.

[24] Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. SemEval-2017 Task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval'17).* 1–14.

[25] Rudi L. Cilibrasi and Paul M. B. Vitanyi. 2007. The Google similarity distance. *IEEE Trans. Knowl. Data Eng.* 19, 3 (2007), 370–383.

[26] Michael Collins and Nigel Duffy. 2002. Convolution kernels for natural language. In *Proceedings of the International Conference on Advances in Neural Information Processing Systems.* 625–632.

[27] Michael Collins and Nigel Duffy. 2002. New ranking algorithms for parsing and tagging: Kernels over discrete structures, and the voted perceptron. In *Proceedings of the 40th Meeting of the Association for Computational Linguistics.* 263–270.

[28] Danilo Croce, Simone Filice, Giuseppe Castellucci, and Roberto Basili. 2017. Deep learning in semantic kernel spaces. In *Proceedings of the 55th Meeting of the Association for Computational Linguistics (Volume 1: Long Papers).* 345–354.

[29] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. 4171–4186.

[30] Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2001. Placing search in context: The concept revisited. In *Proceedings of the 10th International Conference on World Wide Web*. 406–414.

[31] Evgeniy Gabrilovich, Shaul Markovitch, et al. 2007. Computing semantic relatedness using Wikipedia-based explicit semantic analysis. In *Proceedings of the International Joint Conference on Artificial Intelligence*, Vol. 7. 1606–1611.

[32] Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2013. PPDB: The paraphrase database. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 758–764.

[33] Jian-Bo Gao, Bao-Wen Zhang, and Xiao-Hua Chen. 2015. A WordNet-based semantic similarity measurement combining edge-counting and information content theory. *Eng. Applic. Artif. Intell.* 39 (2015), 80–88. DOI: https://doi.org/10.1016/j.engappai.2014.11.009

[34] Daniela Gerz, Ivan Vulić, Felix Hill, Roi Reichart, and Anna Korhonen. 2016. SimVerb-3500: A large-scale evaluation set of verb similarity. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. 2173–2182.

[35] Goran Glavaš, Marc Franco-Salvador, Simone P. Ponzetto, and Paolo Rosso. 2018. A resource-light method for cross-lingual semantic textual similarity. *Knowl.-based Syst.* 143 (2018), 1–9. DOI: https://doi.org/10.1016/j.knosys.2017.11.041

[36] James Gorman and James R. Curran. 2006. Scaling distributional similarity to large corpora. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Meeting of the Association for Computational Linguistics*. 361–368.

[37] Mohamed Ali Hadj Taieb, Torsten Zesch, and Mohamed Ben Aouicha. 2019. A survey of semantic relatedness evaluation datasets and procedures. *Artif. Intell. Rev.* (23 Dec. 2019). DOI: https://doi.org/10.1007/s10462-019-09796-3

[38] Basma Hassan, Samir E. Abdelrahman, Reem Bahgat, and Ibrahim Farag. 2019. UESTS: An unsupervised ensemble semantic textual similarity method. *IEEE Access* 7 (2019), 85462–85482.

[39] Hua He and Jimmy Lin. 2016. Pairwise word interaction modeling with deep neural networks for semantic similarity measurement. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, 937–948. DOI: https://doi.org/10.18653/v1/N16-1108

[40] Felix Hill, Roi Reichart, and Anna Korhonen. 2015. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Comput. Ling.* 41, 4 (2015), 665–695.

[41] Johannes Hoffart, Fabian M. Suchanek, Klaus Berberich, and Gerhard Weikum. 2013. YAGO2: A spatially and temporally enhanced knowledge base from Wikipedia. *Artif. Intell.* 194 (2013), 28–61.

[42] Harneet Kaur Janda, Atish Pawar, Shan Du, and Vijay Mago. 2019. Syntactic, semantic and sentiment analysis: The joint effect on automated essay evaluation. *IEEE Access* 7 (2019), 108486–108503.

