J. Balaji*, T.V. Geetha and P. Ranjani

Graph-Based Bootstrapping for Coreference Resolution

Abstract: Coreference resolution is a challenging natural language processing task, and it is difficult to identify the correct mentions of an entity that can be any noun or noun phrase. In this article, a semisupervised, two-stage pattern-based bootstrapping approach is proposed for the coreference resolution task. During Stage 1, the possible mentions are identified using word-based features, and during Stage 2, the correct mentions are identified by filtering the non-coreferents of an entity using statistical measures and graph-based features. Whereas the existing approaches use morphosyntactic and number/gender agreement features, the proposed approach uses semantic graph-based context-level semantics and nested noun phrases in the correct mentions identification. Moreover, mentions without the number/gender information are identified, using the context-based features of the semantic graph. The evaluation performed for the coreference resolution shows significant improvements, when compared with the word association-based bootstrapping systems.

Keywords: Coreference resolution, bootstrapping, semantic graphs, universal networking language (UNL).

2010 Mathematics Subject Classification: 68 Computer Science, 68T50 Natural Language Processing.

***Corresponding author: J. Balaji,** Department of Computer Science and Engineering, Anna University, Chennai, Tamil Nadu, India, e-mail: jagank.balaji@gmail.com
T.V. Geetha: Department of Computer Science and Engineering, Anna University, Chennai, Tamil Nadu, India
P. Ranjani: Department of Information Science and Technology, Anna University, Chennai, Tamil Nadu, India

1 Introduction

Coreference resolution is a task of identifying entities that corefer with other entities in a piece of text. Coreference resolution is a key task in discourse analysis and in many natural language processing (NLP) applications such as question answering, summarization, machine translation, information extraction.

The term “coreference resolution” indicates that two entities refer to the same object. Whereas the pronominal anaphora resolution is a simpler task of identifying entities associated with pronouns, coreference resolution is the task of identifying all coreferents in a document that refer to the same entity. Table 1 shows the example of anaphora representing a person and its antecedent. Here, the pronoun “avan” in sentence S2 refers to an antecedent “Raman” in sentence S1.

The identification of coreferents (commonly referred as mentions) of an entity is a difficult task because mentions can occur anywhere in the document and they need not necessarily occur along with the entity. An example Tamil Wiki document is shown in Table 3. Moreover, it requires filtering in which the non-coreferents among many identified possibilities for a particular entity are eliminated and the correct set of mentions is selected.

Coreference resolution is carried out using different types of features, which include morphological and lexical information such as number/gender agreement, syntactic information such as nouns and noun phrases, and semantic information such as synonyms, semantic constraints (e.g., person, location), and other semantic relations that specifically identify coreferring entities. There are several types of information ranging from morphology to pragmatics that play an important role in coreference resolution.

Various machine learning approaches have been attempted for the coreference resolution task. Unlike linguistic approaches [9, 10] for resolving pronominal anaphora, there is no standard state-of-the-art algorithms to undertake the coreference resolution task. Rule-based approaches [12, 13] focused on syntax-based information such as number/gender agreement features, and the popular algorithms based on this kind of approach

Table 1. Example Sentence of Anaphora Representing Persons.

Sentence 1
Raman paLLikku sendRaan. (transliterated Tamil sentence)
Raman to school went. (word-level translation)
Raman went to school. (equivalent English sentence)
Sentence 2
avan thervu ezuthinaan. (transliterated Tamil sentence)
He the exam wrote. (word-level translation)
He wrote the exam. (equivalent English sentence)

are centering [9], Hobbs [10], etc. used to resolve pronominal anaphora. However, the original centering theory has been modified by incorporating the semantics for resolving various types of anaphora, in which the single-term antecedents (as shown in Example 1, "Raman" is a single-term antecedent) have been identified [3].

Machine learning approaches [15, 19] seem to be a promising way to overcome the limitations of the rule-based approaches [3], which might fail to capture global patterns of coreference relations as they occur in test data. Generally, learning-based approaches divide the task of coreference resolution into two subtasks, namely, classification and clustering. A classifier is trained on an English Wikipedia corpus to learn the probability that a pair of NPs can possibly be coreferents. The sentences are parsed, and the clustering is performed. Sentences that had no parsed output are ignored. The clustering merges these pairwise links to form distinct coreference chains. Filtering of non-coreferents is performed using the number/gender features and sentence distance [11].

The motivation behind the graph-based bootstrapping in the coreference resolution task is manifold. First, the previous machine learning approaches are highly dependent on the syntactic structure and dependency path information. Second, not all the coreferents of an entity satisfy the number/gender agreement property. Third, the acquisition of large-size standard training data set proves to be immensely challenging for electronically resource-constrained languages such as Tamil. Fourth, coreferring entities can indicate places. An example for the coreferring entities representing "place" is described in Section 4.5.1.

This article focuses on the classification of coreferring entities using a semisupervised pattern-based bootstrapping approach. We assume that the natural language text has already been preprocessed and represented as semantic graphs. This bootstrapping procedure involves the selection of features for the identification of coreferring mentions, and the filtering of non-coreferent and duplicate mentions, to obtain the final list of coreferring entities.

In the earlier work of Balaji et al., a similar pattern-based semisupervised bootstrapping approach has been used for resolving different types of anaphora. It is extended by exploring the various features associated with semantic graphs [6], where a new set of features is introduced to the coreference resolution task. In contrast to the pronoun types acting as cues for the selection of referring expressions [3], the selection of mentions cannot be restricted or narrowed down by cues, and thus, it is difficult to identify the mentions of an entity using cue words.

1.1 Semantic Graph Representation

In this article, we use semantic graphs as input for the coreference resolution. In this approach, we use the universal networking language (UNL) [20] for semantic graph representation of natural language text, which will be described in Section 4.5.2. An existing approach proposed by Balaji et al., enconverted the Tamil sentences into UNL representation using a rule-based approach [4]. These semantic graphs are utilized in building the index for semantic search engine. This approach is tested with 600,000 documents of tourism and 200,000 documents of news domain. In this work, the UNL list (created manually) is utilized for obtaining the concepts of a natural language word. A set of rules has been defined to identify the relationship between the concepts in a sentence, as well as the attributes associated with each concept, and to create the directed graph representation. In the proposed approach, we use the semantic UNL graphs as the input for

the coreference resolution task. However, although we use the existing UNL semantic graphs [4], we focus on the use of the components available in the semantic UNL graphs such as universal word (UW) concepts (represents the English translation of a natural language word and the UNL semantic constraint), relations (46 semantic relations), and attributes (represents mood, tense, aspect, etc.) for coreference resolution. Any semantic representation having similar features could also have been used for identifying the mentions of an entity. The components of the semantic graphs that are used by the proposed bootstrapping approach have been discussed in the following sections.

The rest of the article is organized as follows. Section 2 discusses the related works on coreference resolution. Section 3 describes the semisupervised learning bootstrapping procedure, which includes features, pattern representation, and different stages of bootstrapping in the identification of mentions and filtering of non-coreferents and duplicate mentions. In Section 4, the performance of the proposed approach is discussed and compared with a word association-based bootstrapping system [11]. Finally, future enhancements of the proposed approach are outlined.

