Journal of Emerging Technologies in Web Intelligence

ISSN 1798-0461

Volume 5, Number 4, November 2013

Contents

Theoretical Formulas of Semantic Measure: A Survey

Kalthoum Rezgui
Higher Institute of Management/SOIE Laboratory, Tunis, Tunisia
Email: kalthoum.rezgui@isg.rnu.tn

Hédia Mhiri and Khaled Ghédira
Higher Institute of Management/SOIE Laboratory, Tunis, Tunisia
Email: {hedia.mhiri, khaled.ghedira}@ isg.rnu.tn

***Abstract*—In recent years, several semantic similarity and relatedness measures have been developed and applied in many domains including linguistics, biomedical informatics, GeoInformatics, and Semantic Web. This paper discusses different semantic measures which compute similarity and relatedness scores between concepts based on a knowledge representation model offered by ontologies and semantic networks. The benchmarks and approaches used for the evaluation of semantic similarity methods are also described. The aim of this paper is to give a comprehensive view of these measures which helps researchers to choose the best semantic similarity or relatedness metric for their needs.**

***Index Terms*— semantic similarity, semantic relatedness, ontology, semantic Web, WordNet**

I. Introduction

In the literature, several works on semantic measures have been proposed to compute similarity or relatedness scores between concepts based on semantic networks and ontologies. Some of them explore path lengths among nodes in the hierarchy, others consider in addition the position or depth of nodes in the hierarchy, others rely on statistical analysis of corpora to associate probabilities with concepts in order to compute information content represented by nodes while a last group exploits textual descriptions of concepts in dictionaries. In this paper, we start by presenting a classification of semantic measures. Then, in Section 3 and 4, we present and discuss the different approaches related to the problem of computation of a semantic (similarity/relatedness) score in a knowledge representation model, particularly the case of knowledge modeled in the form of a concept hierarchy. In Section 5, we describe well-known benchmarks and broad evaluation approaches that are mostly used for assessing the quality of semantic measures. Finally, conclusion is presented in Section 6.

II. Concept of Similarity, Distance or Semantic Relatedness

In the literature, three main classes of semantic measures between concepts are commonly quoted:

- **Semantic similarity** when the measure computes whether two concepts are semantically similar, that is, they share common properties and attributes.
- **Semantic relatedness** when the measure computes whether two concepts are semantically related, that is, they are connected in their function. It is considered as a general case of semantic similarity in the works of Resnik [1] and Budanitsky and Hirst [2].
- **Semantic distance** when the measure computes whether two concepts are semantically distant. According to [2], semantic distance is the inverse of semantic relatedness. The idea behind this is that "*the more two terms are semantically related, the more semantically close they are*" [2].

III. Semantic Similarity Measures

In general, the works dealing with semantic similarity measures can be classified into three families of approaches: edge-based approaches, node-based approaches or information-theoretic approaches, and hybrid approaches. Most of these methods exploit particular lexical resources, such as dictionaries, corpus, or well structured taxonomies.

A. Edge-based Approaches

This category of semantic measure approaches is based on the length of paths in a tree to determine the distance between two given concepts. In what follows, we present the similarity measures of Rada et al. [3], Zhong et al. [4], Sussna [5], and Wu and Palmer [6]. The main problem of the proposed approaches is that each similarity measure is tied to a particular application or assumes a particular domain model.

1) The measure of Rada et al.

Rada et al. [3] defined a similarity measure for semantic networks based on taxonomic links "*is-a*". To compute the similarity between two ontology concepts, we calculate the distance between them, denoted as dist *C*1, *C*2, in terms of the minimum number of edges which separate them. The similarity measure is defined by the following formula:

doi:10.4304/jetwi.5.4.333-342

$$Sim_{Rada} = \frac{1}{1+\text{dist}_{\text{RADA}}(C1,C2)} \quad (1)$$

with $dist_{RADA}(C1,C2) = len(C1,C2)$ and $len(C1,C2)$ is the length of the shortest path between $C1$ and $C2$. Despite its simplicity, the distance of Rada does not take into account the positions of edges in the concept hierarchy. However, this information influences on the semantic weight of an edge [7].

2) The measure of Zhong

Zhong et al. [4] defines the similarity between two concepts $C1$ and $C2$ by computing the distance between them. This distance is calculated by the positions of the concepts $C1$ and $C2$ in the hierarchy. The model proposed by [4] implies two assumptions: the semantic differences between upper level concepts are bigger than those between lower level concepts (i.e. two general concepts are less similar than two specialized ones) and that the distance between brothers is greater than the distance between parent and child. The similarity measure of [4] is defined as:

$$Sim_{zhong} = 1 - dist_{zhong}(C1,C2) \quad (2)$$

Zhong defines a score (milestone) for every node in the hierarchy obtained from the following formula:

$$milestone(n) = \frac{1/2}{kl^{(n)}} \quad (3)$$

where k is a predefined parameter that enables to intensify or to decrease the speed of evolution of the score according to the depth (k is set to 2 as used in Corese http://www-sop.inria.fr/edelweiss/software/corese/) and $l(n)$ is the depth of the node n in the hierarchy. The distance between two concepts $C1$ and $C2$ is then defined by the milestones of the latters and their closest common parent $ccp(C1,C2)$ as follows:

$$dist_{zhong}(C1,C2) = dist_{zhong}(C1,ccp) + dist_{zhong}(C2,ccp) \quad (4)$$

with

$$dist_{zhong}(C,ccp) = milestone(ccp) - milestone(C)$$

3) The measure of Sussna

The approach of [5] is based on the following idea: "*Let two pairs of concepts separated by the same number of edges (i.e. same length of the shortest path). Then concepts of the deepest pairs (i.e. the furthest away from the root) are closest semantically*". One thus concludes that even with fix distance in the graph, the semantic distance can change. This assumption is justified by the fact that the deeper a node is, the more it is specialized, thus the more it is representative of a precise notion. The distance formula of Sussna is then based on the depth of nodes in the hierarchy and the distance in terms of nodes number.

Besides, [5] seeks to differentiate the different types of relation. For each relation r, the author attributes a weight or a range $[min_r; max_r]$ of weights according to the type of relations that it represents. For example, relations such as hypernymy, hyponymy, holonymy, and meronymy have weights between $\min_r = 1$ and $\max_r = 2$; for antonymy relation, $\min_r = \max_r = 2.5$; and for synonymy, $\min_r = \max_r = 0$. The weight of each edge of type r from some node $C1$ is reduced by a factor which depends on the number of edges, $edges_r^t$, of the same type leaving $C1$:

$$w(C1 \rightarrow r) = \max_r = \frac{\max_r - \min_r}{edges_r(C1)} \quad (5)$$

where $edges_r(C1)$ is the function that computes the number of edges of type r leaving $C1$. It's important to note that Sussna considers that the relations between concepts are not symmetric. In the majority of cases, two opposite relations do not have the same weight (or the same range of weights) and thus if r' is the opposite of r, $w(C1 \rightarrow_r C2) \neq w(C2 \rightarrow_{r'} C1)$. From weights, Sussna defines the distance between two adjacent concepts $C1$ and $C2$ as:

$$dist_S(C1,C2) = \frac{w(C1 \rightarrow_r C2) + w(C2 \rightarrow_{r'} C1)}{2 \times \min[depth(C1), depth(C2)]} \quad (6)$$

The semantic distance between two arbitrary concepts $C1$ and $C2$ is the sum of the distance between the pairs of adjacent nodes along the shortest path connecting them:

$$dist_{Sussna} = \sum_{(x',y') \in sp(C1,C2)} dist'_{Sussna}(x',y') \quad (7)$$

Although this formula employs in theory different types of relations, it was not validated on this point in practice. Moreover, from the version 1.5 of WordNet [8] this formula is not effective any more. WordNet (Fellbaum, 1998) is a large lexical database of English where nouns, verbs, adjectives and adverbs are grouped into sets of synonyms (or synsets). Each synset represents a distinct concept. The synsets are connected by relations: synonymy (i.e. words that denote the same concept and are interchangeable in many contexts, all the components of a synset are synonyms), hyperonymy (i.e. *is-a* relation), hyponymy (i.e. the reverse relation of hyperonymy or subsumption relation), meronymy (i.e. *part-whole* or *part-of* relation), holonymy (i.e. the reverse relation of meronymy or *has-a* relation), antonymy (i.e. the *complement-of* relation for the opposites). Nevertheless, the measure of [5] remains interesting since it is the first which introduced the idea that depth plays a main role in the distance.

4) The measure of Wu and Palmer

The formula proposed by [6] computes the similarity between two concepts in an ontology restricted to taxonomic links and it is close to the idea of Zhong regarding the use of the closet common parent of both concepts and their depth in the hierarchy. The similarity between $C1$ and $C2$ is defined by the following formula:

$$Sim_{W\&P}(C1,C2) = \frac{2depth(C)}{depth(C1)+depth(C2)} \quad (8)$$

where C is the most specific common subsumer of $C1$ and $C2$, $depth(C)$ is the number of arcs that separates C from the root of the taxonomy, and $depth(Ci)$ is the number of edges that separates the concept Ci from the root via C.

As a conclusion of this section, the semantic similarity measures presented above have the advantages to be easy to implement. However, they do not take into account the information content of concepts. In what follows, we present some information content approaches.

B. Information Theory or Node-based Approaches

In information theory-based approaches, similarity measures employ a corpus with an ontology restricted to hierarchical links and rest on the computation of the information content IC which a concept represents. This weight must be recalculated with each change of the knowledge base. The notion of information content was first introduced by [1] and was measured by the negative log likelihood of the probability of the concept:

$$IC_{\mathrm{Re}snik(C)} = -\log(P(C)) \quad (9)$$

where $P(C)$ denotes the occurrence probability of concept C in a corpus as well as concepts which it subsumes i.e. (its descendants). Concept frequencies used to estimate concept probabilities are obtained by statistically analyzing a corpus. The probability of encountering an instance of concept C is calculated by the following formula:

$$P(C) = \frac{\sum_{w \in words(C)} freq(w)}{N} \quad (10)$$

where $words(C)$ is the set of words (nouns) subsuming the concept C and N is the total number of words present in the corpus. The idea behind the use of the log function is the more probable a concept is, the less information it expresses. In other terms, frequent words are less informative of infrequent ones. The major disadvantage of the measure of information content lies in the obligation to have a corpus to calculate probabilities. Others authors proposed further measures to compute the information content value of a concept, such as [9] based on the depth and the density and [4] based on the depth.

