IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received February 1, 2022, accepted February 28, 2022, date of publication March 8, 2022, date of current version March 18, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3158011

Adjective Phrases in PNL and its Application to Reverse Dictionary

BUSHRA SIDDIQUE AND M. M. SUFYAN BEG, (Senior Member, IEEE)
Department of Computer Engineering, Aligarh Muslim University, Aligarh 202002, India
Corresponding author: Bushra Siddique (bushra.siddique@zhcet.ac.in)

ABSTRACT Owing to the large information content of adjectives in natural language-based data, we have dealt with the subject of adjective phrases in the paradigm of PNL. We have proposed a GCR form for the same and demonstrated its application in building a specific instance of Reverse Dictionary. A Reverse Dictionary takes a natural language description of a concept in the user's mind as input and generates semantically appropriate word/s as output. For example, it could accept descriptions like 'to run slowly,' 'to run but not very fast,' etc., for the word 'jog'. As Reverse Dictionary deals with natural language, handling perception-based information is important in its context for capturing the user intent. The D-RD, Reverse Dictionary for Descriptive words, designed in this paper, takes a natural language description of a human as input and generates appropriate adjectives for usage in text/speech as output. As a part of the design, we have proposed a novel CWW based similarity measure for calculating semantic similarity between adjective phrases. The D-RD is evaluated against two benchmarks, Onelook.com and WantWords Online RD, and is reported to outperform both for the test set under consideration.

INDEX TERMS Computing with words, CWW, GCR, generalized constraint representation, PNL, precisiated natural language, RD, reverse dictionary.

I. INTRODUCTION

Adjective phrases comprise a significant proportion of the data in scenarios involving natural language-based perceptions. Examples of such scenarios include analysis of product reviews, survey responses, comments to social media posts, etc. These phrases carry a significant amount of information content, thus strongly influencing the insights gathered from the data. A few statistics considering the scenario of analysis of product reviews are in line to support the argument. It is reported that online reviews impact the purchasing decision of 93% of the customers. Furthermore, while 92% of the B2B buyers are more likely to purchase after reading a trusted review (impact of positive reviews), 94% of the customers are reported to be convinced by online reviews to avoid a business (impact of negative reviews). [1]

Given the just mentioned statistics, disregarding such data content altogether would significantly lose knowledge. Moreover, employing approaches that could not aptly deal with perception-based information would not provide effective results. For instance, a product review that reads *Quality of product X is very very good* has a higher degree of approval of the product than the one which reads *The quality of the product is very good*. To be able to gather such insights, we propose employing the concept of Precisiated Natural Language (PNL) [1] based on Zadeh's paradigm of Computational Theory of Perception (CTP) [2], [3] which offers the required capabilities. Following, we give a brief introduction to the same.

In the paradigm of Computational Theory of Perceptions (CTP), proposed by Lotfi A. Zadeh [2], the objects of computation are words/ propositions from Natural Language (NL) rather than numbers. CTP rests on Computing With Words (CWW) [4] machinery, which in turn is based on the theory of Fuzzy Logic (FL). The underlying assumption of this paradigm is that the NL propositions can be precisiated in meaning by constraining the value of the variables. This converts these propositions into a form suitable for computation. Meaning precisiation of NL propositions extracts the implicit semantics and converts them into what is termed as the Generalized Constraint Representation (GCR). The subset of Natural Language which could be precisiated in meaning is termed Precisiated Natural Language (PNL).

The associate editor coordinating the review of this manuscript and approving it for publication was Amjad Ali.

[1] https://www.qualtrics.com/blog/online-review-stats/

This work is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 License.
For more information, see https://creativecommons.org/licenses/by-nc-nd/4.0/

In this paper, we deal with constraint explication in PNL and propose a GCR for adjective phrases. To signify our proposal, we demonstrate its application to the problem of Reverse Dictionary.

As the name suggests, Reverse Dictionary functions in a direction reverse to that of a forward dictionary. Specifically, while forward dictionaries map a word to its description, Reverse Dictionary aims to map a description to semantically appropriate word/s. Its primary objective is to address the unavoidable problem of escaped or unknown vocabulary faced by the language producers. Reverse Dictionary deals with the natural language description of the concept in the user's mind. As Natural Language is a system of perceptions, a primary concern is to be able to deal with the perception-based information in RD input. Consider, for example, the vocabulary word *jog*. Its possible user descriptions could be "to run slowly," "to run but not very fast," etc. Hence, to capture the user intent, it is important to deal with perception-based information appropriately. We apply the proposed GCR of adjective phrases in dealing with a specific instance of Reverse Dictionary. Particularly, we aim to employ the concept of PNL at the initial step in the workflow of traditional Information Retrieval (IR) systems. Doing this offers to extract and explicitly deal with implicit semantics, thus offering greater access to the semantic spectrum. Notably, this version of incorporation of PNL doesn't require the specification of membership functions.

The contribution of the paper is outlined as follows:

- Proposal of the Generalized Constraint Representation (GCR) of adjective phrase in Precisiated Natural Language (PNL).
- Proposal of incorporating PNL concepts into the traditional Information Retrieval system. We have demonstrated this by applying the proposed GCR of adjective phrases in designing a framework for a specific instance of Reverse Dictionary.
- Proposal of a novel linguistic hedge-based semantic similarity measure for adjective phrases as a part of the design of the Reverse Dictionary framework.

The paper is organized as follows: In Section II, we describe the adjective phrase in English and the role and analysis of the components of adjective phrases viz. adjectives and adverbs. In Section III, we present an interpretation of the adjective phrases in terms of CWW concepts and the semantics of adjectives and adverbs. In Section IV, we briefly explain the concept of generalized constraint and present the proposal of Generalized Constraint Representation (GCR) of adjective phrases. In Section V, we describe the problem of Reverse Dictionary and report the existing related works. In Section VI, we present the application scenario of the proposed GCR of adjective phrases and the proposed system framework with details. In Section VII, we discuss the implementation details and the evaluation of the results. Finally, we conclude the paper.

II. ADJECTIVE PHRASE IN ENGLISH: ROLE AND ANALYSIS OF ADJECTIVES AND ADVERBS

An adjective phrase is a phrase that modifies a noun. An adjective phrase could be:

- An adjective on its own, such as "tall," or,
- An adjective accompanied by adverbs, such as "not very tall."

An adjective phrase could occur either attributively or predicatively in a sentence. The adjective in the adjective phrase could occur at the start, in the end, or the middle, termed accordingly as head initial, head-final, and head-medial adjective phrase respectively. Sample phrase of each kind (in the same order) are *happy with her*, *very tall* and *quite doubtful about that* respectively. Following, we briefly discuss the role and analysis of the components of an adjective phrase viz. adjective and adverbs.

A. ROLE AND ANALYSIS OF ADJECTIVES

Adjectives are words that describe a noun by constraining its meaning. It is defined as "a word naming an attribute of a noun, such as sweet, red, or technical [1]." Adjectives could describe the following attributes of the noun: size, shape, age, color, origin, material, opinion, quality, type, purpose or participle forms. The count of adjectives for a noun in a sentence could be multiple describing the same or different attributes.