2 Related Work

In this section, we will discuss various machine learning approaches that have been carried out for the coreference resolution task. In general, bootstrapping is a learning task that starts with a small set of labeled data and a large set of unlabeled data. The labeling of instances is carried out through matching iteratively until no more new patterns are generated and no instances are available in the test data [1].

The mention pair model proposed by Soon et al. [19] classifies links (pairs of two mentions) as coreferent or disreferent, followed by a clustering stage in which link decisions have been made by distinguishing the entities. Muller et al. [14] performed coreference resolution using a cotraining algorithm, which puts the features into disjoint subsets when learning from labeled and unlabeled data. Bergsma and Lin [7] presented a bootstrapping approach to identify coreference entities based on the syntactic paths and word associations. The dependency paths and node sequence have been used to identify the coreferent path between entities in the parse tree. Moreover, gender/number information and semantic compatibility have been determined using the corpus-based probability information.

Instead of using syntactic structures and dependency paths, the proposed approach uses rich semantic features such as semantic relations to identify the coreferring entities. In addition, the proposed approach uses scoring functions for the selection of mentions. Another bootstrapping procedure proposed by Kobdani et al. [11] for coreference resolution uses the word association information to identify mentions and labeled using a self-trained approach. Filtering of non-coreferents are based on the sentence distance with a fixed boundary of distance 3, number/gender agreement features [11]. However, the complex structures of noun phrases could not be identified. Ponzetto and Strube [16] presented a machine learning approach for the coreference resolution system using semantic roles that use the syntactic parser to identify the verb predicates of the mentions.

The selection of features plays a major role in identifying the coreferents of an entity. Recasens and Hovy [17] presented 47 features for the coreference resolution task. However, the distance measure between the entity and mention is restricted to five sentences, and moreover, the mentions that satisfy only the agreement features are extracted. The mentions beyond the threshold limit are not captured during the identification process.

The above-discussed approaches used the morphosyntactic features and dependency paths to define a pattern and concentrated only on the number/gender agreement features. However, with only agreement features, most systems failed to identify the correct mentions (without gender information) of an entity. Instead, in the proposed approach, morphosemantic features are used to define a pattern, and independent of the agreement features, the coreferents without the number/gender information are identified using the context-based semantics. Moreover, in this work, mentions of complex structures are therefore identified using context-based features. The scoring schemes are introduced for the selection and filtering of mentions among the possible mentions identified. These steps take place in two stages of bootstrapping, which will be described in detail in Section 4.

3 Universal Networking Language

UNL [20] is a deep semantic representation consisting of UWs, relations, and attributes. The UWs are used to represent the words in a sentence. The UNL relations are used to represent the relationship exist between two different concepts in a sentence. Attributes are used to represent the mood, tense, aspect, etc. The UWs are assigned to a natural language word and are represented using semantic constraints obtained from UNL knowledge base (UNLKB) [21]. The expression of UW is <*headword*(*Semantic constraints*)>. For example, *India*(*icl*>*country*), where *icl* stands for the restriction defining the semantic class where the UW is included (i.e., the UW is included in the semantic class *country*).

The UNL representation is said to be a hypergraph, when it consists of several interlinked or subordinate subgraphs. These subgraphs are represented as hypernodes and correspond to the concept of dependent (subordinate) clauses and a predicate. They are used to define the boundaries between complex semantic entities being represented. A scope is a group of relations between nodes that behave as a single semantic entity in a UNL graph. In the example sentence, "John killed Mary when Peter arrived", the dependent clause "when Peter arrived" describes the argument of a time relation and, therefore, should be represented as a hyper node (i.e., as a subgraph) as represented in Figure 1.

The UNL ontology [22] is a semantic network consisting of the UW system, co-occurrence relations between UWs, and the definition of UWs. The UNL ontology provides the linguistic and semantic information between the concepts. The inheritance property in the UNL ontology infers the relations between the lower UWs with their higher UWs.

A technique called vector symbolic architectures (VSAs) are a class of connectionist models for the representation and manipulation of compositional structures that can be used to model high-level cognitive functions. VSA is a distribution representation model that implements the binding and bundling support required to recover filler of roles in an application [8]. However, the effectiveness of VSA is the assigning of meaning by binding and bundling to the combination of related phrases from complete document corpus.

The focus of this work is limited to traditional coreference processing indicated by word- and context-specific features and does not need the capability of inferencing provided by VSA. Therefore, in this work, we consider the UNL representation as the basic semantic framework on which we build the NLP processing of sentences. In case of coreference, we use the word-based features of UNL such as semantic constraints, attributes, and verb categorization.

4 Bootstrapping for Coreference Resolution

In general, pattern-based bootstrapping is a task of inducing a classifier with a small set of labeled data and a large set of unlabeled data. First, a set of example patterns (sequence of features that occur frequently in a corpus is represented as a pattern), considered to be the preferred confident patterns, is extracted from the training corpus. The patterns are then applied over the test corpus to extract similar instances through exact

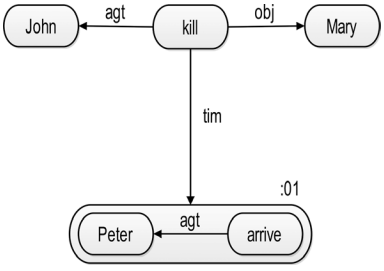

Figure 1. Example UNL Semantic Graph of the Sentence "John killed Mary when Peter arrived".

matching and partial matching. This iterative process continues until all instances in the test corpus has been handled or no new patterns are generated while learning [1].

Figure 2 illustrates the two stage pattern-based bootstrapping procedure for the coreference resolution task. During Stage 1, the possible mentions of an entity are identified, and during Stage 2, the non-coreferents and duplicate mentions are filtered, and the correct mentions are identified by applying the scoring schemes described in Section 4.4. This process continues iteratively to learn new patterns from the test data until no more new patterns are obtained and no input instances are available.

In this first stage, we consider an entity and use word-based features (will be described later) to obtain all possible mentions. However, at this stage, it is not possible to obtain the correct mentions of an entity because the association of the mentions with an entity can be based on matching the additional verb- and semantic-based features and on the frequency with which the mentions and/or their features co-occur with the original entity. Figure 2 shows the bootstrapping flow diagram for the coreference resolution task. The proposed approach has several advantages over the existing systems, which are the following:

(i) identifies the coreferents of an entity within a document of any distance (without any limitation in the sentence distance)
(ii) identifies the coreferents of an entity that has no number/gender agreement information
(iii) identifies the coreferents of an entity representing places (described in Section 4.5.1)
(iv) identifies mentions of complex noun phrases

The input to our bootstrapping approach is a fully connected semantic graph (i.e., pronominal anaphora resolved and connected). The semantic graphs are constructed using an existing rule-based approach [4]. The information associated with nodes and edges are extracted as features. Each node contains a UW concept and its attributes, and each edge represents the UNL semantic relation that exist between two different concepts. These extracted features are then represented as patterns for the bootstrapping procedure. The example patterns obtained are then matched with the unlabeled data to extract possible mentions of an entity at Stage 1, whereas appropriate scoring schemas are used to filter the non-coreferent mentions and duplicate mentions at Stage 2 of the bootstrapping procedure.