1) The measure of Resnik

Resnik [1] proposed an alternative way to evaluate the semantic similarity in a taxonomy based on the notion of information content which considers the most informative class instead of the path length. In particular, [1] defines the similarity between two concepts by calculating the information that they share in common. The hypothesis of Resnik is that if two concepts are semantically close, then their closet common parent is close to them and thus its information content is a good indicator. The information shared by two concepts is indicated by the information content of their most specific common subsumer (*mscs*). Accordingly, the information content of a concept C is thus the negative log likelihood. As the probability of encountering the concept C increases, its information content value decreases. The similarity measure of Resnik is then defined as:

$$Sim_{\mathrm{Re}snik}(C1,C2) = IC(mscs(C1,C2)) \quad (11)$$

where $IC(mscs(C1,C2))=-log(P(mscs(C1,C2))$ and $P(mscs(C1,C2))$ denotes the probability of the most specific concept subsumer. We can observe that the higher the position of the *mscs* of both concepts in the hierarchy, the lower their similarity is. If the taxonomy has a unique top node, its probability will be 1, so if the *mscs* of two concepts is the top node for example, their similarity is $-log(1) = 0$. The main limit of Resnik's measure is that it does not take into account the information content of concepts $C1$ and $C2$. Besides, it does not consider the length of the path from the root node to this *mscs* and the depth of concepts $C1$ and $C2$ [7].

2) The semantic similarity of Seco et al.

Seco et al. [9] proposed another measure of the Information Content value which completely rests on the taxonomic structure of WordNet [8]. The assumption behind their method is that concepts with many hyponyms are less informative than concepts that are leaf nodes. In this method, the IC value of a concept depends on the number of its hyponyms and a constant.

$$IC_{Seco} = \frac{\log\left(\frac{hypo(C)+1}{\max_{wn}}\right)}{\log\left(\frac{1}{\max_{wn}}\right)} = 1 - \frac{\log(hypo(c)+1)}{\log(\max_{wn})} \quad (12)$$

where $hypo(C)$ is the function which returns the number of hyponyms of a given concept and max_{wn} is a constant that is set to the maximum number of concepts existing in the taxonomy. To evaluate their IC metric, [9] compared the results of the similarity measures of Resnik [1], Lin [10], and Jiang and Conrath, [11] when using the IC value of [1] with those when using their IC value by correlating the similarity scores with those of human judgments provided by Miller and Charles [12]. The evaluation confirmed the authors' initial assumption regarding the usefulness of the hierarchical structure and suggests the use of other taxonomies such as Gene Ontology (http://www.geneontology.org/) to have a generalized metric and thus achieving domain independence.

3) The universal similarity measure of Lin

Lin [10] tried to define a universal similarity measure that would be applicable to different domains (e.g. ordinal values, feature vectors, word similarity, and semantic similarity in a taxonomy) or knowledge representation forms. This measure was derived from a set of assumptions and captures the following three intuitions about similarity:

1. The similarity between two objects *A* and *B* is related to their commonality; the more commonality they share, the more similar they are.
2. The similarity between two objects *A* and *B* is related to the difference between them; the more differences they have; the less similar they are.
3. The maximum similarity between two objects *A* and *B* is reached when *A* and *B* are identical, no matter how much commonality they share.

Lin [10] defined the commonality between *A* and *B* as the information content of the proposition that states the commonalities between them:

$$I(common(A,B)) \tag{13}$$

Besides, Lin defined the difference between *A* and *B* as:

$$I(description(A,B)) - I(common(A,B)) \tag{14}$$

where *description(A,B)* is a proposition describing what *A* and *B* are. Based on these assumptions, [10] proved the following similarity theorem: The similarity between *A* and *B* is measured by the ratio between the amount of information needed to state their commonality and the information needed to fully describe what they are:

$$Sim_{Lin}(A,B) = \frac{\log P(common(A,B))}{\log P(description(A,B))} \tag{15}$$

According to [10], it is only necessary to specify the probability computation according to a domain in order to obtain its own similarity measure. To demonstrate this assumption, [10] provides as examples, a similarity between strings, between two words based on a corpus, between two concepts of a taxonomy, and between ordinal values [7]. The similarity measure that Lin [10] proposed between two concepts $C1$ and $C2$ in a taxonomy is expressed by:

$$Sim_{Lin}(C1,C2) = \frac{2\times \log P(mscs(C1,C2))}{(\log P(C1) + \log P(C2))} \tag{16}$$

where the probabilities $P(C)$ are obtained w.r.t Resnik's $P(c)$ (10). In this measure, [10] defined the shared information content between two concepts $C1$ and $C2$ by 2× the information content of their most specific common subsumer ($mscs(c1, c2)$) and the information content of the description by the sum of the descriptions of the two objects. To evaluate his similarity measure, Lin computed the similarity between 28 pairs of concepts taken from WordNet using his measure and those of Resnik [1] and Wu and Palmer [6] and correlated the obtained scores with scores assigned by human subjects in the experiments of Miller and Charles. The comparison shows that his similarity measure presents a slightly higher correlation with human judgments than the other two measures [10].

C. Hybrid Approaches

Hybrid approaches combine edge-based techniques and information content by considering the shortest path between two concepts and the density of all nodes along this same path in the similarity computation. Information content values are obtained through statistical analysis of corpora and are taken into account as a decision factor.

1) The measure of Jiang and Conrath

The measure of Jiang and Conrath [11] combines the information content of the most specific common subsumer and those of the concerned concepts, and consequently, it can mitigate the limits of Resnik's method. Their semantic similarity relies on the importance degree of a link in the graph and the local density of a node, its depth and type [7]. Recalling the definition of information content, they defined the strength of a link as:

$$dist_{J\&C}(C,parent(C)) = IC(C) - IC(parent(C)) \tag{17}$$

Besides, recalling the idea of the shortest path between two concepts in the taxonomy, the semantic distance of Jiang and Conrath between an arbitrary pair of concepts is given by the sum of distances along the shortest path that connects these concepts:

$$\mathrm{dist}_{J\&C}(C1,C2) = \sum_{c\in sp(C1,C2)} \mathrm{mscs}(C1,C2)\,\mathrm{dist}_{J\&C}(C,\mathrm{parent}(C)) \tag{18}$$

where $sp(C1,C2)$ denotes the set of all nodes in the shortest path from $C1$ to $C2$. The node $mscs(C1,C2)$ is removed from $sp(C1,C2)$ in this formula because it has no parent in the set. Considering (17) and (18), the final Jiang and Conrath's semantic distance formula between two concepts $C1$ and $C2$ is defined as:

$$dist(C1,C2) = (IC(C1)+IC(C2)) - (2?IC(mscs(C1,C2))) \tag{19}$$

This distance contains the same components as the Lin's similarity however their combination is not a ratio but a difference. The similarity measure of [11] is then defined by the reverse of the semantic distance:

$$Sim_{J\&C}(C1,C2) = \frac{1}{dist(C1,C2)} \tag{20}$$

2) The similarity measure of Leacock and Chodorow

The similarity measure of Leacock and Chodorow [13] takes into account the path length between concepts in an ontology restricted to taxonomic links and the depth of the taxonomy:

$$Sim_{L\&C}(C1,C2) = -\log\left(\frac{len(C1,C2)}{2譒AX}\right) \tag{21}$$

where $len(C1,C2)$ is the length of the shortest path between two concepts $C1$ and $C2$ and MAX is the maximum taxonomy depth of the information source. The path length is measured by the number of nodes in the path instead of links. The hypothesis of Leacock and Chodorow is to approximate the probability by taking

into account the path length. The measure of [13] enables to avoid the computation of the information content but it keeps the concept of the information theory. It transforms the Rada distance into a similarity. In the same way, measures which consider only the shortest path length are imprecise because they do not take into account the density or depth of concepts.

IV. Semantic Relatedness Measures

Semantic relatedness measures [7, 14-16] compute the degree to which a pair of concepts are related considering the whole set of semantic links among them. Consequently, semantic relatedness is a generalization of semantic similarity. In other terms, similar concepts are also semantically related but the inverse is not necessarily true, that is, concepts which are related by lexical or functional relationship can be dissimilar. The computation of semantic relatedness has many applications in different areas, such as natural language processing, information extraction and retrieval, lexical selection, automatic correction of word errors in text, and word sense disambiguation. In this section, we present some methods which have been proposed to compute degrees of relatedness among texts, words or concepts. These measures can be classified into lexical resource-based measures, Wikipedia-based measures, and Web-based measures according to the source of knowledge utilized.

A. The Relatedness Measure of Hirst and St-Onge

In [14], Hirst and St-Onge proposed a WordNet-based definition of semantic relatedness which seeks relation between two different words considering their synsets. In particular, they defined three types of relation between two words: extra-strong, strong and medium-strong. A relation between two words is strong if: (a) they have a synset in common (e.g. human and person), (b) they are associated to different synsets interlinked by an horizontal link (e.g. precursor and successor), or (c) there is any type of link between a synset associated with each word and one of the words is a compound word that includes the other (e.g. school and private school). A relation between two words is medium-strong if there is an allowable path connecting synsets of the related words. A path is allowable if it does not contain more than five links between synsets and respects one of the eight allowable patterns. The hypothesis behind this is "*The longer the path and the more changes of direction, the lower the weight*"[14]. The authors have associated a direction among the values *Upward* (i.e. a generalization link), *Downward* (i.e. a specialization link), and *Horizontal* (i.e. antonymy or similarity links) for each relation type in WordNet. The directions are assigned according to the type of links in WordNet. The allowable eight patterns of paths in a medium-strong relation are U, UD, UH, UHD, D, DH, HD, H. In this method, the similarity computation is based on the allowable patterns of path. Once a regular path is found, the weight of the relation type (i.e. the path between two words) is defined by:

$$\mathrm{Rel}(C1,C2)=\begin{cases}3\times \mathrm{C(extra-strong)}; 2\times \mathrm{C(strong)}\\ \mathrm{C-len}(C1,C2)-k\times \mathrm{turns}(C1,C2)(\mathrm{medium-strong})\end{cases} \quad (22)$$

The semantic relatedness of [14] is defined by:

$$Sim_{H\&S} = weight(path(C1,C2)) = C - len(C1,C2) - k\ ?tu\ rns(C1,C2) \quad (23)$$

where C and k are two constants (they are fixed to C = 8 and k = 1 [2]), $len(C1,C2)$ is the length of the shortest path taking into account the directions that are affected according to the type of relation in WordNet, and $turns(C1,C2)$ is the number of direction changes in the path. This measure adapts the Rada's measure to further take account of non-hierarchical relations in an ontology. Thus, it has the same limits of the Rada's measure as it does not consider the density or the depth of concepts and it does not make use of information contents that nodes represent (i.e. it assumes that the information content of all nodes is uniform).