The position of an adjective in a sentence could be either attributive or predicative. Attributive adjectives appear before the described noun, as in *She is a tall lady*. In contrast, predicative adjectives appear after the described noun connected through a linking verb, as in *The lady is tall*. While most of the adjectives could take attributive and predicative positions, some could occur only in attributive (such as *main*) or predicative positions (such as *pleased*). The form of adjective is independent of the gender or count of the described noun. For example, the adjective *tall* appears in the same form with the noun *man, woman, people*.

B. ROLE AND ANALYSIS OF ADVERBS

Adverbs are words that modify the meaning of the word to which they are attached. Adverbs could be attached to a verb, adjective, another adverb or a noun. Example for each are *spoke loudly*, *really beautiful*, *very quickly*, *the meetings recently* respectively. In English, adverbs are grouped into the following meaning-related categories [5]:

- manner (e.g. quickly),
- degree (e.g. very),
- time (e.g. later)
- modality (e.g. probably), and
- frequency (e.g. weekly)

Primarily, adverbs intend to describe a verb. It occurs at the start of the clause, next to the verb or at the end of the clause. The degree adverbs, however, occur next to the word (in front) which they modify. Given the adjective phrase which is the subject of this paper, we have considered the adverb of type degree acting as a modifier of the adjective.

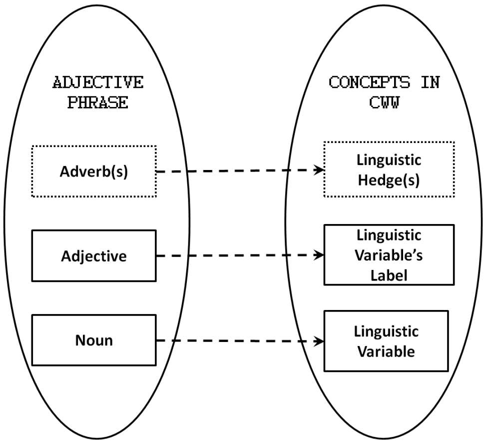

FIGURE 1. Interpretation of an adjective phrase in terms of CWW concepts.

III. CWW CONCEPTS FOR ADJECTIVES AND ADVERBS AND THEIR SEMANTICS

A. INTERPRETATION OF ADJECTIVE PHRASE IN TERMS OF CWW CONCEPTS

As introduced by Zadeh in [6]–[8], a linguistic variable is a variable which take words or sentence as their values. The values of a linguistic variable are termed linguistic labels and are grouped in a 'term set'. For example, 'height' is a linguistic variable, the labels for which could be 'tall', 'short' etc. A linguistic label can be modified in meaning by adding one or more linguistic hedges to it, e.g., adding the hedge 'very' to the label 'tall'. In line with this discussion, we interpret an adjective phrase as follows: the noun described by adjective phrase maps to a linguistic variable, the corresponding adjective maps to a particular label of that linguistic variable, and the optional adverbs corresponds to the linguistic hedges. The mapping is illustrated in Figure 1 (the dashed block depicts it as optional).

B. SEMANTICS OF ADJECTIVES AND ADVERBS

In the theory of PNL, the lexical category of adjectives is identified as *v-imprecise* [9], meaning they do not have a precise value. Some adjectives, however, are *m-precisiable*, i.e., they could be made precisiable in meaning. Mathematically, such adjectives are represented by a fuzzy set with an associated membership function. For instance, adjectives describing attributes that are numerical variables are m-precisiable. Examples of such adjectives include 'young', 'tall', 'fast' corresponding to the attribute 'age', 'height', 'speed' respectively.

Adjectives map to the linguistic label in the context of a linguistic variable (refer Fig 1). Specifically, a linguistic variable contains multiple linguistic labels of which typically two are primary terms (antonyms of each other) and the others are generated through the combination of a linguistic hedge and the modifier [10]. For example, the primary terms for the linguistic variable 'age' could be 'young' and 'old.' Other terms in the term set could be 'very young', 'not old', 'not very young' etc. using the hedge 'very' and the modifier 'not.'

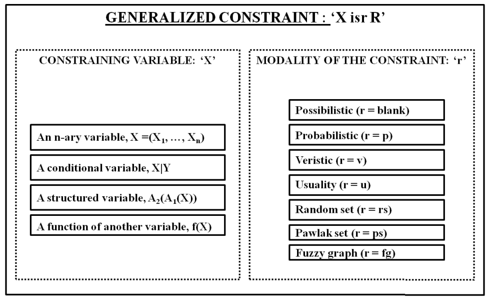

FIGURE 2. Possible forms of the constraining variable *X* and the modality *r* in a generalized constraint.

Degree adverbs mapped to the linguistic hedges are interpreted as operators which modify the fuzzy set associated with the linguistic label to which the adverb is attached. There are primarily three types of operations performed by linguistic hedge [11], [12] as mentioned below:

- Intensification: e.g., indeed.
- Concentration: e.g., very.
- Dilation: e.g., more or less.

IV. GENERALIZED CONSTRAINT REPRESENTATION: CONCEPT AND PROPOSAL FOR AN ADJECTIVE PHRASE

A. CONCEPT OF A GENERALIZED CONSTRAINT

Central to the CWW paradigm, the concept of a generalized constraint converts an NL proposition to a form suitable for carrying out computations. This is achieved by assuming that the NL proposition is the answer to an implicit question. This reveals the following three parameters: the variable whose meaning is constrained (termed as the constrained variable X), how the variable is constrained (termed as the modality of the constraint r) and the value of the constrained variable (termed as the constraining relation R). The GCR of an NL proposition, thus, takes the form (Equation 1),

$$X \text{ } isr \text{ } R \tag{1}$$

The constrained variable X and the modality of the constraint r can have multiple forms, some of which are listed in Fig. 2. The GCR of an NL proposition thus extracts the implicit semantics and conveys the information contained in it. For example, consider the proposition *Monika is young*. The implicit questions for this proposition could be *Who is young?* or *What is the age of Monika?*. Considering the latter question to be more relevant, the constrained variable corresponds to be Monika's age. Taking the modality of the constraint as possibilitic, a fuzzy set labeled 'young' depicts the constraining relation R. The GCR of the sample proposition thus takes the following form (Equation 2):

$$age(Monika) \text{ } is \text{ } young \tag{2}$$

B. PROPOSAL OF THE GENERALIZED CONSTRAINT REPRESENTATION FOR ADJECTIVE PHRASE

Following the discussion of the previous section (refer to Fig. 2), we propose the Generalized Constraint

Representation (GCR) of ***f(x) is R*** for adjective phrases in which the constraining variable X is a function of another variable x and the modality of the constraint is possibilistic. Formally, let the adjective phrase *ap* describe the characteristic/feature, C of the noun n, then the proposed GCR takes the form as shown in Equation 3.