[43] Jay J. Jiang and David W. Conrath. 1997. Semantic similarity based on corpus statistics and lexical taxonomy. In *Proceedings of the 10th International Conference on Research on Computational Linguistics*. 19–33.

[44] Yuncheng Jiang, Wen Bai, Xiaopei Zhang, and Jiaojiao Hu. 2017. Wikipedia-based information content and semantic similarity computation. *Inf. Proc. Manag.* 53, 1 (2017), 248–265. DOI: https://doi.org/10.1016/j.ipm.2016.09.001

[45] Yuncheng Jiang, Xiaopei Zhang, Yong Tang, and Ruihua Nie. 2015. Feature-based approaches to semantic similarity assessment of concepts using Wikipedia. *Inf. Proc. Manag.* 51, 3 (2015), 215–234.

[46] Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. 2019. TinyBERT: Distilling BERT for natural language understanding. *Arxiv Preprint Arxiv:1909.10351* (2019).

[47] Tomoyuki Kajiwara and Mamoru Komachi. 2016. Building a monolingual parallel corpus for text simplification using sentence similarity based on alignment between word embeddings. In *Proceedings of the 26th International Conference on Computational Linguistics: Technical Papers (COLING'16)*. 1147–1158.

[48] Sun Kim, Nicolas Fiorini, W. John Wilbur, and Zhiyong Lu. 2017. Bridging the gap: Incorporating a semantic similarity measure for effectively mapping PubMed queries to documents. *J. Biomed. Inf.* 75 (2017), 122–127.

[49] Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP'14)*. 1746–1751.

[50] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. ALBERT: A lite BERT for self-supervised learning of language representations. In *Proceedings of the International Conference on Learning Representations*.

[51] Thomas K. Landauer and Susan T. Dumais. 1997. A solution to Plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychol. Rev.* 104, 2 (1997), 211.

[52] Thomas K. Landauer, Peter W. Foltz, and Darrell Laham. 1998. An introduction to latent semantic analysis. *Discour. Proc.* 25, 2–3 (1998), 259–284.

[53] Juan J. Lastra-Díaz and Ana García-Serrano. 2015. A new family of information content models with an experimental survey on WordNet. *Knowl.-based Syst.* 89 (2015), 509–526.

[54] Juan J. Lastra-Díaz, Ana García-Serrano, Montserrat Batet, Miriam Fernández, and Fernando Chirigati. 2017. HESML: A scalable ontology-based semantic similarity measures library with a set of reproducible experiments and a replication dataset. *Inf. Syst.* 66 (2017), 97–118.

[55] Juan J. Lastra-Díaz, Josu Goikoetxea, Mohamed Ali Hadj Taieb, Ana García-Serrano, Mohamed Ben Aouicha, and Eneko Agirre. 2019. A reproducible survey on word embeddings and ontology-based methods for word similarity: Linear combinations outperform the state of the art. *Eng. Applic. Artif. Intell.* 85 (2019), 645–665. DOI: https://doi.org/10.1016/j.engappai.2019.07.010

[56] Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *Proceedings of the International Conference on Machine Learning*. 1188–1196.

[57] Yuquan Le, Zhi-Jie Wang, Zhe Quan, Jiawei He, and Bin Yao. 2018. ACV-tree: A new method for sentence similarity modeling. In *Proceedings of the International Joint Conference on Artificial Intelligence*. 4137–4143.

[58] Ming Che Lee. 2011. A novel sentence similarity measure for semantic-based expert systems. *Exp. Syst. Applic.* 38, 5 (2011), 6392–6399.

[59] Omer Levy and Yoav Goldberg. 2014. Dependency-based word embeddings. In *Proceedings of the 52nd Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. 302–308.

[60] Omer Levy and Yoav Goldberg. 2014. Neural word embedding as implicit matrix factorization. In *Proceedings of the International Conference on Advances in Neural Information Processing Systems*. 2177–2185.