B. The Relatedness Measure of Mazuel and Sabouret

Mazuel and Sabouret [7] focused on the issue of semantic relatedness in a semantic network and proposed a new semantic distance to compute the degree of relatedness between two concepts of a taxonomy augmented with non-hierarchical relations. This measure takes into account different kinds of relations (i.e. subsumption (is-a), meronymy (part-of) or any other domain specific relation) and uses a set of rules to discard unallowable paths generated by the presence of non-hierarchical relations. These rules are inspired from the works of the patterns of semantically correct paths of [14]. In this method, [7] distinguished between single-relation paths and multiple-relation paths and proposed measures for each situation. A single-relation path is a path whose edges are all of the same type: a hierarchical path (i.e. representing the relation *is-a*) or a non-hierarchical path. To compute the weight of a hierarchical single-relation path between two concepts x and y in the ontology, [7] reused the Jiang and Conrath measure which is given by the difference between weights of the two concepts:

$$W(path_{X\in\{is\text{-}a,includes\}}(x,y)) = |IC(x) - IC(y)| \quad (24)$$

To compute the weight of a single-relation path when the relation is not hierarchical, [7] proposed a new formula because the information content of nodes is calculated according to the hierarchical structure of the taxonomy. The authors associated a static weight to each relation type that represents its semantic cost. Besides, they based their formula on the n/n+1 function which simulates the log form. Consequently, the weight of a path between two concepts x and y given its weight and its length is defined by:

$$W(path_X(x,y)) = TC_X \times \left(\frac{|path_X(C1,C2)|}{|path_X(C1,C2)|+1}\right) \quad (25)$$

In the case of a mixed-relation path which contains different kinds of relations, [7] proposed to decompose it in an ordered set of *n* sub-paths based on the transitive nature of the edges of a single-relation path. The weight of a mixed-relation path between two concepts *x* and *y* is defined as the sum of weights of sub-paths composing the minimal decomposition of the path:

$$W(path(x,y)) = \sum_{p \in T_{\min}(path(x,y))} W(p) \quad (26)$$

where $T_{min}(path(x, y))$ is the unique ordered set of sub-paths. Besides, [7] demonstrated that the weight of an hierarchical mixed-relation path containing only two kinds of relations: (a) the relation *is-a* between concept *C1* and the mscs of concepts *C1* and *C2*, and (b) the relation *includes* from the mscs to concept *C2* corresponds to the Jiang and Contrath's distance:

$$\begin{aligned} &W(path(C1,C2)) = W(path_{is\text{-}a}(C1,mscs(C1,C2))) \\ &+W(path_{includes}(mscs(C1,C2),C2))) \\ &=|IC(C1)- IC(mscs(C1,C2))|+|IC(mscs(C1,C2))- IC(C2)| \\ &=IC(C1)+ IC(C2)- 2\ ?IC(mscs\ (C1,C2)) \end{aligned} \quad (27)$$

The final distance measure of [7] considers only the semantically correct paths between two concepts *C*1 and *C*2, and thus it corresponds to the minimal weight among the set of valid paths as defined in the following formula:

$$dist(C1,C2) = \min_{\{p \in \Pi_{(C1,C2)} | HSO(p)=true\}} W(p) \quad (28)$$

where $\Pi(C1,C2)$ is the set of elementary paths (i.e. acyclic) between the concepts *C1* and *C2*. To obtain the set of valid paths, the authors used the function $HSO : \Pi(C1,C2) \rightarrow B$ which determines, according to the path patterns of Hirst and St-Onge [14], if a path is semantically correct (i.e. *HSO*(*p*) = *true*) or not (i.e. *HSO*(*p*) = *false*). This distance can be converted in a semantic relatedness measure by using the classic linear conversion of Resnik:

$$rel(C1,C2) = 2\ ?IC_{\max} - dist(C1,C2) \quad (29)$$

To evaluate their semantic relatedness measure, [7] employed two testing sets from the literature: the test of Miller & Charles [12] and the test of WordSimilarity-353 [17]. Besides, they compared their measure with the similarity measures of Rada, Resnik, Lin and Jiang Contrath and the relatedness measure of Hirst St-Onge. Experimental results show that they obtained best correlation w.r.t human judgments. However, [7] considered only the noun sub-part of WordNet 3.0 and the test focused only on the non-hierarchical transitive relation "*part-of*". In addition, this measure rests on a taxonomic model augmented with one heterogeneous relation and needs to be extended to model complex relations between concepts (i.e. intersections, disjunctions of classes, etc.) such as OWL-Lite .

C. Web-based Semantic Relatedness Measure of Gracia and Mena

In [15], Gracia and Mena proposed the *NormalizedWebDistanceNWD(x,y)* which is a generalization of the Cilibrasi and Vitanyi's Normalized Google Distance NGD(x, y) (30) to compute semantic relatedness between two plain words (or search terms) indexed by different Web search engines.

$$NGD(x,y) = \frac{\max\{\log f(x), \log f(y)\} - \log f(x,y)}{\log N - \min\{\log f(x), \log f(y)\}} \quad (30)$$

where $f(x)$ denotes the number of pages containing *x*, $f(x, y)$ denotes the number of pages containing both words *x* and *y*, and *N* is a normalizing factor. Frequencies are computed using Google page counts. The proposed semantic relatedness measure between two search terms *x* and *y* is defined as:

$$relWeb(x,y) = e^{-2NWD(x,y)} \quad (31)$$

In a later version of their work, Gracia and Mena have taken the word relatedness as a basis to define a new measure that computes how much a pair of ontology terms are semantically related. This measure captures the following desirable features:

- **Domain independent:** it computes relatedness between terms from different ontologies by exploiting some elements of their available semantic descriptions.
- **Universality:** it does not rely on specific lexical resources (e.g. corpus, dictionaries, or WordNet) or knowledge representation languages (e.g. OWL).
- **Maximum coverage:** since it uses the Web as knowledge source, it guarantees a maximum coverage of possible interpretations of the words and thus it extends the scope of applications (e.g. word sense disambiguation, ontology matching, etc.).

To do that, the proposed method computes the degree of semantic relatedness between a pair of senses that two ontological terms represent (i.e. a class, a property or an instance) by considering two levels of semantic description: Level0 which represents the term label and its synonyms and Level1 which represents the ontological context of the term. This latter describes the set of other ontological terms and it corresponds to (a) the set of direct hypernyms if the term is a class, or (b) the set of domain classes if the term is a property, or (c) the class it belongs to if the term is an instance. The relatedness between two ontological terms *a* and *b* at Level0 is computed by (32) whereas the relatedness at Level1 is measured by (33).

$$rel_0(a,b) = \frac{\sum_{i,j} relWeb(syn_{ai}, syn_{bj})}{|Syn(a)|.|Syn(b)|} \quad (32)$$

$$i = 1..|Syn(a)|$$

$$j = 1..|Syn(b)|$$

where *Syn*(*a*) and *Syn*(*b*) denotes the set of synonyms of terms *a* and *b* and *OC*(*a*) and *OC*(*b*) denotes their ontological context.

$$rel_1(a,b) = \frac{\sum_{i,j} rel_0(OC_{ai}, OC_{bj})}{|OC(a)|.|OC(b)|} \quad (33)$$

$$i = 1..|OC(a)|$$
$$j = 1.|OC(b)|$$

The final relatedness degree between two ontological terms is the combination of the semantic relatedness values obtained from (32) and (33) after being weighted as follows:

$$rel(a,b) = w_0.rel_0(a,b) + w_1.rel_1(a,b) \quad (34)$$

where $w_0 \geq 0, w_1 \geq 0$ and $w_0 + w_1 = 1$. Gracia and Mena [15] considered only two levels of semantic description for a term based on an assumption derived from Resnik's idea [1] which assumes that the higher a word is in the hierarchy that characterize the sense of an ontological term, the lesser information content it expresses, and consequently it is less significant to characterize the term. As future works, [15] planned to explore other variations of the method by weighting differently the synonyms of a term in (32) or by considering alternative definitions of the ontological context. Finally, the authors proposed a mixed relatedness measure between ontology terms and plain words in order to cover other usage scenarios. Relatedness at *Levels* 0 and 1 are computed by the following equations:

$$rel_0(t,w) = \frac{\sum_i relWeb(syn_{ti}, w)}{|Syn(t)|} \quad i = 1..|Syn(t)| \quad (35)$$

$$rel_1(t,w) = \frac{\sum_i rel_0(OC_{ti}, w)}{|OC(t)|} \quad i = 1..|OC(t)| \quad (36)$$

The final relatedness between an ontology term *t* and a plain word *w* is a combination of results of (35) and (36):

$$rel(t,w) = w_0.rel_0(t,w) + w_1.rel_1(t,w) \quad (37)$$

where $w_0 \geq 0$, $w_1 \geq 0$, and $w_0 + w_1 = 1$. Besides, the authors carried out two experiments to test the application of their Web-based relatedness measure in disambiguation and ontology matching tasks. Experiments were done using a new test data set consisting of 30 pairs of English nouns that are connected with different types of relations (e.g. similarity, meronymy, frequent association, etc.) and rated for semantic relatedness by a group of 30 university graduated persons on a scale ranging from 0 to 4 (i.e. from no relatedness to identical or strongly related words). They used the Spearman's rank correlation coefficient to determine the correlation between their results and those of humans. The results show that Web-based measures present a better correlation with human judgments than WordNet-based measures.