$$f(x) \text{ is } R \rightarrow \ C(n) \text{ is } ap, \tag{3}$$

such that x corresponds to the noun n, f corresponds to the characteristic/feature C of the noun and R corresponds to the adjective phrase *ap*. For example, for the NL proposition, *John is very tall*, in which the adjective phrase *very tall* describes the characteristic *height* of the noun *John*, the corresponding GCR is (Equation 4):

$$height(John) \text{ is } very.tall \tag{4}$$

As mentioned in Section II, there are two variants of an adjective phrase depending upon the presence of accompanying adverbs. We outline the GCR corresponding to each variant, in line with Equation 3, as follows:

1) For an adjective phrase consisting of only an adjective, the GCR reduces to the form shown in Equation 5. The mathematical interpretation of the constraining relation depicted by the *adjective* (i.e., a linguistic label) is a fuzzy set with an associated membership function.

$$f(x) \text{ is } R \rightarrow C(n) \text{ is } adjective \tag{5}$$

2) An adjective phrase consisting of adjectives accompanied by degree adverbs takes the form shown in Equation 6. The mathematical interpretation of the constraining relation depicted by *adverb.adjective* is the fuzzy set corresponding to the adjective modified concerning the linguistics hedge represented by the adverb.

$$f(x) \text{ is } R \rightarrow C(n) \text{ is } adverb.adjective, \tag{6}$$

The corresponding protoform for the proposed GCR of adjective phrases become (Equation 7):

$$B(A) \text{ is } C \tag{7}$$

such that A, B, and C refer to the abstractions for noun (e.g. *John*), characteristic/feature of the noun (e.g., *height*) described by the adjective and the adjective phrase (e.g., *very tall*) respectively. The GCR proposed for an adjective phrase (as in Equation 3-5) [13] is in agreement with the studies [14], [15] which deal with the GCR for the lexical category of adjectives and adverbs individually.

V. REVERSE DICTIONARY: PROBLEM DESCRIPTION AND EXISTING WORKS

A Reverse Dictionary (may be referred to as RD, henceforth) takes as input a natural language description of a concept in the user's mind and generates as output word/s which semantically satisfies the input description. For instance, a sample input to RD might be "a feeling of intense sadness," and the corresponding output could be "melancholy," "desolation," "sorrow," along with other semantically similar words. Notably, the concept may mean anything, ranging from an entity to action or emotion. An RD typically works on the database of dictionary definitions to carry out the intended task.

While writing, it often happens that an appropriate word could not be recalled or is unknown. In such a scenario, a language producer might use a general description of the required word (called circumlocution) or a less apt term to fill in the void. The motivation in addressing the problem of RD is to attend to this need of the language producers. It aims to equip the language producer with the required vocabulary. Technically, an RD addresses the Tip-Of-Tongue (TOT) problem [16], [17], i.e., the word is on the tip of the tongue yet cannot be articulated. Worth mentioning, the available literary resources in the form of forward dictionaries and thesaurus do not suffice in addressing the mentioned problem. While the former is mostly suited for the readers, the latter provides limited functionality to language producers given topic-based entries.

Existing Reverse Dictionary Works and Research Gap: The above-mentioned problem faced by language producers is although not novel but is inescapable. The earliest attempts in the form of printed books [18], [19] with regards to this problem dates back to 1975. The literature witnesses many studies that have addressed this problem but in a limited fashion [20]–[25]. Lately, however, comprehensive studies have been reported by the research community. These include Information Retrieval approach-based Wordster Reverse Dictionary [26], Neural Language Model-based approach by Hill [27], Concept Blending approach by Calvo [28] and Graph-based approach by Thorat [29]. A detailed literature survey of RD works is reported in [30]. Recent works include [31]–[34] based on the study carried out by Hill *et al.* [27]. Owing to the inevitable nature of the problem, in addition to the research works, a few commercial applications are also available for general access, of which Onelook.com [35] is known to be most popular. It currently serves as the benchmark for comparing results of RD works.

Natural language is perception-based. Since the input to the Reverse Dictionary is a description of a concept in NL, it is natural to contain perception-based words. For example, for the vocabulary word *jog*, the input description could be "to run slowly," "to run but not very fast," etc.; for the word *sprint*, the input description could be "to run very very fast" etc. Disregarding the perceptions in user input may result in losing the user intent. Therefore, dealing with perception-based words is of primary importance in the context of Reverse Dictionary. We propose to incorporate the concept of PNL [1] in the framework of RD solution in this regard.

VI. APPLICATION SCENARIO: PROPOSED FRAMEWORK AND WORKING

Physical appearance descriptions of persons are perception-based natural language statements such as *she is thick and*

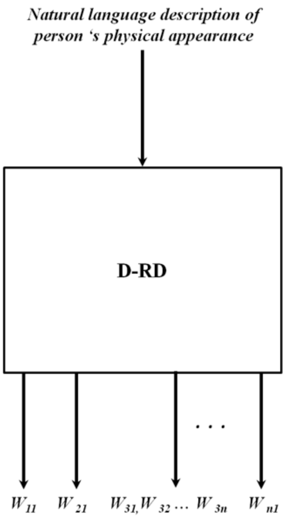

FIGURE 3. Block Diagram of D-RD Reverse Dictionary.

not very tall, *she is fairly tall and extremely good looking*, etc. To describe a person with an appropriate term/adjective while speaking (during a general conversation) or writing (for instance, describing a fictional character in professional writing) is often challenging. We demonstrate the application of the proposed GCR of adjective phrase in designing a specific instance of Reverse Dictionary, termed as D-RD: A Reverse Dictionary for Descriptive Words.

A. PROBLEM DESCRIPTION

A D-RD takes as input a natural language description of a person's physical appearance and provides as output words that best fit the mentioned description. For instance, it could accept the input *someone who is thick and not very tall* and provide the word 'stocky' as output. The block diagram for D-RD is shown in Fig. 3. As shown in the figure, the output may consist of a single word or multiple words depending upon the input description. Multiple words in the response depict that these words collectively satisfy the input description (the details are described in the following section).

B. ARCHITECTURE OF THE PROPOSED FRAMEWORK

Given the problem description mentioned above, a D-RD primarily deals with adjective phrases describing the characteristics of a person. Assuming a forward dictionary at disposal, we incorporate the proposed GCR of adjective phrases for extracting the implicit semantics from the dictionary definitions and the user input description and converting them into a precisiated form.

The idea is to employ PNL concepts into the traditional Information Retrieval (IR) system by incorporating precisiation as an initial step in the workflow. Specifically, instead of treating the definitions and the query as a bag of words, we convert these into the precisiated form and consider this form for further processing.

The precisiated form explicates the implicit semantics, hence, provides the capability to have greater access to the semantic spectrum. Notably, our version of incorporating PNL in the RD framework does not need precise specification of membership functions corresponding to the adjective phrase and its components (viz. adjectives and adverbs). The adjective phrase is, rather, treated as a phrasal unit in the precisiated form. To carry out the processing of the adjective phrases in the precisiated form, we propose a novel linguistic hedge-based semantic similarity measure for calculating the similarity between adjective phrases.