[61] Peipei Li, Haixun Wang, Kenny Q. Zhu, Zhongyuan Wang, and Xindong Wu. 2013. Computing term similarity by large probabilistic isA knowledge. In *Proceedings of the 22nd ACM International Conference on Information & Knowledge Management*. 1401–1410.

[62] Yuhua Li, Zuhair A. Bandar, and David McLean. 2003. An approach for measuring semantic similarity between words using multiple information sources. *IEEE Trans. Knowl. Data Eng.* 15, 4 (2003), 871–882.

[63] Yuhua Li, David McLean, Zuhair A. Bandar, James D. O'Shea, and Keeley Crockett. 2006. Sentence similarity based on semantic nets and corpus statistics. *IEEE Trans. Knowl. Data Eng.* 18, 8 (2006), 1138–1150.

[64] Dekang Lin et al. 1998. An information-theoretic definition of similarity. In *Proceedings of the International Conference on Machine Learning (ICML'98)*. 296–304.

[65] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *Arxiv Preprint Arxiv:1907.11692* (2019).

[66] I. Lopez-Gazpio, M. Maritxalar, A. Gonzalez-Agirre, G. Rigau, L. Uria, and E. Agirre. 2017. Interpretable semantic textual similarity: Finding and explaining differences between sentences. *Knowl.-based Syst.* 119 (2017), 186–199. DOI: https://doi.org/10.1016/j.knosys.2016.12.013

[67] I. Lopez-Gazpio, M. Maritxalar, M. Lapata, and E. Agirre. 2019. Word n-gram attention models for sentence similarity and inference. *Exp. Syst. Applic.* 132 (2019), 1–11. DOI: https://doi.org/10.1016/j.eswa.2019.04.054

[68] Kevin Lund and Curt Burgess. 1996. Producing high-dimensional semantic spaces from lexical co-occurrence. *Behav. Res. Meth. Instrum. Comput.* 28, 2 (1996), 203–208.

[69] M. Marelli, S. Menini, M. Baroni, L. Bentivogli, R. Bernardi, and R. Zamparelli. 2014. A SICK cure for the evaluation of compositional distributional semantic models. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*. European Language Resources Association (ELRA), Reykjavik, Iceland, 216–223. http://www.lrec-conf.org/proceedings/lrec2014/pdf/363_Paper.pdf.

[70] Bryan McCann, James Bradbury, Caiming Xiong, and Richard Socher. 2017. Learned in translation: Contextualized word vectors. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*. Curran & Associates Inc., 6297–6308.

[71] Bridget T. McInnes, Ying Liu, Ted Pedersen, Genevieve B. Melton, and Serguei V. Pakhomov. 2013. UMLS: Similarity: Measuring the relatedness and similarity of biomedical concepts. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 28.

[72] Christopher Meek, Yang Yi, and Yih Wen-tau. 2018. WIKIQA: A challenge dataset for open-domain question answering. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing September 2015*. 2013–2018. https://doi.org/10.18653/v1/D15-1237

[73] Oren Melamud, Jacob Goldberger, and Ido Dagan. 2016. context2vec: Learning generic context embedding with bidirectional LSTM. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*. 51–61.

[74] Rada Mihalcea and Andras Csomai. 2007. Wikify! Linking documents to encyclopedic knowledge. In *Proceedings of the 16th ACM Conference on Information and Knowledge Management.* 233–242.

[75] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *Arxiv Preprint Arxiv:1301.3781* (2013).

[76] Tomáš Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013. Linguistic regularities in continuous space word representations. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies.* 746–751.

[77] George A. Miller. 1995. WordNet: A lexical database for English. *Commun. ACM* 38, 11 (1995), 39–41.

[78] George A. Miller and Walter G. Charles. 1991. Contextual correlates of semantic similarity. *Lang. Cog. Proc.* 6, 1 (1991), 1–28.

[79] Andriy Mnih and Koray Kavukcuoglu. 2013. Learning word embeddings efficiently with noise-contrastive estimation. In *Proceedings of the International Conference on Advances in Neural Information Processing Systems.* 2265–2273.