D. The Extended Gloss Overlap Measure of Banerjee and Pedersen

Banerjee and Pedersen [18] proposed another measure to quantify semantic relatedness between concepts, namely, the extended gloss overlap measure, which is based on the computation of the number of shared words (or overlaps) in the concepts definitions taken from a machine readable dictionary. The basic idea of this approach consists in expanding the glosses of the words being compared by including also glosses of concepts which are recognized to be related to them and their neighbors according to explicit relations provided in the lexical database WordNet. This approach extends the one proposed by Lesk [19] who assumed that related word senses are usually described using the same words and thus he defined a relatedness measure based on gloss overlaps but which considers only overlaps among the glosses of the candidate senses of the target word and those that surround it in the given context. This is a considerable limitation as most dictionary glosses tend to be short and therefore do not provide enough words to find overlaps with. The proposed measure (38) takes as input a pair of synsets and generates a numeric value of semantic relatedness based on the number of overlapping words in their respective glosses as well as in the glosses of synsets they are connected to in a given concept hierarchy. In order to test the proposed relatedness measure, [18] developed an approach to word sense disambiguation (WSD) task which assigns a sense to a target word in a given context that is the most related to the senses of its neighbors using this measure. Evaluation of the measure based on the approach of comparison to human judgments showed a satisfactory correlation coefficient, but word sense disambiguation experiments showed that considering extended gloss overlaps improves the disambiguation results and yields much better than the original Lesk Algorithm [19]. The authors plan to augment the scores of overlaps with global statistics about the word occurrences and to evaluate the measure on different NLP tasks. The relatedness score between the inputs synsets *A* and *B* is measured as the sum of scores of phrasal gloss overlaps between them:

$$relatedness(A,B) = \sum_{\forall (R1,R2) \in RELPAIRS} score(R1(A), R2(B)) \quad (38)$$

where *RELPAIRS* denotes the set of all possible relation pairs formed from the set of relations defined in WordNet (e.g. hypernyms, hyponyms, meronyms, holonym, also-see relation, attribute, pertainym), and *score*() is the function which detects and scores the phrasal gloss overlaps between the inputs. The scoring mechanism consists in assigning a phrasal *n* word overlap the score of n^2.

V. Semantic Measures Evaluation

From the literature, a way to evaluate the results of semantic similarity measures is to find a good correlation between the computed similarity scores and the average similarity ratings provided by human evaluators in benchmarks, such as, Miller and Charles [12] and WordSimilarity-353 [17]. The higher the correlation of a method, the better the method is, i.e. the more it approaches the results of human judgments. A correlation is a number between -1 and +1 which measures the degree of relationship between two variables. A positive value for the correlation implies a positive association whereas a negative value implies a negative or inverse association. The two most commonly used measures of correlation are the Pearson's correlation coefficient and the Spearman's rank correlation coefficient. The Pearson's correlation coefficient enables to analyze linear relations between two variables based on their actual values. The correlation coefficient ranges between -1 and +1 and it is interpreted as follows:

- Near to -1: the two vectors are opposite or negative agreement/disagreement.
- Around 0: the two vectors are independent or no agreement.
- Near to 1: the two vectors are dependent positive agreement.

Let two vectors of length $x_1, x_2, \ldots, x_n$ and $y_1, y_2, \ldots, y_n$, the Pearson's correlation coefficient is defined as follows:

$$p = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2 \sum_i (y_i - \bar{y})^2}} \quad (39)$$

The Spearman rank correlation coefficient (40) is a non-parametric measure of correlation which uses ranks to calculate the correlation rather than absolute values. The correlation coefficient is a number ranging also between -1 (total disagreement) and +1 (total agreement). A positive correlation is one in which the ranks of both variables increase together. A negative correlation is one in which the ranks of one variable increase as the ranks of the other variable decrease. A correlation of +1 or -1 will arise if the relationship between the two variables is exactly linear. A correlation close to zero means that there is no linear relationship between the ranks.

$$p = 1 - \frac{6\sum_{i=1}^{n} d_i^2}{n^3 - n} \quad (40)$$

Some software packages were already proposed that enable to compute similarity, such as the WordNet::Similarity package (http://talisker.d.umn.edu/cgi-bin/similarity/similarity.cgi) and the Nuno Seco package [9]. The WordNet::Similarity package consists of Perl modules that implement the following WordNet-based measures: Resnik [1], Lin [10], Jiang and Conrath [11], Leacock and Chodorow [13], Hirst and St-Onge [14], Wu and Palmer [6], the extended gloss overlaps measure of Banerjee et al. [18], and two measures based on context vectors by Patwardhan and Pedersen [20]. The Nuno Seco package is implemented in java and can be downloaded from the rubric "extension" in WordNet site. In what follows, we present the most commonly used similarity benchmarks, namely, Miller and Charles [12] and WordSimilarity-353 [17].

A. Benchmarks for Semantic Measures Evaluation

1) Miller and Charles test

The test of Charles and Miller [12] is a set of 30 pairs of nouns with their similarity ratings determined by human judgments. 38 undergraduate students have participated to the test and were asked to rate the similarity of each pair on a scale from 0 (not similar) to 4 (perfect synonymy). The average rating of each pair represents a good estimate of how similar the two words are. Most of works presented in the previous section end with an evaluation w.r.t 30 pairs selected by Miller and Charles. This protocol enabled to fix a work base for the research community on semantic distances. In fact, if a distance measure reached the coefficient of 0.91, then it will be representative of the real distance of a human judgment. However, the test of Miller and Charles is based on synonymy judgment, thus, it is mainly oriented to evaluate similarity measures not relatedness measures. Most of selected word couples do not have functional relations between them since it was explicitly requested to human subjects to judge similarity between concepts. Hence, the dataset of Miller and Charles is not adapted to test functional relations between two concepts and accordingly to evaluate a semantic relatedness measure.

2) The WordSimilarity-353 test collection

The WordSimilarity-353 benchmark [17] is a set of 353 English word pairs for which subjects were asked to estimate the similarity or relatedness of words on a scale from 0 (totally unrelated words) to 10 (strong related or identical words). The WordSimilarity-353 test set can be used to train and to test algorithms implementing semantic measures. It was proposed in order to mitigate the problems of Miller and Charles test. It includes all the 30 noun pairs of Miller and Charles. Agirre et al. [21] proposed to split the WordSimilarity-353 dataset into two subsets (http://alfonseca.org/eng/research/wordsim353.html), the first subset contains the union of similar and unrelated pairs and focuses on computing similarity whereas the second subset contains the union of related and unrelated pairs and focuses on computing relatedness.

As a conclusion, human judgments of similarity and relatedness provided in benchmarks presented above are supposed to be correct by definition and give clear evaluation of the performance of a measure. However, the main drawback of this approach lies in the difficulty of obtaining a large set of reliable and subject-independent judgments for comparison.

B. Approaches for Semantic Measures Evaluation

In [2], Budanitsky and Hirst focused on comparing the performance of five WordNet-based measures (Hirst and

St-Onge, Jiang and Conrath, Leacock and Chodorow, Lin, and Resnik) based on two evaluation approaches: comparison of computed semantic relatedness or similarity scores with human judgments and comparison of the performance of these measures in a particular application. To compute the frequency of concepts needed in the information-theoretic approaches, [2] used the Brown Corpus of American English [22]. For the first evaluation method, [2] computed the correlation coefficients between human and computer ratings of the word pairs of Rubenstein-Goodenough and Miller-Charles in order to determine the strength of the linear association between them. The comparison has shown that the difference between the values of the highest and lowest correlation coefficients for the test of Miller and Charles and the test of Rubenstein-Goodenough are in the order of 0.1 and 0.06 respectively. Besides, [2] employed the upper bound for the Miller and Charles word pairs to compare the performance of the selected measures on it and they found that the correlation coefficients compare quite favorably with this upper bound. Moreover, [2] concluded that the measures do not react in the same way toward the increase of the size of the dataset. In fact, while the correlation coefficients with human judgments of *relH&S*, *simL&C* and *simR* improve, those of *distJ&C* and *simLin* deteriorate.

As [2] point out, though human evaluation approach is considered as the best method to evaluate a similarity or a relatedness measure, its main drawback stands for the difficulty of acquiring considerable amounts of test sets of word pairs with human-assigned scores. Besides, [2] continue to add that they need in NLP tasks human judgments of the relatedness of word-senses not just words. This need can be satisfied by exploring contexts. In order to overcome the problems posed by this approach, [2] used an application-based evaluation approach which compares relatedness measures based on their ability to detect and correct semantic anomalies such as malapropisms. To test the measures, they created a corpus of malapropisms. Then, they tried to detect and correct them by an algorithm that uses the five measures of semantic relatedness with different searching scopes. They considered it as a retrieval task and evaluated it in terms of Precision, Recall, and F-measure. The analysis of differences between measures' results for the malapropism suspicion phase shows that the Jiang and Conrath's measure outperforms the others in all scopes. The results for malapropism detection phase shows also that the measure of Jiang and Conrath does better than the other measures. Besides, evaluation shows that though the measure of Hirst and St-Onge is the only one among the others that focuses on computing semantic relatedness, it presents poor performance in both stages. To support the evaluation results of [2] regarding the performance of the Jiang and Conrath's measure, we considered other approaches in the NLP domain that also apply WordNet-based measures and we perceived that these results are consistent with the experiments' results of approaches of Stevenson and Greenwood [23], Kohomban and Lee [24], and Patwardhan et al.[25]. In fact, [23] proposed a semantic similarity approach to information extraction pattern acquisition which relies on comparing patterns similarity using their own measure. This measure takes into account pattern vectors and their transposes and a similarity matrix which contains information about semantic similarity between pairs of lexical items. They experimented several measures in order to populate the semantic similarity matrix and found that the measure defined by Jiang and Conrath is the most effective one. Similarly, [24] described a method to learn generic semantic classes of a given word instance in order to mitigate the lack of training data problem in word sense disambiguation. In this method, [24] computed the relatedness between the sense of the test word and the most frequent sense of it within the candidate class using different similarity measures. The experiments showed that the measure of Jiang and Conrath gives best results for this task. In the same way, [25] carried out word sense disambiguation experiments to evaluate the same five measures of semantic relatedness that have been also compared by [2] in addition to the extended gloss overlap measure. Experiments were performed using noun data gathered from the English Lexical sample task of SENSEVAL-2 (http://www.senseval.org/). Similarly, the authors found that the extended gloss overlap measure of Banerjee and Pedersen [18] and the semantic distance measure of Jiang and Conrath [11] result in the highest accuracy.