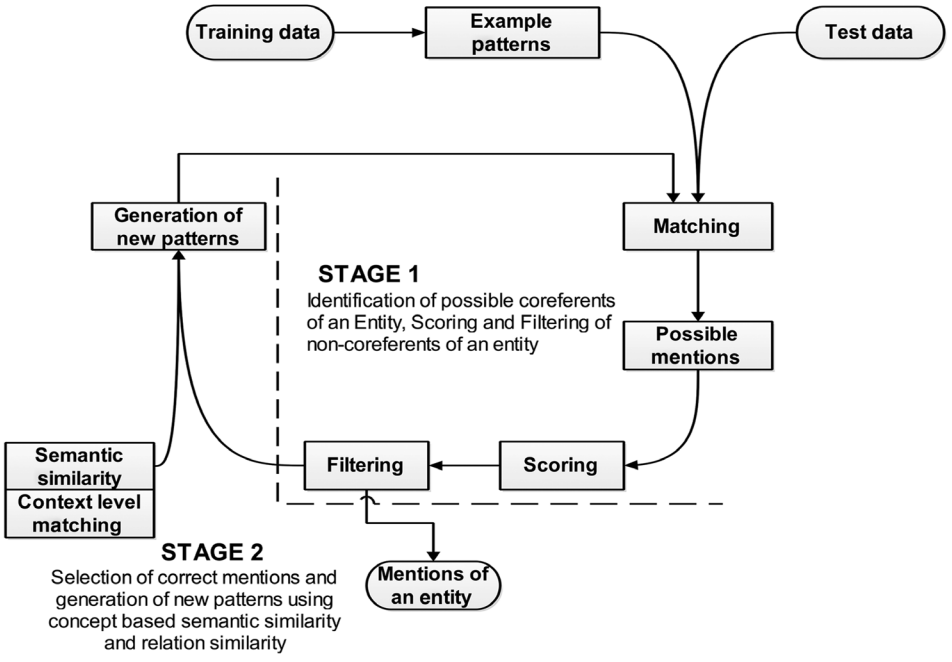

Figure 2. Bootstrapping for Coreference Resolution.

The preprocessed information such as the POS tag and semantic constraints are stored in the nodes while building the semantic graphs [4]. Moreover, nested graphs [5] have also been built along with the simple semantic graph construction, in which the complex noun phrases can be possible mentions of an entity. Thus, the information discussed above is extracted from the semantic graphs to define the pattern. This approach uses the semantic graph in which the anaphora resolution has already been performed [6] and incorporated with the semantic graphs. The anaphoric nodes are connected together to form a single large semantic network. However, some concepts in the graph do not have any connection with the other concepts but can still be possible mentions of unconnected entities.

To deal with such cases, we use various features such as sentence identifier, synonyms, distance measure, POS information, frequency of occurrence (which we will discuss in the following sections in detail) to connect dangling concepts in the semantic graph.

4.1 Identification of Features

The coreference resolution task requires a diverse set of features to determine the appropriate mentions of the corresponding entity. Recasens and Hovy [17] described the linguistic issues behind the selection of a set of 47 features, which were classified into classical, language-specific, corpus-based, and novel features. In this approach, we have used features such as concepts, attributes, and relations between the concepts available in the semantic graphs. Among the features we described, some are similar to those of Recasens and Hovy [17]. However, we used additional features to identify the correct mentions of an entity. The mentions of an entity can consist of single or multiple concepts. Both mention types are identified using various features described in Table 2.

In this article, we introduce a new set of features for correct mention identification. One such feature is the type of verb by which the possible mentions can be identified. From the linguistic analysis of a Tamil text, we find that some types of verbs such as stative verbs and transitive verbs are helpful in identifying the coreferents of an entity. These types are identified using the semantic constraints such as *aoj>thing* for stative verbs and *agt>thing*, *obj>thing* for transitive verbs, which are associated with a word. Here, *aoj*, *agt*, and *obj* are UNL semantic relations [23]. Another new set of features is the context-based features such as UNL

Table 2. Features Used for Coreference Resolution.

Word-based features	
POS tag of a word (POS)	To identify the mention is a valid possible coreferent
Semantic constraint associated with a word (SC)	To identify the semantic classes such as the person, organization, location
Attributes associated with a word (Attr)	Used to represent number, gender, aspect, mood, etc.
Verb type (Verb)	To decide whether the mentions belong to the corresponding entity
Context-based features	
Semantic relations associated with a word (Relation)	Semantic relations of interest take part in the identification of mentions
Nested graph identifier (NG id)	Identifier to find the mentions of complex structures
Classical features	
Sentence frequency	Number of sentences where the mentions occur
Concept frequency	Frequency of the occurrence of a concept in a document
Statistical features	
Word mention distance	The distance between the words and mentions
Sentence distance	The distance between the entity occurred sentence and its mentions occurred sentences
Frequency of mentions and mention pairs	Frequency of mentions and its pairs
Probabilistic scoring for filtering	Scoring function to filter out the non-coreferent and duplicate mentions

semantic relations, which are used in the identification of the correct mentions. To identify the complex noun phrases, we introduce a new feature called the nested graph identifier, which is used to identify mentions of complex noun phrases.

4.2 Pattern Representation

Using the features discussed above, the pattern needed for coreference resolution is defined. The pattern is defined as $<Mention_i$, Entity, Verb>, where $i=1, 2, 3, \ldots, N$, representing any number of mentions. $Mention_i$ can be individual concepts or a group of concepts. The group of concepts is represented as a subgraph connected with the semantic relations. The entity tuple defined here is the original referent of the mentions. This tuple can be a noun, noun phrase, or entity. The features associated with each component of the pattern are defined as follows.

$$<[POS+SC+Attr+Relation]_i,$$
$$[POS+SC+Attr+RelationPOS+SC+Attr+Relation]_i+NG\_id,$$
$$Entity+SC+Attr+Relation, Verb+SC+Attr>$$

where $i=1, 2, 3, \ldots, N$.

The value i represents any number of coreferents and/or subgraphs for an entity in the pattern. The tuples defined in the pattern are described in Table 2. $Mention_i$ is tagged with the word-based features and the semantic relation connected with that mention if the mention is a single-term candidate. $Mention_i$ is tagged with the context-based features such as the nested graph identifier along with the word-based features if the mention is a group of concepts. An example sentence and its semantic graph are shown in Figure 3.