Based on the just presented idea, the architecture of the proposed system is illustrated in Fig. 4. It comprises three primary modules: Precisiation module (viz. Definition Precisiation module and Query Precisiation module), a Candidate Selection module, and Ranking module; and two databases containing the dictionary definitions and World Knowledge in the form of characteristics tables etc. The Precisiated Definitions database is generated as a part of the initial one-time processing carried out by the system.

A high-level working of the system is outlined as follows: The dictionary definitions contained in the dictionary database are precisiated using the Definition Precisiation module. It is a one-time process carried out in advance and stored in the Precisiated Definitions database. The user input, when obtained, is fed to the Query Precisiation module to obtain its corresponding precisiated form. These precisiation modules assume the availability of the World Knowledge in the form of a characteristic table. The Candidate Selection module generates candidates in the form of precisiated definitions corresponding to the precisiated query. These candidates are then fed to the Ranking module which generates a ranked list of responses in the form of vocabulary words comprising the system's output.

C. WORKING OF THE PROPOSED SYSTEM

In line with the discussion of the high-level working of the proposed system, following, we provide a detailed description of the working of the primary modules:

1) PRECISIATION MODULE

This module incorporates and implements the proposed Generalized Constraint Representation (GCR) of the adjective phrases. The idea is to extract the implicit semantics in the dictionary definitions as well as the user input description by converting them into their GCR (henceforth, may be referred to as the precisiated forms). It is carried out by the definition precisiation module and the query precisiation module, respectively. For carrying out precisiation, we assume the existence of a characteristic table in the world knowledge database. It stores the characteristics indexed by the words which describe them. For example, a sample entry of the table might be "Height: tall, short, ... " meaning that words like tall, short etc. describe the characteristic 'height' of a person.

Precisiation is carried out by undertaking the following steps:

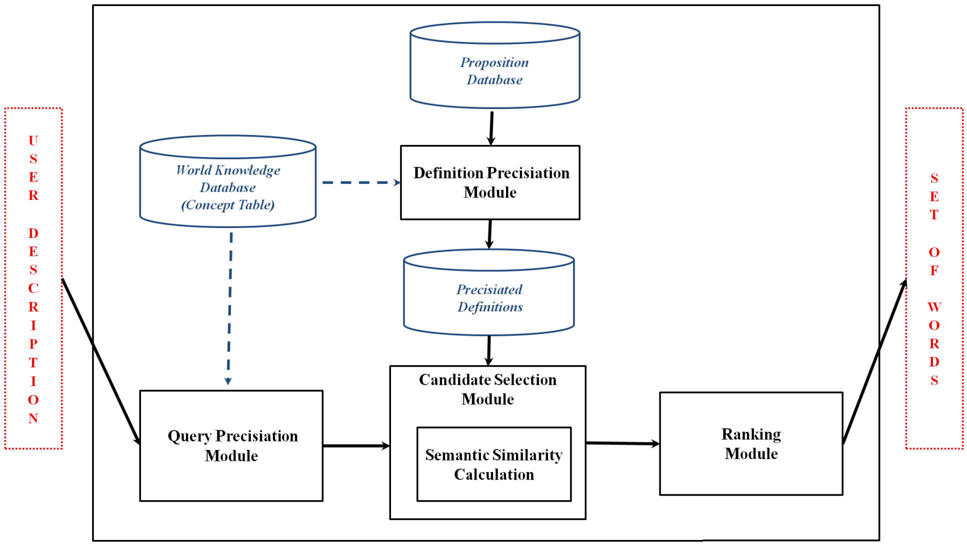

FIGURE 4. Architecture of the proposed system.

1) The definition/user input description is parsed to extract the embedded adjective phrases.
2) For each identified adjective phrase, the head adjective is looked up in the characteristic table to get the corresponding characteristic name.
3) After the lookup is completed, the characteristic name *c* corresponding to the adjective phrase *ap* is represented in the form of the proposed GCR. The GCR corresponding to each adjective phrase is ANDed to constitute the precisiated form of the dictionary definition (or the input description as the case may be).

For instance, considering the dictionary definition of the word *stocky*: 'A stocky person has a body that is broad, solid, and short. [2]' Assuming the characteristic table contains the adjectives 'broad,' 'solid' and 'short' indexed by characteristics 'size,' 'build' and 'height' respectively, the corresponding precisiated form will be as shown below:

$$size(stocky)\ IS\ broad\ AND\ build(stocky)\ IS\ solid \\ AND\ height(stocky)\ IS\ short \quad (8)$$

It is to be noted that, while precisiation of the dictionary definitions is done initially as a one-time process, precisiation of the query is done on the fly.

2) CANDIDATE SELECTION MODULE

This module selects those precisiated dictionary definitions (termed as 'potential candidates' in the later discussion) whose corresponding vocabulary words might constitute the response set for a given user input description. The following three criteria are taken into account while selecting potential candidates.

[2]Collins Dictionary, https://www.collinsdictionary.com/

a: CRITERIA FOR CANDIDATE SELECTION

The criteria for the selection of potential candidates are as follows:

Criteria 1: The precisiated definition contains at least one characteristic mentioned by the user (i.e., in the precisiated query).

- This is the basic condition which a potential candidate must qualify to appear in the result set.
- This ensures that the potential candidate is relevant to the user requirement.

Criteria 2: For the characteristics present in the query, not all the adjective phrases for the corresponding characteristic in the precisiated definition is in contrast (have negative semantic similarity) with the adjective phrase present in the precisiated query.

- The second criterion assures that while the candidate is relevant, it conforms to the user requirement.
- This filtering essentially discards those precisiated definitions which have the characteristics (some or all) mentioned by the user. Still, all the values are contrasting to that which the user requires.
- This criterion demands a measure of semantic similarity between the adjective phrases (corresponding to a characteristic) occurring in the precisiated definition and the precisiated query. Given this, we propose a novel semantic similarity measure for adjective phrases as explained in detail in the following subsection (Subsection VI-C3.b).

Criteria 3: The overall similarity of the precisiated definition with the user input description is above a certain threshold.

- This ensures that the candidates which are relevant as well as conformant to the requirement are of good quality.

- This filtering essentially discards those precisiated definitions with some or all the characteristics mentioned by the user such that some or all are having positive similarity with the values mentioned by the user. Yet, the overall similarity is below a minimum value.
- This criterion demands a measure of an aggregated similarity measure which is chosen to be the mean of the individual characteristic similarity (in the implementation).

The algorithm for candidate selection is outlined in Section VI-D (refer to Algorithm 3 and the related algorithms, Algorithm 4 - 8). Given the criterion just mentioned, we take into account the following two threshold values, which are design decisions, for making the selection for the candidates:

- Individual characteristic similarity threshold α concerning Criteria 2.
- Aggregated similarity threshold β concerning Criteria 3.

b: SEMANTIC SIMILARITY MEASURE FOR ADJECTIVE PHRASES

To be able to access the semantic spectrum adequately, we employ the concepts of Computing With Words (CWW) in devising the similarity measure [36]. Particularly, we map the adjective and adverb/s in the adjective phrase to the CWW concepts of linguistic variable and linguistic hedge/s respectively. For example, in the adjective phrase, "very very tall" the adjective *tall* depicts the linguistic variable and the adverb *very* depicts the linguistic hedge. We have considered two types of linguistic hedges: concentrators and dilators. While concentrators (e.g. *very*) intensify the meaning of the adjective to which it is attached, dilators (e.g. *fairly*) weaken the impact of its meaning. Additionally, we have considered the inverter hedge *not* which complements the meaning of the adjective to which it is attached.