VI. Conclusion

In this paper, we presented a classification of semantic measures and discussed the basics of the various approaches proposed for each class. The benchmarks and evaluation approaches that are commonly used by researchers to assess the quality of their semantic measure proposals are also stated. This survey could help researchers to choose the most appropriate similarity or relatedness measure for their needs.

References

[1] P. Resnik, "Using information content to evaluate semantic similarity in a taxonomy," in *Proc. 14th International Joint Conf. on Artificial Intelligence* - Volume 1, 1995, pp. 448–453.

[2] A. Budanitsky and G. Hirst, "Evaluating WordNet-based Measures of Lexical Semantic Relatednes," *Computational Linguistics*, vol. 32, no. 1, 2006, pp. 13–47.

[3] R. Rada, H. Mili, E. Bicknell, and M. Blettner, "Development and application of a metric on semantic nets," *IEEE transactions on systems, man, and cybernetics*, vol. 19, no. 1, 1989, pp. 17–30.

[4] J. Zhong, H. Zhu, J. Li, and Y. Yu, "Conceptual graph matching for search," in *Proc. 10th International Conf. on Conceptual Structures: Integration and Interfaces*, London, UK: Springer-Verlag, 2002, pp. 92–106.

[5] M. Sussna, "Word sense disambiguation for free-text indexing using a massive semantic network," in *Proc. Second International Conf. on Information and Knowledge Management*, New York, NY, USA: ACM, 1993, pp. 67–74.

[6] Z. Wu and M. Palmer, "Verbs semantics and lexical selection," in *Proc. 32nd Annu. Meeting on Association for*

Computational Linguistics, Las Cruces, New Mexico, 1994, pp. 133–138.

[7] L. Mazuel and N. Sabouret, "Semantic relatedness in semantic networks," in *Proc. 18th European Conf. on Artificial Intelligence*, Amsterdam, The Netherlands: IOS Press, 2008, pp. 727–728.

[8] C. Fellbaum, WordNet: An Electronic Lexical Database. MIT Press, 1998.

[9] N. Seco, T. Veale, and J. Hayes, "An intrinsic information content metric for semantic similarity in wordnet," in *Proc. 16th European Conf. on Artificial Intelligence*, 2004, pp. 1089–1090.

[10] D. Lin. "An Information-Theoretic Definition of Similarity", in *Proc. of 15th International Conf. on Machine Learning*, SanFrancisco, CA, USA, 199, pp. 296–3048.

[11] J. J. Jiang and D. W. Conrath, "Semantic similarity based on corpus statistics and lexical taxonomy," in *Proc. International Conf. Research on Computational Linguistics*, Taiwan, 1997, pp. 19–33.

[12] G. Miller and W. Charles, "Contextual correlates of semantic similarity," *Language and cognitive processes*, vol. 6, no. 1, 1991, pp. 1–28.

[13] C. Leacock and M. Chodorow, "Combining local context and WordNet similarity for word sense identification," in *WordNet: A Lexical Reference System and its Application*, C. Fellbaum, Ed. Cambridge, Massachusetts: MIT Press, 1998, pp. 265–283.

[14] G. Hirst and D. St-Onge, "Lexical chains as representations of context for the detection and correction of malapropisms," in *WordNet: An Electronic Lexical Database (Language, Speech, and Communication)*, MIT Press, 1998, pp. 305–332.

[15] J. Gracia and E. Mena, "Web-based measure of semantic relatedness," in *Proc. 9th international conf. on Web Information Systems Engineering*, Berlin, Heidelberg: Springer-Verlag, 2008, pp. 136–150.

[16] E. Blanchard, M. Harzallah, H. Briand, and S. Kuntz, "A typology of ontology-based semantic measures," *in Proc. Open Interop Workshop on Enterprise Modelling and Ontologies for Interoperability*, 2005.

[17] L. Finkelstein, E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman, and E. Ruppin, "Placing search in context: the concept revisited," *ACM Transactions On Information Systems*, vol. 20, no. 1, January 2002, pp.116–131.

[18] S. Banerjee and T. Pedersen, "Extended gloss overlaps as a measure of semantic relatedness," in *Proc. 18th International Joint Conf. on Artificial intelligence*, 2003, pp. 805–810.

[19] M. Lesk, "Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone," in *Proc. 5th Annu. International Conf. on Systems Documentation*, pp. 24–26, 1986.

[20] S. Patwardhan and T. Pedersen, "Using WordNet-based context vectors to estimate the semantic relatedness of concepts," in *Proc. EACL 2006 workshop making sense of sense - Bringing computational linguistics and psycholinguistics together*, 2006, pp. 1–8.

[21] E. Agirre, E. Alfonseca, K. Hall, J. Kravalova, M. Pasca, and A. Soroa, "A study on similarity and relatedness using distributional and WordNet-based approaches," in *Proc. Human Language Technologies: The 2009 Annu. Conf. of the North American Chapter of the Association for Computational Linguistics*, 2009, pp. 19–27.

[22] N. W. Francis and H. Kucera, "Frequency Analysis of English Usage: Lexicon and Grammar," *J. of English Linguistics*, vol. 18, no. 1, 1982, pp. 64–70.

[23] M. Stevenson and M. A. Greenwood, "A Semantic Approach to IE Pattern Induction," in *Proc. 43rd Annu. Meeting on Association for Computational Linguistics*, 2005, pp. 379–386.

[24] U. S. Kohomban and W. S. Lee, "Learning semantic classes for word sense disambiguation," in *Proc. 43rd Annu. Meeting on Association for Computational Linguistics*, 2005, pp. 34–41.

[25] S. Patwardhan, S. Banerjee, and T. Pedersen, "Using measures of semantic relatedness for word sense disambiguation," in *Proc. 4th International Conf. on Computational Linguistics and Intelligent Text Processing*, 2003, pp. 241–257.

Mining Opinion Targets from Text Documents: A Review

Khairullah Khan [1,2]
[1] Computer and Information Sciences Department, Universiti Teknologi PETRONAS, Malaysia
[2.] Institute of Engineering and Computing Sciences, University of Science & Technology Bannu Pakistan
khairullah_k@yahoo.com

Baharum B. Baharudin
Computer and Information Sciences Department, Universiti Teknologi PETRONAS, Malaysia
baharbh@petronas.com.my

Aurangzeb Khan
Aurangzebb_khan@yahoo.com
[2.] Institute of Engineering and Computing Sciences, University of Science & Technology Bannu Pakistan

***Abstract*—Opinion targets identification is an important task of the opinion mining problem. Several approaches have been employed for this task, which can be broadly divided into two major categories: supervised and unsupervised. The supervised approaches require training data, which need manual work and are mostly domain dependent. The unsupervised technique is most popularly used due to its two main advantages: domain independent and no need for training data. This paper presents a review of the state of the art unsupervised approaches for opinion target identification due to its potential applications in opinion mining from user discourse. This study compares the existing approaches that might be helpful in the future research work of opinion mining and features extraction.**

***Index Terms*—Opinion Mining; Sentiment Analysis; Opinion Targets; Machine Learning**

I. INTRODUCTION

What other people think is naturally important for human guidance. Through opinions, humans can flux together diverse approaches, experiences, wisdom and knowledge of people for decision making. Humans like to take part in discussions and present their points of view. People often ask their friends, family members, and field experts for information during the decision making process. They use opinions to express their points of view based on experience, observation, concept, beliefs, and perceptions. The point of view about something can either be positive (shows goodness) or negative (shows badness), which is called the polarity of the opinion.

Opinions can be expressed in different ways. The following example sentences show different ways of opinion representations.

Shahid is a good Cricket player.
The meal was quite good.
The hotel was expensive.
Terrorists deserve no mercy!
Hotel A is more expensive than B.
Coffee is expensive but tea is cheap.
This player is not worth any price and I recommend that you don't purchase it.

An opinion has three main components i.e. the opinion holder or source of opinion, the object about which the opinion is expressed and the evaluation, view or appraisal which is called the opinion. For opinion identification, all these components are important.

Opinion can be collected from different sources e.g. individual interaction, newspapers, television, internet etc.; however, the internet is the richest source of opinion collection. Before the World Wide Web (WWW), people collected opinions manually. If an individual was to make a decision, he/she typically asked for opinions from friends and family members. Organizations conducted surveys through focused groups for collecting public opinion. This type of survey was expensive and laborious. Now, the internet provides this information with a single click and a very little cost.

With the advent of web 2.0, the internet allows web users to generate web content online and post their information independently. Due to this facility of the internet, web users can participate in a collaborative environment around the globe. Hence, the internet has become a rich source for social networks, customer feedback, online shopping etc. According to a survey, more than 45,000 new blogs are created daily along with 1.2 million new posts each day [1]. The information collected through these services is used for various types of decision making e.g. social network for: political, religious, security, and policy making; customer feedback for: products sales, purchases, and manufacturing. The

doi:10.4304/jetwi.5.4.343-353

trend of online shopping portals is increasing day by day. The vendors collect customer feedback for future trend prediction and product improvement through these portals. Opinion is the key element which has provided the inspiration for this work.

Although the internet is a rich source of opinions, having millions of blogs, forums and social websites with a large volume of updated information, unfortunately the web data is typically unstructured text which cannot be directly used for knowledge representation. Moreover, such a huge volume of data cannot be processed manually. Hence, efficient tools and potential techniques are needed to extract and summarize opinions. Research communities are trying for efficient utilization of the web information for knowledge requisition; this is in order to present it to the user in a well understandable and summarized manner. With the emergence of web 2.0, the task of posting and collecting opinions through the Web has become easy; however, the quality control, processing, compilation, and summarization have become potential research problems.