The example given in Figure 3 shows the features extracted from each component for representing the pattern. In addition to the above-described features, $Mention_i$ and the entity of a pattern are tagged using the

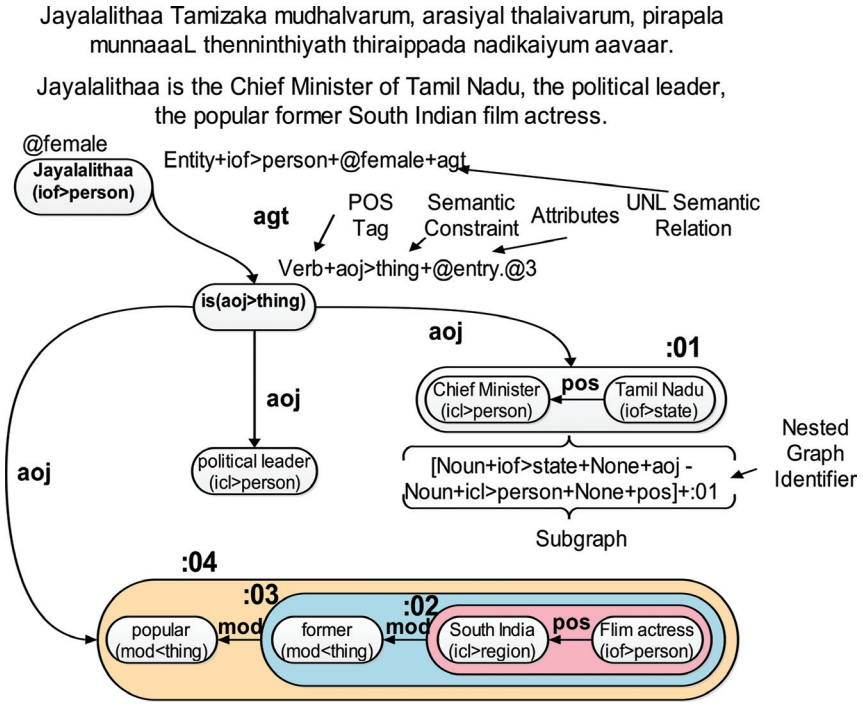

Figure 3. Example Semantic Graph.

classical features to obtain the mention pair easily. These feature-tagged patterns are then given for matching to select the possible mentions.

4.2.1 Word-Based Features

Word-based features consist of language-specific features such as the number/gender agreement features that are utilized in the identification of coreferential relations that occur in a language. Recasens and Hovy [17] described language-specific features such as elliptical subjects, grammatical gender, and nouns in detail to identify the coreferential relations for the Spanish language. Moreover, the semantic information is achieved using the WordNet to obtain the synonyms. Instead of using the features that are language dependent, we represent the various information using the attributes (lexical, syntax, and semantics available in UNL) associated with the concepts in the semantic graphs. As discussed earlier, the semantic graphs and their associated information are obtained from a language-independent generic semantic representation.

Furthermore, the proposed approach uses the UNL ontology, an external knowledge source [22], and a manually created UNL list. The UNL list is utilized to obtain the synonyms of a natural language word, and thus, the list contains a natural language word, translated or transliterated word, and the UNL semantic constraints obtained from the UNLKB [20]. The UNL list is created using a semisupervised technique, and the entries are manually corrected by linguistic experts. The list contains around 100,000 unique entries. The UNL ontology is used for measuring the similarity between the semantic constraints. The UNL ontology is an abstraction of semantic constraints arranged in hierarchical relations [22]. This UNL ontology is utilized in the determination of the semantic classes of the mentions. As discussed earlier, the type of verb is determined based on the semantic constraints through which the mentions are identified for a specific entity.

4.2.2 Context-Based Features

The context-based features include the semantic relations that can take part in the identification of the correct mentions of an entity. Recasens and Hovy [17] concentrated on the modifier relations of the mentions in the identification of the correct mentions. Similarly, Soon et al. [19] focused on the modifier and possessor relations of noun phrases to identify the mentions. However, in this approach, we investigated contextual features, such as the semantic relations of different types, including participant, attributive, adjunct, time, and location relations for identifying the mentions of an entity. Balaji et al. [5] broadly classified the relations into coordinating and subordinating relations based on the types for rule-based anaphora resolution, and later the authors generalized the above classification for resolving various anaphora using bootstrapping [3]. However, the classification of the relations is specific to the type of anaphora to be resolved, where the relations are identified based on the cue words or phrases indicating the type of anaphora and the triggering tuples under consideration. The approach described in this work is one where the mentions cannot be determined by the cue words or phrases, and instead, the mentions of an entity can be determined using the semantic relations that exist between the concepts. The detailed analysis of the context-based features is described in Section 4.5.2

4.2.3 Statistical Features

This set of features is utilized at both the selection and filtering stages. The features considered here are distance measures such as word distance, sentence distance, and mention distance; frequency of the occurrence of mentions and mention pairs; scoring to filter duplicate mentions; and probabilistic computation

on the identification of mentions. The probability is based on the graph properties such as the number of edges (relations) directly connected to a node (mention), number of nodes (possible mentions) directly connected with the same edge (relation), and the number of nodes (mentions) connected with indirect links (relations). The direct and indirect links are decided based on the dependent UNL relations described in Section 4.5.2, which is similar to the coordinating and subordinating relations [3, 6]. Although some of the measures have been described by Recasens and Hovy [17], the proposed approach introduces new scoring functions based on word-based features, such as entities, mentions, type of verb connecting entities and mentions, and context-based features such as the relations existing between the entities and mentions to filter out the non-coreferents.

4.3 Selection of Possible Mentions

We use a set of features defined in the pattern in the selection of possible mentions. Using the features defined in Table 2, the mentions that can be single entities and/or complex noun phrases are extracted. The possible mentions are identified using the word-based features represented in the pattern. In addition, the context-based features such as the nested graph identifier are also used for extracting the complex noun phrases that can be possible mentions. Among the set of possible mentions obtained during Stage 1, the exact mentions of an entity are extracted from the set using filtering. The set contains coreferents of an entity and non-coreferents that do not corefer with the corresponding entity. The task during Stage 2 is to identify the correct mentions of an entity by eliminating the non-coreferents through filtering. Filtering can be performed, using a frequency-based scoring of mention pairs and context-based scoring for near matching.

4.4 Filtering

The task of filtering is to remove the non-coreferent mentions and the duplicate mentions of an entity. The set of possible mentions are identified during Stage 1. However, not all mentions identified during Stage 1 are exact mentions of an entity because some mentions in the set may corefer to a different entity and some can be duplicate mentions. To find the exact mentions of an entity, filtering is required. In our approach, we use specific features of the semantic graph such as the concepts (including verbs and their semantic constraints) and relations connecting the concepts to obtain the correct mentions of an entity. Filtering is performed based on the statistical measures and measures based on classical features such as the frequency of the occurrence of mentions and graph-based features (e.g., the distance of mention pairs, probabilistic computation based on connected nodes, mentions, and relations of interest for mentions identification). Here we perform two levels of filtering. The filtering score $Filter1(M_{E_S})$ is based on relations, whereas $Filter2(M_{E_S})$ is based on semantic constraints of verbs. The first filter score is given in Equation (1).

$$Filter1(M_{E_S}) = \frac{N((M_i, E_S), R_j)}{N(M_i) \cdot N(E_S)}, \tag{1}$$

where M_{E_S} is the set of mentions for a specific entity, M_i is the number of mentions (where $i = 1, 2, 3, \ldots, N$), E_S is the specific entity, R_j is the semantic relation existing between entity and mention directly/indirectly (where $j = 1, 2, 3, \ldots, M$), and N is the number of occurrences.