The underlying idea of the proposed similarity measure is to place the adjective phrases (that have to be compared) along a linear line and compute the distance between them. Intuitively, lower is the distance between the two phrases; higher is the similarity between the two. The key step in calculating this similarity is to calculate the position of the adjective phrase along the linear line. We perform this by using the mathematical expression of the linguistic hedge. Given that the linguistic value X is defined by the membership function $[\mu_X(u)]$ of U, then the modified membership function due to the effect of hedge h is given as:

$$h(X) = [\mu_X(u)]^e \tag{9}$$

The exponent value e depends upon the nature and the strength of the linguistic hedge h. While concentrators have an exponent value greater than 1, for dilators, their value is less than 1. Furthermore, the magnitude of the exponent value in the mathematical expression of the linguistic hedge (concentrator/ dilator) is a measure of the effect the linguistic hedge makes on the adjective. We use these considerations in calculating the position of the adjective phrase along a linear line with respect to the position of the head adjective (fixed initially). The algorithm for semantic similarity is outlined in the following section (refer to Algorithm 4-7) and detailed in [36].

3) RANKING MODULE

This module performs the ranking of the given set of potential candidates to generate a ranked list of responses as output. The vocabulary words corresponding to the potential candidates (which are precisiated definitions) constitute the output response. Given the nature of the user requirement, the system is designed to generate two kinds of responses: simple response and composite response. Additionally, the system also generates suggestive response. Furthermore, three parameters are accounted for while ranking the potential candidates. The details of these aspects are given below:

a: TYPES OF RESPONSE

Following are the types of the response generated by the system:

1) Simple response: These are responses consisting of a single vocabulary word. It corresponds to the scenario where a single adjective fitting the user description is available. For example, for the description 'someone who is extremely ugly', the simple response, consisting of a single adjective, may correspond to the word 'hideous.'
2) Composite response: These are responses consisting of multiple vocabulary words. The vocabulary words in a composite response collectively satisfy the query. It corresponds to the scenario where a single adjective could not satisfy the query, possibly due to the inexistence of such an adjective in the language or the limited vocabulary addressed by the system. For example, for the description 'someone who is short, small in size, fairly pale and not healthy', a single adjective would not fit. The composite response, consisting of multiple adjectives, may correspond collectively to the adjectives 'pygmy and cadaverous'.
3) Suggestive response: In addition to the simple and composite responses, the system generates yet another kind of response termed the suggestive response. These are words that satisfy the characteristics required by the user but have some additional characteristics associated with them as well. It is helpful in cases where the user might have left out some features unknowingly or forgetfully in the input description, and the generated suggestive responses turn out to be even more befitting. While outputting suggestive responses, the system displays the additional characteristics and the corresponding adjectives as well. For example, for the description 'someone who is weak', the simple response may correspond to the word 'feeble'. The suggestive response for the same description may

correspond to the adjective 'etiolated' having the additional characteristic of 'pale looking'.

b: PARAMETERS OF RANKING

A potential candidate has one/more characteristic-value pairs associated with it. Specifically, the characteristic corresponds to the characteristic function depicted by the adjective appearing in the dictionary definition and the value corresponds to the adjective itself. Similarly, the user query has one/more characteristic-value pairs associated with it.

The potential candidates obtained after candidates selection has the following information available with it, based on which ranking is performed (in the order as listed, refer Algorithm 9 in Section VI-D):

- Similarity value:
  Corresponding to each characteristic in the characteristic-value pairs, the similarity score of the values appearing in the potential candidate with the values appearing in the query. As mentioned, a novel hedge-based similarity algorithm is proposed to calculate the similarity between two values (in the form of an adjective phrase). The similarity values could be above or below relative to a threshold value (referred to as α as in Section VI-C2.a). Also, a possible case could be that one/more characteristics appearing in the user query are not appearing in the potential candidate.
  For ranking, we consider the count of characteristics that have similarity values greater than the threshold. The higher is the count of a potential candidate, the higher it appears in ranking. Intuitively, if a potential candidate holds all the characteristics mentioned by the user in the query and has similarity values above the threshold for each one of those, then it is a good fit for the user's need.
- Suggestive features:
  Those characteristics of a potential candidate that do not appear in the query are termed as suggestive features. As stated previously, these are incorporated in the output words of the response list in the anticipation that it might be more befitting to the user's need. The ideal case is however, the exact match of the potential candidate's characteristics set with that of the user query. Hence, for the ranking process, we consider the count of suggestive features in increasing order of values.
- Aggregate similarity score:
  The overall similarity of a potential candidate with the query in terms of the values of the common characteristics. Values above a certain threshold (referred to as β as in Section VI-C2.a) are considered. The higher is the value for a potential candidate, the higher it appears in ranking.

D. ALGORITHMS

In line with the discussion presented in the previous subsection (Section VI-C), we now outline the algorithms of the primary modules of the framework. Table 1 lists the notation used.

TABLE 1. Notations used.

Notation	Meaning
w	Vocabulary word
d_w	Definition of a vocabulary word
D_w	Precisiated definition of a vocabulary word
q	Query i.e. user input description
Q	Precisiated query
c_{adj}	characteristic described by adjective adj
ap	Adjective phrase
$cSet_Q$	Characteristic set of precisiated query Q
$cSet_C$	Characteristic set of a potential candidate
$value_Q(c)$	Value of characteristic c in precisiated query
$value_{D_w}(c)$	Value of characteristic c in precisiated definition
$value_C(c)$	Value of characteristic c in a candidate
CT	Characteristics table
α	Threshold for individual characteristic semantic similarity score
β	Threshold for aggregate semantic similarity score
$sim_{CWW}(ap_1, ap_2)$	CWW based similarity measure between adjective phrase ap_1 and ap_2
$CS(Q)$	Definitions comprising the candidate set for precisiated query Q
C	A candidate belonging to CandidateSet(Q)
$aggSim(C)$	Aggregate similarity of candidate c
$hedge(ap)$	Hedge set of adjective phrase ap
RS	Result set

Algorithm 1 Precisiate Definition.

Input: definition, d_w

$D_w = null$
Extract the adjective phrases ap in d_w
for all ap **do**
  Get the head adjective adj of ap
  $c_{adj} = lookup(adj)$ in CT
  Append '$c_{adj}(w) = ap$' to D_w separated by '&'
end for

Algorithm 2 Precisiate Query.