With the growing need of opinion analysis a new area called Opinion Mining is gradually emerged in the field of Natural Language Processing (NLP) and Text Mining. OM is a procedure used to extract opinion from a text. "OM is a recent discipline at the crossroads of information retrieval, text mining and computational linguistics which tries to detect the opinions expressed in natural language texts" [1]. OM is a field of knowledge discovery and data mining (KDD) which uses NLP and statistical machine learning techniques to differentiate opinionated text from factual text. OM tasks involve opinion identification, opinion classification (positive, negative, and neutral), target identification, source identification and opinion summarization. Hence, OM tasks require techniques from the field of NLP, Information Retrieval (IR); and Text Mining. The main issue is how to automatically identify opinion components from unstructured text and summarize the opinion about an entity from a huge volume of unstructured text. An overview of the OM concept is shown in the Figure 1.

Figure 1. Overview of opinion mining process

The focus of this study is opinion target identification for the opinion mining process. The problem of opinion target identification is related to the question: "opinion about what?'. Opinion target identification is essential for opinion mining. For example, the in-depth analysis of every aspect of a product based on consumer opinion is equally important for consumers, merchants and manufacturers. In order to compare the reviews, it is required to automatically identify and extract those features which are discussed in the reviews. Furthermore, analysis of a product at feature level is more important e.g. which features of the product are liked and which are disliked by consumers [2] . Hence, feature mining of products is important for opinion mining and summarization. The task of feature mining provides a base for opinion summarization[3]. There are various problems related to opinion target extraction. Generally speaking, if a system is capable of identifying a target feature in a sentence or document, then it must be able to identify opinionated terms or evaluative expressions in that sentence or document. Thus in order to identify opinion targets at sentence or document level, the system should be able to identify evaluative expressions. Also, some features are not explicitly presented and are predicted from term semantics called implicit features. The focus of this paper is on explicit feature.

Opinion target identification is basically a classification problem which is defined as: to classify noun phrase or term as opinion target or not [4]. There are two widely used classification methods i.e. supervised and unsupervised. The supervised method needs prior knowledge annotated through manual process. Unsupervised classification depends on heuristics procedures and rules which do not need previous knowledge. Hence there are two main advantages of unsupervised method over supervised: Supervised technique need training data which manually labeled while unsupervised do not need hand-crafted training datasets, moreover supervised techniques are generally domain dependent as training data are manually labeled for specific domain [5, 6]. This paper provides a review of existing unsupervised approaches which has been popularly employed for opinion targets extraction within the past few years. The main goal of this work is to identify potential techniques for opinion targets extraction that might be helpful in the future research work in opinion mining. Hence the main contribution of this paper is the analysis of the factors that affect the existing unsupervised learning techniques of the opinion target extraction.

The entire paper is organized as follows: Section II explains related work and existing unsupervised approaches for opinion target extraction from unstructured reviews. Section III provides comparative analysis of the existing approaches and Section IV Concludes the paper.

II. Unsupervised Approaches for Opinion Targets Identification

The unsupervised techniques has been popularly used for opinion target identification [6-17].

Popescu & Etzioni [9] used an unsupervised technique to extract product features and opinions from unstructured reviews. This paper introduces the OPINE system based on the unsupervised information extraction approach to mine product features from reviews. OPINE uses syntactic patterns for semantic orientation of words for identification of opinion phrases and their polarity.

Carenini, Ng et al. [15] developed a model based on user defined knowledge to create a taxonomy of product features. This paper introduces an improved unsupervised method for feature extraction that uses the taxonomy of the product features. The results of the combined approach are higher than the existing unsupervised technique; however, the pre-knowledge base mechanism makes the approach domain dependent.

Holzinger, Krüpl, & Herzog [10] use domain ontologies based on tabular data from web content to bootstrap a knowledge acquisition process for extraction of product features. This method creates a wrapper for data extraction from Web tables and ontology building. The model uses logical rules and data integration to reason about product specific properties and the higher-order knowledge of product features.

Bloom, Garg, & Argamon [14] describe an unsupervised technique for features and appraisal extraction. The authors believe that appraisal expression is a fundamental task in sentiment analysis. The appraisal expression is a textual unit expressing an evaluative attitude towards some target. Their paper proposed evaluative expressions to extract opinion targets. The system effectively exploited the adjectival appraisal expressions for target identification.

Ben-David, Blitzer et al. [16] proposed a structural correspondence learning (SCL) algorithm for domain classification. The idea depends on perception to get a prediction of new domain features based on training domain features; in other words, the author describes under what conditions a classifier trained on the source domain can be adapted for use in the target domain? This model is inspired by feature based domain classification. Blitzer, Dredze et al. [17] extended the structural SCL algorithm for opinion target identification.

Lu and Zhai [18] proposed automatic integration of opinions expressed in a well-written expert review with opinions scattered in various sources such as blogs and forums. The paper proposes a semi-supervised topic model to solve the problem in a principled way. The author performed experiments on integrating opinions about two quite different topics, i.e. a product and political reviews. The focus of this paper is to develop a generalized model that should be effective on multiple domains for extraction of opinion targets.

Ferreira, Jakob et al. [11] describe an extended pattern based feature extraction using a modified Log Likelihood Ratio Test (LRT), which was initially employed by [7] for target identification. This paper also presented an extended annotated scheme for product features, which was initially presented by [8] and a comparative analysis between feature extraction through Association Mining and LRT techniques.

The association rule mining for target extraction is initially implemented by [8] for target extraction, and extended by Chen et al. [12] using semantic based patterns for frequent feature refinement and identification of infrequent features.

One of the latest work on feature level analysis of opinion is reported by [6]. This paper describes a semi-supervised technique for feature grouping. Feature grouping is an important task for summarization of opinion. Same features can be expressed by different synonyms, words or phrases. To produce a useful summary, these words and phrases are grouped. For feature grouping the process generate an initial list to bootstrap the process using lexical characteristics of terms. This method empirically showed good results.

Goujon [4] presents a text mining approach based on linguistic knowledge to automatically detect opinion targets in relation to topic elements. This paper focuses on identification of opinion targets related to the specific topic. This approach exploits linguistic patterns for target identification.

The two most frequently reported unsupervised approaches for target and opinion identification are Association Mining (AM) [19] and Likelihood Ratio Test (LRT) approach [20]. The following sub sections provide a detail overview these two approaches.

A. Association Mining Approach

The Association Mining approach for product features extraction (AME) was employed by [8] for the first time. In this work, they extract frequent features through association rule mining technique [19]. This algorithm was originally used for market basket analysis which predicts dependency of an item sale on another item. Based on the analogy of the market basket analysis the authors in [8] assume that the words in a sentence can be considered as bought items. Hence the association between terms can predict features and opinion words association. The implementation of this technique was very successful in features extraction. Later on this approach is extended by [12] for the same task with semantic based pruning for frequent features refinement and identification of infrequent features. The subsequent approach improved the results of opinion target identification through association rule mining algorithm.

The AME approach formulates the process of opinion target identification into two steps. In the first step, it extracts frequent features through the Apriori algorithm and in the second step it employs a pruning algorithm to refine the candidate features from irrelevant features. The overall process is shown in a block diagram Figure 2.

The Apriori algorithm is called the king of data mining techniques as it was introduced in the early stages of the data mining field and has been potentially exploited for data mining and knowledge discovery. This algorithm has two steps: in step 1, it generates frequent item sets from a set of transactions that satisfies a user's specified minimum support, and in the second step, it

discovers association rules from the frequent item sets discovered in step 1.

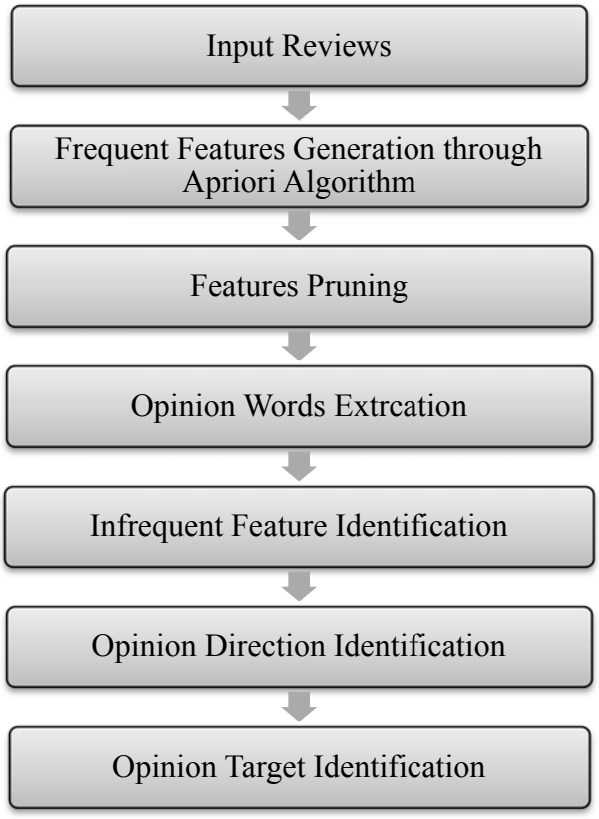

Figure 2: Association Mining approach for opinion target extraction[8]

The association mining approaches uses the first step of the Apriori algorithm for extraction of product features that are frequently discussed in the review documents. The Apriori algorithm generates frequent feature sets from nouns in the reviews. This approach formulates the process of frequent feature identification as presented below.

Frequent Features Identification

The algorithm searches for frequently occurring product features in the input documents using the following steps.

- Each sentence is considered as a transaction.
- Each noun phrase in the sentence is considered as an item. Feature sets are created from the items.
- The algorithm then iterates through all the feature sets and counts the frequencies of each individual feature.

Based on the total number of candidate features a threshold value is calculated which is called the minimum support. Any feature having a frequency less than the minimum support threshold are discarded from the features' list. The authors in this work consider a feature set as frequent if it appears in more than 1% (minimum support) of the review sentences.

Features Pruning

The second step of this approach is pruning, which is used to refine the features obtained in step 1. The following two pruning steps are described.

- *Compactness Pruning*

Compactness is used to check features that contain two or three words and remove those features which are not co-occurring more than at least two times. For example, having the phrase "battery life" if it appears in two or more sentences at a distance of at most three words in between them then it is a compact feature. However, if it does not co-occur at least two times then it is removed from the feature list.

- *Redundancy Pruning*

Redundancy pruning is used to remove redundant features that contain single words. A feature is considered as redundant if it occurs in a compact feature and has a lower frequency then the p-support. The p-support is different from the general support count in association mining. For example, "life" occurs 6 times and "battery life' occurs 5 times then in the candidate features, the feature "life" alone is considered as a redundant feature. This work only considers nouns for the features and this rule does not consider any other lexical categories at all.