According to Equation (1), $Filter2(M_{E_S})$ gives the filter score for the mention for the entity E_S and is defined as the ratio of the number of times the particular mention M_i occurs for the specific entity E_S in the context of the relation R_j and the number of times mention M_i and number of times entity E_S occur independent of each other. This filtering score is used for first-level filtering of non-referents from among the set of mentions. However, even after this filtering, some non-referents may still be present.

The next scoring function estimates the mentions to be the correct referent of an entity for such cases. This scoring uses additional features such as the semantic aspects of the verb to decide which entity the

mention belongs to if more than one entity refers to a common mention. The second filtering score is given in Equation (2).

$$Filter2(M_{E_S}) = \frac{N((M_i, E_S), V_{M_{E_S}})}{N(M_i) \cdot N(E_S)}, \tag{2}$$

where M_{E_S} is the set of mentions for a specific entity, M_i is the number of mentions (where $i = 1, 2, 3, \ldots, N$), E_S is the specific entity, $V_{M_{E_S}}$ is the verb match for the specific entity and mention, and N is the number of occurrences.

According to Equation (2), $Filter2(M_{E_S})$ gives the filter score for the mention for the entity E_S and is defined as the ratio of the number of times the particular mention M_i occurs for the specific entity E_S in the context of the verb $V_{M_{E_S}}$ and the number of times mention M_i and number of times entity E_S occur independent of each other. This filtering score is used for first-level filtering of non-referents from among the set of mentions. Exact matching and filtering yields mentions of entities. However, there are some instances that have still not been tackled for which we need to generate new patterns.

Analysis of the sentences of a Tamil text shows that the mentions can occur anywhere in the document and do not necessarily occur with the corresponding entity. Moreover, the gender agreement feature failed to identify the mentions in some cases because the gender information of mentions need not exactly match with the gender of the entity.

An example of a Tamil document shown in Table 3, taken from Wikipedia [24], shows the failure of these two cases. For the purpose of understanding, only the sentences in which the mention occurred are

Table 3. Example 2: A Sample Wikipedia Document Contains Mentions of an Entity M.G.R at Different Sentences.

Sentence 1 ***m.g.r*** endra peyaril pugaz peRRa, **maruthur gopalamenon ramachandran (m.g. ramachandran) tamilth thiraippada nadikar**aakavum 1977 mudhal iRakkum varai **tamilnaattin mudhalamaichchar**aakavum irunthavar. (In the name of **M.G.R**, a famous **maruthur gopalamenon ramachandran (M.G. Ramachandran)**, the **Tamil film actor and Chief Minister of Tamil Nadu** from 1977 upto his death.)
Sentence 9 ***m.g.r*** **thiraippada iyakkunar**um **thayaarippaalar**umaavaar. (**M.G.R** is a **film director** and **producer.**)
Sentence 31 ***m.g.r*** oru **munnaNith tamilth thesiyavaathi**yaagavum, **diraavida munnetrak kazakaththin mukkiya uruppinar**aagavum thikaznthaar. (**M.G.R** was a **leading Tamil nationalists**, and was a **leading member of the Dravida Munnetra Kazhagam.**)
Sentence 75 thiraisevaikkaana pattangal **idhyakkani, puratchi nadikar, nadika mannan, makkal nadikar, palkalai venthar, makkal kalainjar, kalaiarasar**, etc. (Awards for Screen service: Sweetheart, revolution actor, cast King, people actor, Chancellor of the University, People's Artist, Art King.)
Sentence 80 pothusevaikkaana pattangal **koduththu sivantha karam, kaliyugak kadavul, niruththiya chakkaravarthi, ponmanach chemmal, makkal thilagam, idhaya deivam, puratchith thalaivar** etc. (Awards for public service: The red hand, Kaliyuka God, nritya Emperor, ponmana Semmal, people tilak, heart deity, revolution Chairman.)
✻ – Entity
✻ – Correct mentions of an entity
✻ – Mentions of an entity having common gender information
✻ – Mentions that change over time for an entity
✻ – Mentions without gender information

represented with the sentence identifiers and its equivalent English sentences are shown in parentheses below each Tamil sentence. (The entity and its mentions are represented in boldface.) Some words in Sentence 80 do not have their equivalent English translation, and thus, the transliteration is provided.

The example given in Table 3 shows that the sentences in the document have an entity "M.G.R" with different mentions. It is to be noted that the mentions (given in boldface) in sentences 1, 9, and 31 occur with the corresponding entity, and therefore, they can be easily identified. In contrast, sentences 75 and 80 do not have the corresponding entity, and some mentions do not have the gender information. From this observation, it is clear that the mentions could not occur within the boundary limit of the sentences and the gender information could not completely take part in filtering out the non-coreferents of an entity. The proposed approach uses features of the semantic graph, such as the concepts and the relations connecting the concepts to obtain the correct mentions of an entity described in Example 2. The concepts and the relations between the concepts are considered for filtering to extract the correct mentions of an entity.

4.5 Generation of New Patterns

After matching and filtering are performed, the mentions of an entity that are not identified under exact matching are given for partial matching. Partial matching is carried out by modifying the word- and context-based features. The semantic constraint tuples represented in a pattern are masked, and a search is done for similar instances in the test data. The instances obtained through masking are then given for similarity matching, in which the similarity between the masked semantic tuples is achieved using the UNL ontology [22] and is measured using a similarity scoring. The semantic similarity has been described in the existing approach for anaphora resolution [6]. Similarly, the semantic constraints of mentions are obtained by measuring the semantic similarity between the example patterns of the mentions and the possible mentions in the test data. The similarity of the semantic constraints is achieved using the semantic UNL ontology in which the semantic constraints are arranged in hierarchical relations such as "is a" and "instance of". Partial matching is then carried out using context-level features such as the UNL semantic relations available in the UNL-based semantic graphs. The semantic relations are tagged along with the other tuples in the pattern. The detailed analysis of the semantic similarity of constraints and dependent UNL relations is described below.

4.5.1 Semantic Similarity of Constraints

The semantic similarity between constraints is useful in identifying the mentions representing places. For example, *Madurai*, *koodalnagar*, and *thoongaa nagaram* are coreferents in which the first two entities are singleton candidates and *thoongaa nagaram* is formed as a subgraph that is connected with the modifier relation. The semantic constraints of the entities *Madurai* and *koodalnagar* is *iof>city*, and the headword *nagaram* in *thoongaa nagaram* is *icl>city*. The semantic constraints of all the three entities have the same parent *icl>place*, and thus, they are considered as semantically similar entities. This is achieved using the semantic abstraction of the semantic constraints, which are available through the semantic UNL ontology [22]. The subgraph is identified using the subgraph identifier and marked as the coreferent of the entity. Semantic similarity is measured by the distance between the parent semantic constraints in UNL ontology and is given in Equation (3).

$$SIM(M_i, M_j) = DIST(M_{Parent}(M_i, M_j)), \tag{3}$$

where M_i is the UNL semantic constraint of mentions in the input instance, M_j is the UNL semantic constraint of mention in an example pattern, M_{Parent} is the parent UNL Semantic constraint in UNL ontology, and DIST is the distance function.