Input: query, q

$Q = null$
Extract the adjective phrases ap in q
for all ap **do**
  Get the head adjective adj of ap
  $c_{adj} = lookup(adj)$ in CT
  Append '$c_{adj}(q) = ap$' to Q separated by '&'
end for

VII. EVALUATION OF THE PROPOSED FRAMEWORK

We have carried out separate evaluations for the simple responses and the composite responses. The results of the proposed CWW based semantic similarity measure could be referred to in [36].

A. IMPLEMENTATION DETAILS

We have implemented the proposed framework considering the most commonly used characteristics for describing the physical appearance of humans. The considered characteristics are 11 and are listed as follows: *physical strength,*

Algorithm 3 Candidate Selection.

Input: precisiated query, Q

Initialize $CS(Q)$ to *null*
for all D_w **do**
  for all $c \in cSet_Q$ **do**
    if $value_{D_w}(c) \neq null$ and $c \notin RS$ **then**
      Add D_w to $CS(Q)$
      break
    end if
  end for
end for
for all $C \in CS(Q)$ **do**
  if $sim_{CWW}(value_Q(c), value_C(c)) < \alpha$ for all c in $cSet_Q$ **then**
    remove candidate C from $CS(Q)$
  end if
  if $aggSim(C) < \beta$ **then**
    remove candidate C from $CS(Q)$
  end if
end for
return $CS(Q)$

Algorithm 4 Similarity Calculation.

Input: adjective phrases ap_1 and ap_2

Get the adjectives adj_1 and adj_2 of ap_1 and ap_2 respectively
$pos_1, pos_2 = getInitialPosition(adj_1, adj_2)$
for i in $(1, 2)$ **do**
  for each hedge h in $hedge(ap_i)$ **do**
    $pos_i = getPositionwrtHedge(pos_i, adj_{ap_i}, h)$
  end for
  if inverter count in ap_i is odd **then**
    $pos_i = getPositionwrtInverter(pos_i)$
  end if
end for
$difference = |pos_2 - pos_1|$
$sim_{CWW}(ap_1, ap_2) = 1 - difference$
return $sim_{CWW}(ap_1, ap_2)$

health, body structure, body size, height, weight, physique, looks, beauty, body movement, and appearance. For the same, we compiled a vocabulary of 250 words and compiled its dictionary definitions from 6 different sources[3,4,5,6,7,8]. The system accepts queries in the form of *'someone who is'* followed by a list of adjective phrases. Sample queries are listed in Table 2.

The collected word-definition pairs are converted into the proposed precisiated form as an initial one-time offline step.

[3]Wiktionary, https://en.wiktionary.org/wiki/Wiktionary:Main_Page
[4]Cambridge Dictionary, https://dictionary.cambridge.org/
[5]Macmillan Dictionary, https://www.macmillandictionary.com/
[6]Merriam Webster Dictionary, https://www.merriam-webster.com/
[7]Oxford Dictionary, https://www.oxfordlearnersdictionaries.com/
[8]Collins Dictionary, https://www.collinsdictionary.com/

Algorithm 5 getInitialPosition.

Input: adjectives adj_1, adj_2

if adj_1 and adj_2 are synonyms of each other **then**
  if adj_1 is primary_term_1 and adj_2 is primary_term_1 **then**
    $pos_1 = pos_2 = 1.0$
  else
    $pos_1 = pos_2 = 0.0$
  end if
else if adj_1 and adj_2 are antonyms of each other **then**
  if adj_1 is primary_term_1 **then**
    $pos_1 = 1.0, pos_2 = 0.0$
  else
    $pos_1 = 0.0, pos_2 = 1.0$
  end if
else { the adjectives are not comparable}
  $pos_1 = pos_2 = -999$
end if
return pos_1, pos_2

Algorithm 6 getPositionwrtHedge.

Input: position *position*, adjective *adj*, hedge h

if *adj* is primary_term_1 and h is a concentrator **then**
  $position += (1 - 1/power(h))$
else if *adj* is primary_term_1 and h is a dilator **then**
  $position -= power(h)$
else if *adj* is primary_term_2 and h is a concentrator **then**
  $position -= (1 - 1/power(h))$
else if *adj* is primary_term_2 and h is a dilator **then**
  $position += power(h)$
end if
return *position*

As required by the precisiation step, we have made available the concept table, which comprises the considered characteristics and the list of corresponding adjectives. It is to be noted that the precisiation of the definitions is carried out in a semi-automated manner; those definitions (complex phrase forms) which are in a form that could not be converted automatically into the precisiated form are converted manually. The implementation considers 15 hedges, out of which 8 are concentrators (viz. *very very, extremely, very, slightly, little, too, excessively, dazzlingly*), 6 are dilators (viz. *almost, rather, somewhat, quite, fairly, more or less*) and an inverter *not*. The mathematical expressions of the linguistic hedges are taken from literature.

B. EVALUATION OF SIMPLE RESPONSE

1) TEST SET PREPARATION FOR SIMPLE RESPONSE

We have randomly selected 25 words from the total vocabulary addressed by the system. We have formulated the query corresponding to these words using WordNet definitions and have evaluated the generated responses. Notably, the evaluation carried out for the test set is unbiased as the

Algorithm 7 getPositionwrtInverter.

Input: position *position*, adjective *adj*

if $position == 1$ **then**
  $position = 0;$
else if $position == 0$ **then**
  $position = 1;$
else if $(position > 1.0)$ **then**
  if *adj* is primary_term_1 **then**
    $position = 1 - (position - 1)$
  else
    $position = position * -1$
  end if
else
  if *adj* is primary_term_2 **then**
    $position = position + 1$
  else
    $position = position * -1$
  end if
end if
return *position*

Algorithm 8 Aggregate Similarity Calculation.

Input: candidate, C

$sum = 0$
$count = 0$
for all characteristic c of candidate C **do**
  if $sim_{CWW}(value_Q(c), value_C(c)) > \alpha$ **then**
    $sum += sim_{CWW}(value_Q(c), value_C(c))$
    $count += 1$
  end if
end for
$aggSim(C) = sum/count$
return $aggSim(C)$

Algorithm 9 Ranking.

Input: candidate set, $CS(Q)$

for all candidate $C \in CS(Q)$ **do**
  Get the count $c'(C)$ of characteristics that are absent, $|cSet_Q - cSet_C|$ or have similarity values below threshold, $sim_{CWW}(value_Q(c), value_C(c)) < \alpha$
  Get the count of suggestive features, $c''(C) = |cSet_C - cSet_Q|$
  Get the value of aggregate score, $aScore(C) = aggSim(C)$
end for
Sort the candidates in the order of increasing value of c'
For $C's$ with equal value of c', sort in the order of increasing value of c''.
For $C's$ with equal value of c'', sort in the order of decreasing value of *aScore*.
Output the top required number of candidates in the sorted list:
for all candidate C in the sorted list **do**
  if $c'(C) = 0$ **then**
    Create a simple response with the single vocabulary word corresponding to C. Add suggestive features to the simple response (if $c''(C) \neq 0$). Set $Q = null$.
  else
    Create a composite response with the vocabulary word corresponding to C. Add suggestive features to the composite response (if $c''(C) \neq 0$).
    Update Q by removing characteristics from $cSet_Q$ that are satisfied and execute for the updated Q.
  end if
end for

WordNet definitions are not used in compiling the system's database.