B. Association Mining by Wei et al. (2010)

This approach uses a semantic-based refinement of the frequent features obtained through the association mining approach. This work describes a model based on a list of positive and negative subjective adjectives defined in the General Inquirer (GI). The aim of semantic-based refinement is to overcome the following two limitations of the [8] approach:

- Frequent but non Product Features,
- Infrequent but Product Features.

This approach describes the following three semantic-based pruning rules to handle these limitations.

Co-occurrence-based Pruning

The previously described association mining approach is based on the frequency of noun phrases to discover frequent features. However, some of the noun phrases in a document may have a high frequency but not be an opinion target. This rule is designed to address this limitation. This rule is defined as:

- For each frequent feature a count is carried out for the number of review sentences in which the feature co-occurred with subjective adjectives.
- If the count obtained in the previous step is less than a prescribed co-occurrence threshold value (this study considers it as 1) then it is removed from the frequent feature list.

The formal representation of this model is given as below.

$$\text{IF} \quad \sum_{i=1}^{|S|} co - occur(f, ow, s_i) < \propto \text{ Then } F = F - \{f\} \quad (1)$$

Where

$$co - occur(fqf, ow, s_i) = \begin{cases} 1 \text{ if } \exists \text{ op} \in \text{ow such that f} \in s_i \text{ and op} \in s_i \\ 0 \quad \text{otherwise} \end{cases} \quad (2)$$

Here |S| represents the number of sentences, f is a frequent feature, s_i a sentence, ow an opinion word, and F frequent feature sets. In this step, the frequent features are considered as product features.

Opinion-based Infrequent Feature Identification

The earlier approach employs the nearest adjective as opinion words to identify infrequent features in the review sentences that do not contain frequent features. This approach may not be effective for all adjectives e.g. "such/JJ thing/NN", "whole/JJ lot/NN", "simple/JJ point/NN" etc. Similarly in a sentence "The/DT picture/NN is/VBZ not/RB rich/JJ in/IN color/NN", the noun closest to the adjective "rich" is "color" but picture is not the target feature, rather the word color is target. To address this limitation, the author describes the following rule:

If a review sentence contains a subjective adjective, then this rule first examines the word or group of words immediately after the subjective adjective in the sentence. If the word after the adjective is a noun or noun phrase, then it is considered as an infrequent feature and is added to the list of frequent features. If the word after the adjective is not a noun phrase, then the heuristic searches for a noun phrase before the adjective in the sentence. For example, with the sentence "this/WDT camera/NN has/VBZ excellent/JJ picture/NN quality/NN", according to this rule, "picture quality" is the actual feature. Hence, this rule satisfies both conditions of the nearest adjective and is similar to the previous approach; moreover, the situation as described in the previous sentence where the feature is picture and as the word "in/IN" is not a noun after the subjective adjective thus it searches for the nearest noun before the subjective adjective.

Conjunction-based Infrequent Feature Identification

Some of the features rarely occur and thus the frequency based approach fails to identify them. However, based on the conjoined relation with other features they can be easily identified. This rule is described as follows:

For every conjunction of nouns and noun phrases in each review sentence, if one has been identified as a target feature, then this rule includes the remaining nouns and noun phrases in the conjunction as a product feature. The mathematical model of this rule is defined as:

$$\text{If } \exists\ np_i \in CN \text{ such that } np_i \in PF, \quad \text{Then}$$

$$\forall\ np_j \in CN \text{ and } np_j \neq np_i\ PF \cup \{np_j\} \quad (3)$$

Where np_i and np_j represents a noun or noun phrase in conjunction (CN) with the identified features, and PF represents product features already identified in the previous step.

Based on the above three rules, this approach improved both precision and recall of the association mining approach for opinion target identification. This approach reported an average improvement of about 10.7% in recall and 2.5% in precision.

C. Likelihood Ratio Test Approach

The other potentially employed unsupervised classification technique is the Likelihood Ratio Test (LRT). The LRT was introduced by [20] and has been reported in different NLP tasks. The LRT was employed by [7] for product feature extraction and sentiment analysis. One of the latest approaches for product feature identification using the LRT technique is described by [11]. The LRT technique assumes that a feature related to the topic is explicitly presented by a noun phrase in the document using syntactic patterns associated with subjective adjectives. The overall process is explained in the Figure 3.

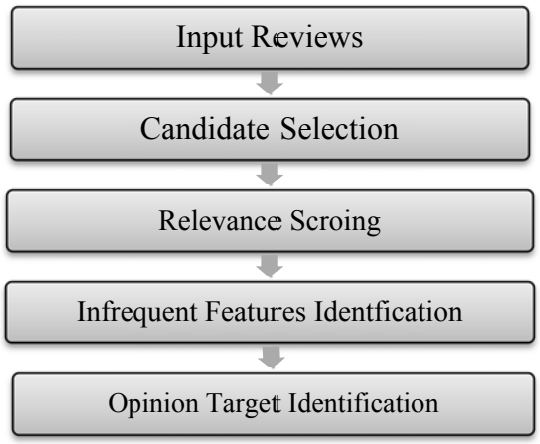

Figure 3: Opinion targets extraction [7, 11]

Yi, Nasukawa et al. [7] described different linguistic patterns termed as base noun phrases for candidate selection and then employs relevance scoring to refine the candidate features. The overall process of the likelihood ratio test based target extraction is defined as below.

Selection of Candidate Feature using Linguistic Patterns

In this approach the selection process of candidate features is based on noun phrase patterns. The following patterns are employed in this work.

- *Base Noun Phrases (BNP)*

These patterns are used to extract candidate features using the following combination of noun (NN) and adjective (JJ).

NN, NN NN, JJ NN, NN NN NN, JJ NN NN, JJ JJ NN.

- *Definite Base Noun Phrase (dBNP)*

These patterns present noun phrases (BNP) with the definite article "the" before the BNP. The idea behind these patterns is that some proper nouns start with the article "the" therefore these patterns are useful for named entity extraction

- *Beginning Definite Base Noun Phrases (bBNP)*

This pattern presents a sequence of definite noun phrases followed by verbs. This pattern describes that the noun phrase in between the article "the" and a verb are mostly observed as features.

Relevance Scoring

Yi, Nasukawa et al. [7] presented unsupervised technique for relevance scoring of candidate features. This paper employed two unsupervised techniques, i.e. The Mixture Model, and LRT. However, the results show that the LRT performed relatively good. The likelihood ratio test is formulated as:

Let D_c denoted topic relevant collection of documents and D_n represents collection of documents not relevant to the topic. Then a base noun phrases occurring in the D_c are candidate feature to be classified as topic relevant or topic irrelevant using the likelihood ratio test as: if the likelihood score of BNP satisfies the predefined threshold value then BNP is considered as target feature. The LRT value for any BNP x is calculated as:

Let n_1 denotes the frequency of a BNP in a Dc, n_2 represents sum of frequencies of all BNPs in D_c except x, n_3 denoted frequency of x in D_n, and n_4 represents the sum of frequencies of all BNPs in D_n except the frequency of x.

Then the ratios of relevancy of the BNP x to topic and non-topic, which are presented by r_1 and r_2 respectively, can be calculated as below.

$$r_1 = \frac{n_1}{n_1+n_2} \quad (4)$$

$$r_2 = \frac{n_3}{n_3+n_4} \quad (5)$$

Thus the combined ratio is calculated as:

$$r = \frac{n1+n_3}{n_1+n_2+n_3+n_4} \quad (6)$$

Hence to normalize the ratios with log:

$$lr = (n_1+n_2)\log(r) + (n_3+n_4)\log(1-r) - n_1\log(r_1) - n_3\log(1-r_1) - n_2\log(r_2) - n_4\log(1-r_2) \quad (7)$$

Hence the likelihood ratio is calculated as below.

$$-2\log\Lambda = \begin{cases} -2*lr & if\ r_2 < r_1 \\ 0, & if\ r_2 \geq r_1 \end{cases} \quad (8)$$

The likelihood is directly proportional to the value of $-2\log\Lambda$.

D. *Likelihood Approach by Ferreira et al. (2008)*

A more extensive study of the LRT approach for opinion target identification is presented by this paper. As mentioned in the previous sub section, the LRT was employed by [7]; however, due to non-availability of proper data sets for evaluation measures the author only calculated precision.

Ferreira et al. (2008) performed an evaluation on the state-of the art datasets, which are manually, annotated corpuses created by [8]. Furthermore, they have modified the algorithm using subsequent similarity measures based on the following two rules.

Identification of Feature Boundaries for Patterns

The earlier work [7] used BNPs, dBNPs and bBNPs for candidate feature identification. Noun phrases in these patterns are considered as candidate features. However, there is no rule mentioned for multiple matches. For example, in the pattern "battery life", three features can be reflected: "battery life", "battery", and "life". The recent work [11] extended the earlier algorithm, which only selects the longest BNP patterns. For example, in the above expression this rule considers only "battery life" as a feature.

Classification of Patterns with an Adjective Noun (JJNN)

Most of the candidate BNPs is combinations of JJNN patterns. The adjective sometimes represents features e.g. "digital images" and sometimes it represents an opinion e.g. beautiful image; hence, it is required to classify the subsequent adjectives in the candidate patterns. Subsequent similarity rule is employed by [11], which have improved the results. Another main contribution of this paper is the new annotation scheme of the features in the existing dataset that were originally employed by [8]. According to the revised annotation scheme, the number of features was increased as their focus was on all features.

III. COMPARATIVE ANALYSIS

This section describes the analysis of the unsupervised approaches that has been potentially employed for opinion targets extraction. As explained in section II there are most popular used techniques that have been employed for opinion targets extraction.

A. *Analysis of Factors Affecting the Existing Approaches*

This section explains the analysis of the factors affecting the existing unsupervised techniques of opinion targets extraction. We have performed analysis on the bench mark dataset that have been employed by the existing approaches. The experimental setup is divided into two broad categories. The first category is related to candidate selection based on linguistic patterns while the

second one is focusing on features selection based relevance scoring.