Next, we will discuss the handling of context-level semantics such as the UNL semantic relations necessary to obtain the correct mentions.

4.5.2 Dependent UNL Relations

Certain UNL relations connected with entities and/or noun phrases participate in the determination of mentions along with the type of verb in a sentence. From the analysis of semantic graphs, we can find different conditions where an entity or noun phrase with connected UNL relations are used to determine the mentions. However, Balaji et al. [3] proposed a set of coordinating and subordinating relations obtained from semantic graphs in resolving various anaphors; we focus on the identification of the correct mentions using UNL semantic relations. The rules are based on the semantic relations connected with pronouns and the semantic relations connected with the possible referring expressions. In the same vein, the rules are defined based on the semantic relations connected with entities and those connected with possible coreferents, either directly or indirectly (i.e., an entity and the possible coreferents are directly connected with semantic relations or connected via verbs). These conditions are utilized during partial matching in which all the conditions defined below are given for matching to obtain the correct mentions of an entity. The following are the conditions used in the identification of mentions.

(i) If an entity or a noun phrase, represented as (M_i), forms a participant relation (*PR*) with a stative verb (*SV*) and another single/multiple entities or noun phrases, represented as (M_j), of the same sentence form attribute relations (*AR*) with a stative verb (*SV*), then the entities and/or noun phrases connected with the participant and attributive relations are represented as coreferents. Thus, the condition is represented in a formal notation as given in Equation (4).

$$PR(M_i,SV) \wedge AR(M_j,SV) \Rightarrow Corefer(M_i,M_j), \text{ where } i,j=1, 2, 3, \ldots, N. \tag{4}$$

(ii) If an entity or a noun phrase, represented as (M_i), forms a participant relation (*PR*) with a transitive verb (*TV*) and another single/multiple entities or noun phrases, represented as (M_j), of the same sentence form adjunct manner relations (*MR*) with a transitive verb (*TV*), then the entities and/or noun phrases connected with the participant and adjunct manner relations are represented as coreferents. This is given in Equation (5).

$$PR(M_i,TV) \wedge MR(M_j,TV) \Rightarrow Corefer(M_i,M_j), \text{ where } i,j=1, 2, 3, \ldots, n. \tag{5}$$

(iii) If an entity or a noun phrase, represented as (M_i), forms the participant relation (*PR*) with a transitive verb (*TV*) and more than one entity or noun phrase, represented as (M_j), of the same sentence form specifier relations (*SR*) with a transitive verb (*TV*), then the entities and/or noun phrases connected with specifier relations are represented as coreferents. Equation (6) describes the above condition.

$$PR(M_i,TV) \wedge SR(M_j,TV) \Rightarrow Corefer(M_j), \text{ where } j<1. \tag{6}$$

(iv) If an entity or a noun phrase, represented as (M_i), forms an adjunct location relation (*LR*) with a transitive verb (*TR*) and another single/multiple entities or noun phrases, represented as (M_j), of the same sentence form participant relations (*PR*) with a transitive verb (*TR)*, then the entities and/or noun phrases connected with the adjunct location and participant relations are represented as coreferents. This is defined in Equation (7).

$$LR(M_i,TV) \wedge PR(M_j,TV) \Rightarrow Corefer(M_i,M_j), \text{ where } i,j=1, 2, 3, \ldots, n. \tag{7}$$

Using the conditions described above, the dependent UNL relations are identified to narrow down the correct mentions at Stage 2 among the possible mentions obtained at Stage 1, respectively. In addition, a new combination of relations is learned from the above conditions, and thus, the correct mention of an entity is obtained.

5 Evaluation

The evaluation of the proposed system is twofold: one is to evaluate the performance of a coreference resolution task which is performed using different measures such as B CUBE (B^3) [2] and bilateral assessment of

noun phrase coreference (BLANC) [18]; the other is the investigation of the bootstrapping approach, which is evaluated based on the number of iterations and the number of new patterns generated. We also compared the proposed approach with the bootstrapping approach presented by Kobdani et al. [11], which is considered as the baseline system.

The number of mentions obtained is investigated by comparing the true mentions (i.e., mentions that actually occur for an entity in a corpus) and the extracted mentions (i.e., mentions obtained for an entity by the suggested bootstrapping approach). The measures B^3 is expressed in terms of precision (P) and recall (R) and F measure. Thus, the F measure is defined as the harmonic mean between precision and recall and is given in Equation (8).

$$F \text{ measure} = \frac{(2 \cdot P \cdot R)}{P+R} \tag{8}$$

The B^3 measure is used to determine the precision and recall of each mention and then compute the weighted average of the individual precision and recall of each mention. For a mention M_i, the individual precision is determined by the number of mentions obtained by the proposed bootstrapping approach that actually corefer to an entity among the extracted mentions. Therefore, the precision for a given mention M_i is given in Equation (9).

$$Precision(M_i)_{B^3} = \frac{|E_{M_i} \cap T_{M_i}|}{|E_{M_i}|}. \tag{9}$$

The individual recall is determined by the correct mentions of an entity, among the total true mentions of an entity in the corpus. Therefore, the recall is defined in Equation (10).

$$Recall(M_i)_{B^3} = \frac{|E_{M_i} \cap T_{M_i}|}{|T_{M_i}|}, \tag{10}$$

where $E_{(M_i)}$ is the extracted mentions of an entity and $T_{(M_i)}$ is the true mentions of an entity actually present in the corpus.

In contrast to the B^3 measure, BLANC considers both the coreferents and non-coreferents of an entity. Precision and recall are computed separately for the coreferents and non-coreferents of an entity. Thus, precision and recall are defined in Equations (11) and (12), respectively.

$$Precision_{\text{BLANC}} = \frac{R_c}{2(R_c + W_c)} + \frac{R_n}{2(R_n + W_n)} \tag{11}$$

and

$$Recall_{\text{BLANC}} = \frac{R_c}{2(R_c + W_n)} + \frac{R_n}{2(R_n + W_c)}, \tag{12}$$

where R_c is the number of correct coreferents of an entity, W_c is the number of incorrect coreferents of an entity, R_n is the number of correct non-coreferents of an entity, and W_n is the number of incorrect non-coreferents of an entity.

Table 4. B^3 and BLANC Measures.

Measures	Precision	Recall	*F* measure
B^3	81.62	71.96	76.48
BLANC	75	68	70.7

Table 4 shows the performance of the coreference resolution based on the B^3 and BLANC measures.

The *F* measure is high for B^3 when compared with the BLANC measure. This is because some mentions can refer to more than one entity, and they can also change over time. In this article, we have not focused on the mentions that can change over a time for a single entity. Thus, the BLANC measure is low when compared with the B^3 measure. Moreover, the B^3 measure is computed for each entity and does not focus on the true negative- and false-positive mentions as computed for BLANC. The mentions of an entity can be classified into three types – mentions common to more than one entity (e.g., Tom Cruise, Jake Gyllenhaal – "actor"), mentions specific to an entity (e.g., Madurai – "the city of temples"), and mentions that can change over time (i.e., temporal coreference – single mention can refer to more than one entity based on time information) (e.g., M. Karunanidhi – "former chief minister").