[80] Muhidin Mohamed and Mourad Oussalah. 2019. SRL-ESA-TextSum: A text summarization approach based on semantic role labeling and explicit semantic analysis. *Inf. Proc. Manag.* 56, 4 (2019), 1356–1372.

[81] Saif M. Mohammad and Graeme Hirst. 2012. Distributional measures of semantic distance: A survey. *Arxiv Preprint Arxiv:1203.1858* (2012).

[82] Alessandro Moschitti. 2006. Efficient convolution kernels for dependency and constituent syntactic trees. In *Proceedings of the European Conference on Machine Learning*. Springer, 318–329.

[83] Alessandro Moschitti. 2008. Kernel methods, syntax and semantics for relational text categorization. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management.* 253–262.

[84] Alessandro Moschitti, Daniele Pighin, and Roberto Basili. 2008. Tree kernels for semantic role labeling. *Comput. Ling.* 34, 2 (2008), 193–224.

[85] Alessandro Moschitti and Silvia Quarteroni. 2008. Kernels on linguistic structures for answer extraction. In *Proceedings of the Conference of the Association for Computational Linguistics: Human Language Technologies, Short Papers.* 113–116.

[86] Alessandro Moschitti, Silvia Quarteroni, Roberto Basili, and Suresh Manandhar. 2007. Exploiting syntactic and shallow semantic kernels for question answer classification. In *Proceedings of the 45th Meeting of the Association of Computational Linguistics.* 776–783.

[87] Alessandro Moschitti and Fabio Massimo Zanzotto. 2007. Fast and effective kernels for relational learning from texts. In *Proceedings of the 24th International Conference on Machine Learning*. 649–656.

[88] Roberto Navigli and Simone Paolo Ponzetto. 2012. BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artif. Intell.* 193 (2012).

[89] Douglas L. Nelson, Cathy L. McEvoy, and Thomas A. Schreiber. 2004. The University of South Florida free association, rhyme, and word fragment norms. *Behav. Res. Meth. Instrum. Comput.* 36, 3 (2004), 402–407.

[90] Joakim Nivre. 2006. *Inductive Dependency Parsing*. Springer.

[91] Matteo Pagliardini, Prakhar Gupta, and Martin Jaggi. 2018. Unsupervised learning of sentence embeddings using compositional n-gram features. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers).* 528–540.

[92] Ankur Parikh, Oscar Täckström, Dipanjan Das, and Jakob Uszkoreit. 2016. A decomposable attention model for natural language inference. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing.* 2249–2255.

[93] A. Pawar and V. Mago. 2019. Challenging the boundaries of unsupervised learning for semantic similarity. *IEEE Access* 7 (2019), 16291–16308.

[94] Ted Pedersen, Serguei V. S. Pakhomov, Siddharth Patwardhan, and Christopher G. Chute. 2007. Measures of semantic similarity and relatedness in the biomedical domain. *J. Biomed. Inf.* 40, 3 (2007), 288–299.

[95] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP'14).* 1532–1543.

[96] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies.* 2227–2237.

[97] Mohammad Taher Pilehvar and Jose Camacho-Collados. 2019. WiC: the word-in-context dataset for evaluating context-sensitive meaning representations. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers).* 1267–1273.

[98] Mohammad Taher Pilehvar, David Jurgens, and Roberto Navigli. 2013. Align, disambiguate and walk: A unified approach for measuring semantic similarity. In *Proceedings of the 51st Meeting of the Association for Computational Linguistics (Volume 1: Long Papers).* 1341–1351.

[99] Mohammad Taher Pilehvar and Roberto Navigli. 2015. From senses to texts: An all-in-one graph-based approach for measuring semantic similarity. *Artif. Intell.* 228 (2015), 95–128. DOI: https://doi.org/10.1016/j.artint.2015.07.005.