2) EVALUATION RESULTS OF SIMPLE RESPONSE

For the simple responses, the obtained results are compared against the commercial benchmark Outlook Reverse Dictionary [35] and WantWords, an open-source Online Reverse Dictionary [37]. While the D-RD is designed to generate adjective words only, Onelook and WantWords RD are generic and deal with words belonging to different part-of-speech. For a fair comparison, we have considered the response list under the 'Adjectives' tab for both. The evaluation metrics considered are Accuracy@1, Accuracy@10, Rank median, and Mean Reciprocal Rank. Some sample responses and the evaluation scores are reported in Table 2 and Table 3 respectively.

Discussions. Based on the values of the evaluation metrics, following needs to be highlighted:

- On the basis of accuracy@1 metric scores, it is concluded that the D-RD generates correct response at the topmost position for a greater number of test queries in comparison to the Onelook and WantWords RDs.
- Furthermore, considering the top 10 responses, the accuracy of the D-RD improves significantly and is again higher than the other two comparison benchmarks.
- The lower value of rank median for D-RD indicates that it can generate correct response near the top position in the response list. Specifically, the rank median for D-RD is calculated to be 1, i.e., it can generate the correct response at the topmost position for most test queries.
- A RD should return the best relevant word at the highest position in the response list. Given this, the MRR metric is important to consider. The higher value of the MRR metric for D-RD relative to those of Onelook and WantWords justifies its better performance for the test set under consideration.

C. EVALUATION OF COMPOSITE RESPONSE

1) TEST SET PREPARATION FOR COMPOSITE RESPONSE

To evaluate composite responses, we have considered queries addressing more than three characteristics. This is given the

TABLE 2. Sample queries and their responses.

S.No.	Query	System Response	Rank	Type of Response	Suggestive features
1	someone who is not thin	thick	1	Simple	-Not suggestive-
2	someone who is excessively fat	obese	1	Simple	-Not suggestive-
3	someone who is weak	feeble	1	Simple	-Not suggestive-
4	someone who is weak	etiolated	7	Simple	pale
5	someone who is not short, too good-looking, too beautiful, fast	statuesque and quick	1	Composite	large
6	someone who is large, fairly awkward, too ugly, rather untidy	ungainly and hideous	5	Composite	-Not suggestive-
7	someone who is rather weak, slightly sick, skinny, fast	frail and quick	1	Composite	-Not suggestive-
8	someone who is healthy, large, excessively beautiful, tall	strapping and gorgeous	1	Composite	strong

TABLE 3. Evaluation metric scores for simple responses.

Metric/ System	Accuracy@1	Accuracy@10	Rank Median	Mean Reciprocal Rank
D-RD	0.40	0.76	1	0.6303
Onelook.com	0.16	0.56	7	0.4508
WantWords Online RD	0.08	0.28	4	0.4656

consideration that on average, a vocabulary word encompasses not more than three characteristics.

We have constructed three bins corresponding to the characteristic count of 4 and 5, and a total of 25 queries are formulated for evaluation (individual count corresponding to bins being 12 and 13). The query formulation is done through the following strategy:

- Randomly choosing a characteristic from the characteristic list.
- For the chosen characteristic randomly choosing an adjective from its corresponding list of adjectives.
- Further, we decide the occurrence of a hedge (i.e., an adverb) randomly. If yes, we randomly pick a hedge from the corresponding list of hedges.

Some sample queries are listed in Table 2.

2) EVALUATION RESULTS OF COMPOSITE RESPONSE

Given that the generation of composite response is a feature particular to the proposed framework, this could not be compared directly with the standard benchmark (unlike the evaluation of the simple responses). Some sample composite responses are shown in Table 2. To evaluate composite responses, we have engaged a panel of 4 English experts for manual evaluation. The basis of evaluation is the degree of semantic association and conformance of the topmost response generated by D-RD with the user input description. The responses are marked on a scale of 0-3. The scores 0, 1, 2 and 3 correspond to the poor, weak, medium, and strong semantic association, respectively.

Based on evaluation scores collected from the panel of experts for the complete test set, the average score of D-RD is calculated to be 2.2. Specifically, the average values for queries consisting of 4 and 5 characteristics are 2.27 and 2.13. The D-RD, thus, generates composite responses having medium to strong semantic association.

VIII. CONCLUSION

In this paper, we have proposed the GCR form for adjective phrases and demonstrated its application in building D-RD, a Reverse Dictionary for descriptive words. In the process of design of D-RD, we have proposed a novel CWW-based semantic similarity for adjective phrases [36]. One of the key features of D-RD is that it is designed to generate composite responses (multiple adjectives which collectively satisfy a query) and suggestive responses (adjectives with additional characteristics which satisfy the query), as the case may be. The evaluation is carried out against Onelook.com and WantWords, and it is found that D-RD outperforms both for the test set under consideration. The D-RD could be extended to tune the generated responses based on additional contextual information such as gender for which the adjective is sought and the polarity (positive/negative) of its usage. Furthermore, the design of D-RD could be extended/applied to build other related specific instances such as Reverse Dictionary for descriptive words of speech, behavior, product feedback, etc.

REFERENCES

[1] L. A. Zadeh, "Precisiated natural language (PNL)," *AI Mag.*, vol. 25, no. 3, pp. 74–92, 2004.

[2] L. A. Zadeh, "A new direction in AI: Toward a computational theory of perceptions," *Ai Mag.*, vol. 22, no. 1, pp. 73–84, 2001.

[3] L. A. Zadeh, "From computing with numbers to computing with words: From manipulation of measurements to manipulation of perceptions," *Int. J. Appl. Math. Comput. Sci.*, vol. 12, no. 3, pp. 307–324, 2002.

[4] L. A. Zadeh, "Fuzzy Logic= computing with words," in *Computing with Words in Information/Intelligent Systems*. Cham, Switzerland: Springer, 1999, pp. 3–23.

[5] J. Eastwood, *Oxford Guide to English Grammar*. London, U.K.: Oxford Univ. Press, 1994.

[6] L. A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning—I," *Inf. Sci.*, vol. 8, no. 3, pp. 199–249, 1975.

[7] L. A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning—II," *Inf. Sci.*, vol. 8, no. 4, pp. 301–357, Jan. 1975.

[8] L. A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning—III," *Inf. Sci.*, vol. 9, no. 1, pp. 43–80, Jan. 1975.