Among these classifications of mentions, we focus on mentions of an entity other than the temporal coreferents. Mentions falling under this category are valid only when they are specified along with the time information. However, the temporal type of mentions that can be specified by the time information has not been considered. This is the reason why the BLANC score of the suggested bootstrapping system is lower than the B^3 score.

As discussed earlier, Kobdani et al. [11] determined the coreferents using the number/gender agreement features for coreference resolution. However, in some cases, the number/gender agreement features do not act as a valid feature, which is discussed below and shown in Table 3. The evaluation measures are shown in Table 5.

There are certain cases in which the entities or noun phrases do not have number or gender information. As shown in Example 1, *ponmanach chemmal*, *kalaich chudar*, *ithayakkani*, etc. are some cases in which gender agreement does not exist, but these can corefer with an entity. The approach was investigated, with and without the number/gender agreement features, and the performance using the B^3 measure showed that there was an increase in the *F* measure associated with B^3 measure in the proposed system. Because the usage of only the agreement features resulted in false association, the use of semantic features and relations along with the semantic connection improved the performance of the coreference resolution task. From Table 5, it is clearly shown that without the number/gender agreement features, the *F* value for both increases up to 15%.

The approach was also tested without the semantic graph features and was evaluated with the rest of the features to observe the differences in the performance. In Table 5, the number/gender agreement features [11] are considered for coreference resolution, which produces an *F* value of 0.67. This *F* value is lowered when the context-based features are introduced in the proposed approach, and the agreement features are not considered.

Thus, the *F* measure is very low when the evaluation is carried out without semantic relations and nested graphs. The low *F* value for "without semantic relations and nested graphs" indicates that we do not consider the number/gender agreement word-based features. Moreover, some mentions do not occur along with the entity, and some are complex structures. Such cases are identified with rich semantic features associated with the mentions that are not connected with any other entities or concepts. The dependent UNL relations described in Section 4.5.2 take care of these issues and thus increase the performance from 0.28 to 0.73.

In the case of nested graphs, the complex noun phrases are difficult to identify if the length of the phrase is too large. Moreover, it is difficult to detect the boundary of the complex noun phrases, which are possible mentions. Without the nested graphs, the proposed bootstrapping approach can identify the complex mentions partially. The partially identified mentions cannot be valid and do not convey the complete meaning. Hence, those partial phrases are not considered as correct mentions. With the nested graphs, the boundary

Table 5. B^3 Measure for With and Without Context-Based Features.

Features	B^3		
	Precision	Recall	*F* measure
With number/gender agreement (Kobdani et al. [11])	0.77	0.6	0.67
Without semantic relations without nested graphs	0.33	0.25	0.28
With semantic relations without nested graphs	0.85	0.65	0.73
With semantic relations with nested graphs	0.88	0.68	0.77

detection of noun phrases is clearly represented using the nested graph identifier. The correct mentions are identified based on the headword of the complex noun phrases. Thus, it increases the performance from 0.73 to 0.77 when the nested graph is included.

Next, we have compared the proposed graph-based bootstrapping approach with the baseline system presented by Kobdani et al. [11]. Kobdani et al. use word association information for coreference resolution, in which an unsupervised self-training algorithm has been attempted. To evaluate this system's performance, we use the word association features alone for the identification of correct mentions of an entity using the proposed approach.

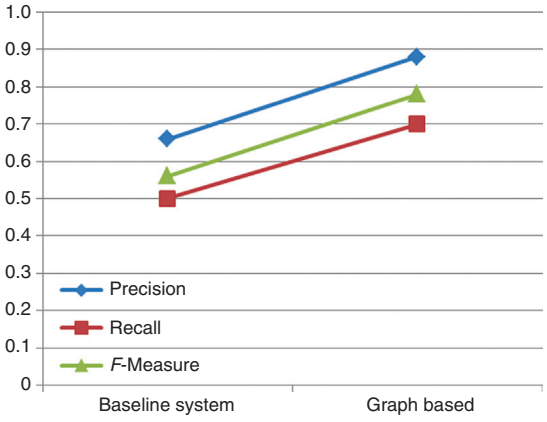

Figure 4. Comparison – Baseline System vs. Graph Bootstrapping.

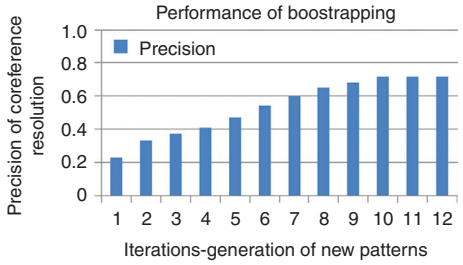

Figure 5. Performance of the Proposed Bootstrapping Approach.

Table 6. Number of New Patterns Generated at the End of Each Iteration.

Iterations	Number of new patterns generated
0	10
1	3
2	7
3	12
4	15
5	18
6	23
7	27
8	30
9	31
10	32
11	32
12	32

The performance of our bootstrapping approach is investigated using the tourism and news domain corpus. We have taken 5000 documents as the training data and extracted the most frequently occurring example patterns. Independent of the training data considered here, 10,000 documents from each domain are tagged with the appropriate features such as POS, UNL attributes, UNL semantic constraints, and UNL relations and used as the test data. From the results obtained, word association information alone does not identify the correct mentions of an entity completely. Instead, the features used by our approach identify the correct mentions of an entity, increasing the *F* measure from 56% to 78% (shown in Figure 4). This is because we use rich semantic features such as semantic relations and subgraphs (complex structure of noun phrases), which can be possible mentions. As discussed earlier, tackling and incorporating the temporal information into the coreference resolution task will increase the performance of the proposed bootstrapping approach.

The performance of the bootstrapping method for different iterations proposed in this work is shown in Figure 5. Initially, we start with a set of 10 example patterns (shown in Table 6 as 0^{th} iteration) obtained from the training data given for matching with the test data. The selection of these example patterns are decided based on the analysis that these patterns cover the most common examples in the test corpus. From this iterative procedure, we have obtained an average of 32 new patterns at different iterations. Figure 5 shows the precision for the generation of new patterns at different iterations.

Figure 5 shows the performance evaluation of the bootstrapping approach for 12 iterations. While performing partial matching, no new patterns were generated after the 10^{th} iteration. The results are shown for 12 iterations, and Table 6 clearly indicates that no new patterns were generated after the 10^{th} iteration. This is because, although partial matching may give rise to possible new patterns, mentions, being non-coreferents, are filtered. As already discussed, the performance of the proposed approach is further improved by including the temporal information along with the mentions.

Table 6 shows the number of new patterns generated at the end of each iteration. The number of new patterns generated are summed up and shown in the table. Initially, we start with 10 example patterns and produce three new patterns at the end of the first iteration. At the end of the second iteration, we obtained four new patterns, which are summed up with the previous iteration. Finally, at the end of the 10^{th} iteration, we have obtained 32 new patterns. The process continues until no more new patterns are generated and no input instances exist in the test data. Because iterations 8 and 9 generate only one new pattern, it continues for the next consecutive iterations.