[100] Rong Qu, Yongyi Fang, Wen Bai, and Yuncheng Jiang. 2018. Computing semantic similarity based on novel models of semantic representation using Wikipedia. *Inf. Proc. Manag.* 54, 6 (2018), 1002–1021. DOI: https://doi.org/10.1016/j.ipm.2018.07.002.

[101] Z. Quan, Z. Wang, Y. Le, B. Yao, K. Li, and J. Yin. 2019. An efficient framework for sentence similarity modeling. *IEEE/ACM Trans. Aud. Speech Lang. Proc.* 27, 4 (Apr. 2019), 853–865. DOI: https://doi.org/10.1109/TASLP.2019.2899494.

[102] Roy Rada, Hafedh Mili, Ellen Bicknell, and Maria Blettner. 1989. Development and application of a metric on semantic nets. *IEEE Trans. Syst. Man Cyber.* 19, 1 (1989), 17–30.

[103] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *Arxiv Preprint Arxiv:1910.10683* (2019).

[104] Philip Resnik. 1995. Using information content to evaluate semantic similarity in a taxonomy. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence.* 448–453.

[105] M. Andrea Rodríguez and Max J. Egenhofer. 2003. Determining semantic similarity among entity classes from different ontologies. *IEEE Trans. Knowl. Data Eng.* 15, 2 (2003), 442–456.

[106] Terry Ruas, William Grosky, and Akiko Aizawa. 2019. Multi-sense embeddings through a word sense disambiguation process. *Exp. Syst. Applic.* 136 (2019), 288–303. DOI: https://doi.org/10.1016/j.eswa.2019.06.026

[107] Herbert Rubenstein and John B. Goodenough. 1965. Contextual correlates of synonymy. *Commun. ACM* 8, 10 (1965), 627–633.

[108] David Sánchez, Montserrat Batet, and David Isern. 2011. Ontology-based information content computation. *Knowl.-based Syst.* 24, 2 (2011), 297–303.

[109] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter. *Arxiv Preprint Arxiv:1910.01108* (2019).

[110] Frane Šarić, Goran Glavaš, Mladen Karan, Jan Šnajder, and Bojana Dalbelo Bašić. 2012. Takelab: Systems for measuring semantic text similarity. In ** SEM 2012: The 1st Joint Conference on Lexical and Computational Semantics–Volume 1: Proceedings of the Main Conference and the Shared Task, and Volume 2: Proceedings of the 6th International Workshop on Semantic Evaluation (SemEval'12).* 441–448.

[111] Tobias Schnabel, Igor Labutov, David Mimno, and Thorsten Joachims. 2015. Evaluation methods for unsupervised word embeddings. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing.* 298–307.

[112] Aliaksei Severyn and Alessandro Moschitti. 2012. Structural relationships for large-scale learning of answer re-ranking. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval.* 741–750.

[113] Aliaksei Severyn, Massimo Nicosia, and Alessandro Moschitti. 2013. Learning semantic textual similarity with structural representations. In *Proceedings of the 51st Meeting of the Association for Computational Linguistics (Volume 2: Short Papers).* 714–718.

[114] Yang Shao. 2017. HCTI at SemEval-2017 task 1: Use convolutional neural network to evaluate semantic textual similarity. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval'17).* 130–133.

[115] John Shawe-Taylor, Nello Cristianini et al. 2004. *Kernel Methods for Pattern Analysis.* Cambridge University Press.

[116] Carina Silberer and Mirella Lapata. 2014. Learning grounded meaning representations with autoencoders. In *Proceedings of the 52nd Meeting of the Association for Computational Linguistics (Volume 1: Long Papers).* 721–732.

[117] Roberta A. Sinoara, Jose Camacho-Collados, Rafael G. Rossi, Roberto Navigli, and Solange O. Rezende. 2019. Knowledge-enhanced document embeddings for text classification. *Knowl.-based Syst.* 163 (2019), 955–971. DOI: https://doi.org/10.1016/j.knosys.2018.10.026

[118] Gizem Soğancıoğlu, Hakime Öztürk, and Arzucan Özgür. 2017. BIOSSES: A semantic sentence similarity estimation system for the biomedical domain. *Bioinformatics* 33, 14 (07 2017), i49–i58. arXiv: Retrieved from https://academic.oup.com/bioinformatics/article-pdf/33/14/i49/2515 7316/btx238.pdf.