[9] L. A. Zadeh, "From search engines to question answering systems—The problems of world knowledge, relevance, deduction and precisiation," in *Capturing Intelligence*, vol. 1. Amsterdam, The Netherlands: Elsevier, 2006, pp. 163–210.
[10] L. A. Zadeh, "PRUF—A language for the representation of meaning in natural languages," *Int. J. Man-Mach. Stud.*, vol. 10, no. 4, pp. 395–460, 1978.
[11] L. A. Zadeh, "Outline of a new approach to the analysis of complex systems and decision processes," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-3, no. 1, pp. 28–44, Jan. 1973.
[12] L. A. Zadeh, "A fuzzy-set-theoretic interpretation of linguistic hedges," *J. Cybern.*, vol. 2, no. 3, pp. 4–34, Jan. 1972.
[13] B. Siddique and M. S. Beg, "Generalized constraint representation of adjectives in PNL," in *ICT Systems and Sustainability*. Singapore: Springer, 2022, pp. 827–835.
[14] A. Soto, J. A. Olivas, and M. E. Prieto, "Using generalized constraints and protoforms to deal with adjectives," in *Proc. IEEE Int. Fuzzy Syst. Conf.*, Jun. 2007, pp. 1–6.
[15] A. Soto, J. A. Olivas, and M. E. Prieto, "Using generalized constraints and protoforms to deal with adverbs," in *Proc. EUSFLAT Conf.*, 2007, pp. 119–126.
[16] B. L. Schwartz and J. Metcalfe, "Tip-of-the-tongue (TOT) states: Retrieval, behavior, and experience," *Memory Cognition*, vol. 39, no. 5, pp. 737–749, Jul. 2011.
[17] R. Brown and D. McNeill, "The 'tip of the tongue' phenomenon," *J. Verbal Learn. Verbal Behav.*, vol. 5, no. 4, pp. 325–337, 1966.
[18] T. M. Bernstein and J. Wagner, *Bernstein's Reverse Dictionary*. New York, NY, USA: Times Books, 1975.
[19] D. Edmonds, *The Oxford Reverse Dictionary*. London, U.K.: Oxford Univ. Press, 1999.
[20] H. V. Crawford and J. Crawford, "Reverse electronic dictionary using synonyms to expand search capabilities," U.S. Patent 5,649 221, Jul. 15, 1997.
[21] I. D. El-Kahlout and K. Oflazer, "Use of Wordnet for retrieving words from their meanings," in *Proc. Global Wordnet Conf. (GWC)*, 2004, pp. 118–123.
[22] S. Bilac, W. Watanabe, T. Hashimoto, T. Tokunaga, and H. Tanaka, "Dictionary search based on the target word description," in *Proc. 10th Annu. Meeting Assoc. NLP (NLP)*, 2004, pp. 556–559.
[23] D. Dutoit and P. Nugues, "A lexical database and an algorithm to find words from definitions," in *Proc. 15th Eur. Conf. Artif. Intell.*, 2002, pp. 450–454.
[24] M. Zock and S. Bilac, "Word lookup on the basis of associations: From an idea to a roadmap," in *Proc. Workshop Enhancing Using Electron. Dictionaries (ElectricDict)*, 2004, pp. 29–35.
[25] M. Zock and D. Schwab, "Lexical access based on underspecified input," in *Proc. workshop Cognit. Aspects Lexicon (COGALEX)*, 2008, pp. 9–17.
[26] R. Shaw, A. Datta, D. VanderMeer, and K. Dutta, "Building a scalable database-driven reverse dictionary," *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 3, pp. 528–540, Mar. 2013.
[27] F. Hill, K. Cho, A. Korhonen, and Y. Bengio, "Learning to understand phrases by embedding the dictionary," *Trans. Assoc. Comput. Linguistics*, vol. 4, pp. 17–30, Dec. 2016.
[28] H. Calvo, O. Méndez, and M. A. Moreno-Armendáriz, "Integrated concept blending with vector space models," *Comput. Speech Lang.*, vol. 40, pp. 79–96, Nov. 2016.
[29] S. Thorat and V. Choudhari, "Implementing a reverse dictionary based on word definitions using a node-graph architecture," in *Proc. 26th Int. Conf. Comput. Linguistics (COLING)*, 2016, pp. 2797–2806.
[30] B. Siddique and M. M. S. Beg, "A review of reverse dictionary: Finding words from concept description," in *Proc. Int. Conf. Next Gener. Comput. Technol.*, Cham, Switzerland: Springer, 2018, pp. 128–139.
[31] L. Zheng, F. Qi, Z. Liu, Y. Wang, Q. Liu, and M. Sun, "Multi-channel reverse dictionary model," in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 1, 2020, pp. 312–319.
[32] Y. Morinaga and K. Yamaguchi, "Improvement of neural reverse dictionary by using cascade forward neural network," *J. Inf. Process.*, vol. 28, pp. 715–723, Nov. 2020.
[33] H. Yan, X. Li, and X. Qiu, "BERT for monolingual and cross-lingual reverse dictionary," 2020, *arXiv:2009.14790*.
[34] A. Agrawal, A. R, K. S. A. Shanly, K. Vaishnaw, and M. Singh, "Reverse dictionary using an improved CBoW model," in *Proc. 8th ACM IKDD CODS 26th COMAD*, Jan. 2021, p. 420.
[35] Onelook.com. *OneLook Reverse Dictionary*. [Online]. Available: http://www.onelook.com/
[36] B. Siddique and M. M. S. Beg, "'Computing with words'-based semantic similarity measure for adjective phrases," in *Proc. Int. Conf. Electr., Electron. Comput. Eng. (UPCON)*, Nov. 2019, pp. 1–5.
[37] F. Qi, L. Zhang, Y. Yang, Z. Liu, and M. Sun, "WantWords: An open-source online reverse dictionary system," in *Proc. Conf. Empirical Methods Natural Lang. Processing, Syst. Demonstrations*, 2020, pp. 175–181.

BUSHRA SIDDIQUE received the B.Tech. degree (Hons.) in computer engineering and the M.Tech. degree (Hons.) in software engineering from Aligarh Muslim University, Aligarh, India, in 2014 and 2016, respectively, where she is currently pursuing the Ph.D. degree in the area of computer engineering.

Her current research interests include data mining, artificial intelligence, information retrieval, and soft computing.

M. M. SUFYAN BEG (Senior Member, IEEE) received the B.Tech. degree in electronics from Aligarh Muslim University, India, in 1992, the M.Tech. degree in microelectronics from the IIT Kanpur, India, in 1994, and the Ph.D. degree in the area of computer technology from the IIT Delhi, India, in 2004.

From November 2005 to October 2006, he visited the University of California at Berkeley, USA, as a BT Fellow. He worked as a Professor at the Department of Computer Engineering, Jamia Millia Islamia (A Central University), New Delhi, from March 2007 to January 2014. Over there, he also worked as the Head of the Department of Computer Engineering, from February 2008 to February 2011. Since January 2014, he has been a Professor at the Department of Computer Engineering, Aligarh Muslim University. Since August 2015, he has been working as the Principal of the Z. H. College of Engineering and Technology. He has been an invited lecturer at many places in the USA, U.K., Saudi Arabia, Pakistan, and India. He has published a book, 28 book chapters, and over 165 research articles on a wide range of topics in web mining and soft computing. His current research interests include parallel and distributed processing, soft computing, question answering systems, natural language processing, and web mining and searching.