While performing partial matching, no new patterns were generated after the 10^{th} iteration. Table 6 clearly indicates that no new patterns were generated after the 10^{th} iteration. This is because, although partial matching may give rise to possible new patterns, mentions, being non-coreferents, are filtered. As already discussed, the performance of the proposed approach is further improved by including the temporal information along with the mentions.

Table 6 shows the number of new patterns generated at the end of each iteration. The number of new patterns generated are summed and shown in the table. Initially, we start with 10 example patterns and produce three new patterns at the end of the first iteration. At the end of the second iteration, we obtained four new patterns, which are summed up with the previous iteration. Finally, at the end of the 10^{th} iteration, we have obtained 32 new patterns. The process continues until no more new patterns are generated and no input instances exist in the test data. Because iterations 8 and 9 generate only one new pattern, it continues for the next consecutive iterations.

6 Conclusion

In this article, a semisupervised graph-based two-stage bootstrapping approach has been described for the task of coreference resolution. The input to the proposed bootstrapping approach is the semantic graph, which is a directed acyclic graph representation. The information associated with the semantic graphs is extracted as features and represented as patterns. The patterns are then given to extract the possible mentions during Stage 1, and the exact mentions are identified by eliminating the non-coreferents in Stage 2. The

possible mentions are identified using the word-based features, and the non-coreferents are filtered using the verb match and the context-based features such as semantic relations, along with the scoring functions. The new patterns are generated using similarity matching of the semantic constraints and the graph level matching of the semantic relations. The rules for graph matching have also been discussed in detail. The proposed approach was investigated by evaluating the agreement features and context-based features. The proposed approach was evaluated against the baseline system described by Kobdani et al., wherein the agreement features and distance measures are considered for the identification of the correct mentions of an entity [11]. The use of the semantic graph-based features increases the performance of the proposed bootstrapping approach when compared with the word association-based bootstrapping system, where both distance and number/gender agreement features have been considered for mention identification. In future work, we will focus on tackling the mentions that can change over time for the same entity. We shall also explore the use of UNL attributes for improving the performance coreference resolution task.

Acknowledgments: We thank the Ministry of Communications and Information Technology, DEIT, New Delhi, for funding this project under the consortium for the development of Cross-Lingual Information Access.

Received June 29 2013; previously published online December 21, 2013.

Bibliography

[1] S. Abney, Understanding the Yarowsky algorithm, *Comput. Linguist.* **30** (2004), 365–395.
[2] A. Bagga and B. Baldwin, Algorithms for scoring coreference chains, in: *Proceedings of the Linguistic Coreference Workshop* at LREC '98, pp. 563–566, Granada, Spain, 1998.
[3] J. Balaji, T. V. Geetha, P. Ranjani and K. Madhan, Anaphora resolution using universal networking language, in: *Indian International Conference on Artificial Intelligence*, IICAI-2011, Banglore, India, 2011.
[4] J. Balaji, T. V. Geetha, P. Ranjani and K. Madhan, Morpho-semantic features for rule-based Tamil enconversion, *Int. J. Comput. Appl.* **26** (2011), 11–18.
[5] J. Balaji, T. V. Geetha and P. Ranjani, Semantic parsing of Tamil sentences, in: *Workshop on Machine Translation and Parsing Indian Languages*, MTPIL, COLING-2012, Mumbai, India, 2012.
[6] J. Balaji, T. V. Geetha and P. Ranjani, Two-stage bootstrapping for anaphora resolution, in: *24th International Conference on Computational Linguistics*, COLING-2012, Mumbai, India, 2012.
[7] S. Bergsma and D. Lin, Bootstrapping path-based pronoun resolution, in: *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, ACL-44, Sydney, Australia, pp. 33–40, 2006.
[8] R. Gayler, Vector symbolic architectures answer Jackendoffs challenges for cognitive neuroscience, in: P. Slezak (Ed.), *ICCS/ASCS International Conference on Cognitive Science*, pp. 133–138, University of New South Wales, Sydney, Australia, 2003.
[9] B. J. Grosz, S. Weinstein and A. K. Joshi, Centering: a framework for modeling the local coherence of discourse, *Comput. Linguist.* **21** (1995), 203–225.
[10] J. Hobbs, Resolving pronoun references, *Read. Nat. Lang. Process.* (1986), 339–352. ISBN:0-934613-11-7 (online version).
[11] H. Kobdani, H. Schueltze, M. Schiehlenand and H. Kamp, Bootstrapping coreference resolution using word associations, in: *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, June 19–24, 2011, Portland, OR, pp. 783–792, 2011.
[12] S. Lappin and H. J. Leass, An algorithm for pronominal anaphora resolution, *Comput. Linguist.* **20** (1994), 535–561.
[13] R. Mitkov, Robust pronoun resolution with limited knowledge, in: *Proceedings of ACL-COLING*, Morgan Kaufmann Publishers/ACL, pp. 869–875, 1998.
[14] C. Muller, S. Rapp and M. Strube, Applying co-training to reference resolution, in: *Association of Computational Linguistics*, ACL 02, pp. 352–359, 2002.
[15] V. Ng and C. Cardie, Improving machine learning approaches to coreference resolution, in: *Proceedings of Association of Computational Linguistics*, Association for Computational Linguistics, Philadelphia, Pennsylvania, pp. 104–111, 2002.
[16] S. P. Ponzetto and M. Strube, Semantic role labeling for coreference resolution, in: *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics*, EACL '06, Association for Computational Linguistics, Trento, Italy, pp, 143–146, 2006.

[17] M. Recasens and E. Hovy, A deeper look into features for coreference resolution, in: *Proceedings of the 7th Discourse Anaphora and Anaphor Resolution Colloquium on Anaphora Processing and Applications*, DAARC '09, Springer, Goa, India, pp. 29–42, 2009.
[18] M. Recasens and E. Hovy, BLANC: implementing the Rand index for coreference evaluation, *Nat. Lang. Eng.* **17** (2011), 485–510.
[19] W. M. Soon, H. T. Ng and D. C. Y. Lim, A machine learning approach to coreference resolution of noun phrases, *Comput. Linguist.* **27** (2001), 521–544.
[20] Universal networking language (UNL). http://www.undl.org/unlsys/unl/unl2005/.
[21] Universal networking language (UNL) knowledge base. http://www.unlweb.net/wiki/UNL_Knowledge_Base.
[22] Universal networking language (UNL) ontology (2000). http://www.undl.org/unlsys/uw/UNLOntology.html.
[23] Universal networking language (UNL) relations. http://www.undl.org/unlsys/unl/unl2005/Relation.htm.
[24] http://ta.wikipedia.org/wiki/%E0%AE%AE._%E0%AE%95%E0%AF%8B._%E0%AE%87%E0%AE%B0%E0%AE%BE%E0%AE%AE%E0%AE%9A%E0%AF%8D%E0%AE%9A%E0%AE%A8%E0%AF%8D%E0%AE%A4%E0%AE%BF%E0%AE%B0%E0%AE%A9%E0%AF%8D.