[119] Md Arafat Sultan, Steven Bethard, and Tamara Sumner. 2014. DLS@ CU: Sentence similarity from word alignment. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval'14).* 241–246.

[120] Md Arafat Sultan, Steven Bethard, and Tamara Sumner. 2015. DLS@ CU: Sentence similarity from word alignment and semantic vector composition. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval'15).* 148–153.

[121] Yu Sun, Shuohuan Wang, Yu-Kun Li, Shikun Feng, Hao Tian, Hua Wu, and Haifeng Wang. 2020. ERNIE 2.0: A continual pre-training framework for language understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence.* 8968–8975.

[122] David Sánchez and Montserrat Batet. 2013. A semantic similarity method based on information content exploiting multiple ontologies. *Exp. Syst. Applic.* 40, 4 (2013), 1393–1399. DOI: https://doi.org/10.1016/j.eswa.2012.08.049

[123] David Sánchez, Montserrat Batet, David Isern, and Aida Valls. 2012. Ontology-based semantic similarity: A new feature-based approach. *Exp. Syst. Applic.* 39, 9 (2012), 7718–7728. DOI: https://doi.org/10.1016/j.eswa.2012.01.082

[124] Kai Sheng Tai, Richard Socher, and Christopher D. Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. In *Proceedings of the 53rd Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. 1556–1566.

[125] Junfeng Tian, Zhiheng Zhou, Man Lan, and Yuanbin Wu. 2017. ECNU at SemEval-2017 task 1: Leverage kernel-based traditional NLP features and neural networks to build a universal model for multilingual and cross-lingual semantic textual similarity. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval'17)*. 191–197.

[126] Nguyen Huy Tien, Nguyen Minh Le, Yamasaki Tomohiro, and Izuha Tatsuya. 2019. Sentence modeling via multiple word embeddings and multi-level comparison for semantic textual similarity. *Inf. Proc. Manag.* 56, 6 (2019), 102090. DOI: https://doi.org/10.1016/j.ipm.2019.102090

[127] Julien Tissier, Christophe Gravier, and Amaury Habrard. 2017. Dict2vec: Learning Word embeddings using lexical dictionaries. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP'17)*. 254–263.

[128] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the International Conference on Advances in Neural Information Processing Systems.*

[129] Mengqiu Wang, Noah A. Smith, and Teruko Mitamura. 2007. What is the jeopardy model? A quasi-synchronous grammar for QA. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL'07)*. 22–32.

[130] Zhiguo Wang, Haitao Mi, and Abraham Ittycheriah. 2016. Sentence similarity learning by lexical decomposition and composition. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING'16)*. 1340–1349. arXiv:1602.07019.

[131] Zhibiao Wu and Martha Palmer. 1994. Verbs semantics and lexical selection. In *Proceedings of the 32nd Meeting on Association for Computational Linguistics.* Association for Computational Linguistics, 133–138.

[132] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R. Salakhutdinov, and Quoc V. Le. 2019. XLNet: Generalized autoregressive pretraining for language understanding. In *Proceedings of the International Conference on Advances in Neural Information Processing Systems.* 5753–5763.

[133] G. Zhu and C. A. Iglesias. 2017. Computing semantic similarity of concepts in knowledge graphs. *IEEE Trans. Knowl. Data Eng.* 29, 1 (Jan. 2017), 72–85. DOI: https://doi.org/10.1109/TKDE.2016.2610428

[134] Will Y. Zou, Richard Socher, Daniel Cer, and Christopher D. Manning. 2013. Bilingual word embeddings for phrase-based machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing.* 1393–1398.

Received April 2020; revised September 2020; accepted November 2020