B. *Datasets*

This section describes the datasets that have been used for the analysis and evaluation in this work. In this work, benchmark datasets of the customer reviews about five different products are employed. These datasets have been reported in numerous works for opinion mining and target identification. These datasets are crawled from amazon review sites and are manually annotated by [8]. The datasets are freely available from the authors' website[1]. In these datasets, each product feature with opinion scoring is properly tagged in each sentence through a manual process according to a prescribed annotation scheme as shown below.

- A sentence is considered as opinionated if it contains positive or negative comments about features of the product.
- Positive and negative comments are opinion statements containing adjectives that either have a positive or negative orientation.
- A product feature is the characteristic of the product about which opinions are expressed by the customers.

The datasets contain customer reviews about four different electronic products, i.e. Camera (Canon G3 and Nikon Coolpix 4300), DVD player (Apex AD2600 Progressive-scan), mp3 player (Creative Labs Nomad Jukebox Zen Xtra 40GB) and cell phone (Nokia 6610). The summary of each dataset is given in Tables 1: including the total number of reviews (number of documents), total number of sentences, number of sentences with opinions and targets with percentage, total distinct base noun phrases which count each distinct BNP as 1; the total target features shows the count of all target features in each dataset, the average target features shows target features out of the total distinct BNPs, the target types show the number of distinct target features in each dataset and the ratio of target features to the total target occurrence.

C. *Experimental Setup*

Although the results are of the aforementioned techniques have been already given in the respective papers and there is no need to reproduce it. However in order to empirically prove the factors affecting the existing approach we have performed analysis on the factors that affect the performance of the existing approaches.

As mentioned in the existing approaches there are two phases of the target extraction techniques. The first phase is related to candidate selection while the second phase is related to relevance scoring. In the candidate selection process patterns of language elements with grammatical relations are employed to identify candidate features. In relevance scoring phase the candidate features are refined using unsupervised machine learning techniques. Hence our experimental setup is divided into the following two phases to identify strength and limitations of the existing approaches in each phase.

D. *Analysis of Patterns for Candidate Selection*

This section provides a comparative analysis of the linguist patterns that have been employed for candidate selection. As mentioned earlier both AME and LRT approaches are using noun phrase for candidate selection. However there is a difference between the selections. AME uses association between the noun phrases and top features with highest frequency is selected that qualify the minimum support as target features. While The LRT select the noun phrases based on grammatical sequence of terms. In order to investigate best patterns for candidate selection the following patterns are examined: Base noun phrase (BNP), Definite based noun phrases (dBNP), Beginning definite base noun phrases (bBNP), and Combined base noun phrase pattern (cBNP). The first four patterns have already been discussed. While the cBNP pattern is employed by [23] which is set of patterns defined as below.

- Noun Phrase-Verb Phrase-Adjective (NP VB JJ)
- Noun Phrase-Verb Phase-Adverb Adjective (NP VB RB JJ)
- Noun Phrase-Verb Phase-Adverb Adjective NN (NP VB RB JJ NN)
- Definite Base Noun Phrase (dBNP)
- Preposition Based Noun Phrase (iBNP)
- Subjective Base Noun Phrase (sBNP)

In order to extract these patterns from the datasets the Stanford part of speech tagger and textSTAT software has been used, The Stanford part of speech tagger is employed for part of speech tagging [21], while TextStat 3.0 is employed for pattern extraction and test analysis [2]. This software is simple and has been used by a number of works for searching terms and strings in English texts [22].

The comparative results are shown in figures 4, 5 and 6. The precision of bBNP is higher than the other patterns as it extracts fare number of features. While the recall of BNP pattern is higher as it extracts all BNPs, however, its recall is very low due to its false negative features. The F-score of our proposed cBNP is significantly higher than the other patterns. Thus the overall performance of cBNP is good.

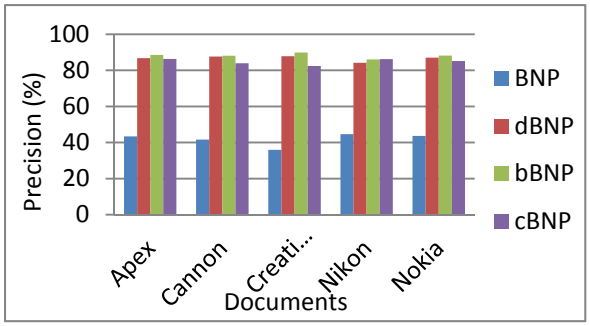

Figure 4: Precision of candidate selection based on dependency patterns

[1] http://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html

[2] http://neon.niederlandistik.fu-berlin.de/en/textstat/

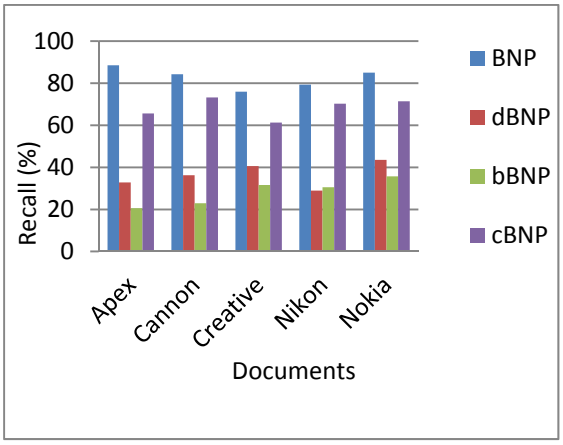

Figure 5: Recall of candidate selection based on dependency patterns

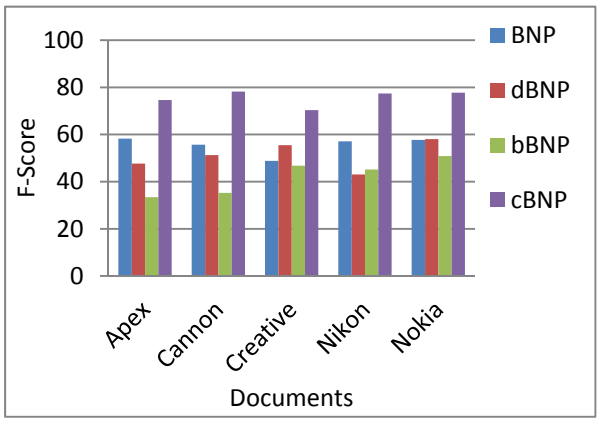

Figure 6: F-Score of candidate selection based on dependency patterns

E. Analysis of Frequency Based Relevance Scoring

This section demonstrates how the target extraction techniques are affected by the threshold values. In order to analyze this problem, we conducted experiment for finding infrequent features on each data set. Table 2 shows the sample of target features which have zero LRT values due to their rear occurrence in the review dataset. Hence based on the low frequency distribution a number of target features cannot be predicted by the unsupervised learning. Table 3 shows the ratio of infrequent features classified by LRT technique.

Refer to the results in Sections B and C there are two main issues related to the extraction of opinion targets. The first issue is related to linguistic patterns that have been employed for candidate selection. The results can be greatly improved with the use of proper patterns. As shown in the graphs in figures 4, 5 and 6, the F-score is significantly improved with the use of combination of patterns.

The other main issue is related to frequency based relevance scoring for features selection. It has been observed that even within large documents there exist a lot of features which have very low frequency hence cannot be detected even by adjusting a small value of threshold. Hence the recall of the unsupervised techniques is greatly affected due high ratio of false negative value.

As discussed in the previous sections the existing unsupervised approaches exploit linguistic patterns and frequency based relevance scoring techniques to identify opinion targets. However there are certain issues related to both patterns selection and relevance scoring that might affect the performance of the techniques. Since most of the work consider base noun phrases as opinion targets. However all base noun phrase in text cannot be opinion targets. Hence the existing research work has been primarily focused on the problem of selecting dependency patterns for targets identification. For example some sentences in a review document may not have opinion while other sentences may have more than one base noun phrases with few opinion targets. For example the sentences “The/DT camera/NN comes/VBZ with/IN a/DT second/JJ battery/NN. I/PRP purchased/VBD it/PRP in/IN a/DT departmental/NN store/NN.” do not have any opinion targets although it have base noun phrase. While in the sentence “The battery/NNP is/VBZ very/RB good/JJ even/RB when/WRB using/VBG flash/NN and/CC lcd/NN” there is only one opinion target “battery” although it has three different BNPs. Hence simply selecting BNPs provides a large false positive ratio. To overcome this issue the existing worked has proposed various solutions. For example the association mining approach assumes that opinion targets are frequently discussed in reviews. However this approach suffers from two major issues i.e. frequent but not opinion target and infrequent but opinion target. As mentioned earlier, to overcome these problems the existing works have proposed pruning. Although the performance have been improved with pruning rules. However, the results show that there is still gap for further improvement.

The Likelihood Ratio Test approach assumes that the Base Noun Phrases with dependency patterns containing subjective adjective are best candidate for opinion targets instead of simply selecting base noun phrases. Hence this technique depends on opinionated expression. However, the question about how to identify opinionated expressions! is itself a challenging problem. There can be more than one noun phrases with adjective in sentences. For example the sentence “The/DT picture/NN quality/NN is/VBZ not/RB rich/JJ in/IN color/NN” have two candidate base noun phrases “Picture quality” and “Color”, and one adjective “rich”. Although the “color” is a feature that can be occurred many items in different sentences; however, in this case the “picture quality” is basically opinion target. According dBNP pattern mentioned earlier, “The picture quality” can to be correctly selected as opinion target from the above sentence. However these patterns are not effective in many cases. For example if we look into the review sentence “this dvd play is basically junk”; it has opinion targets “player” but do not satisfy the dBNP pattern rules. Since LRT based approach also depends on frequency distribution therefore it also suffer from the same two main issues i.e. frequent but none opinion targets and rarely occurred but opinion targets.

TABLE 1:
SUMMARY OF THE FIVE PRODUCT DATASETS WITH MANUALLY TAGGED OPINION TARGETS BY [8]

Description	Dataset				
	Apex	Cannon	Creative	Nikon	Nokia
Reviews	99	45	95	34	41
Total sentences	739	597	1716	346	546
Target types	110	100	180	74	109

TABLE 2:
SAMPLE SET OF INFREQUENT FEATURES

Dataset	**Infrequent Features**
Apex	read,look,sound,price,door,size,design,quality,support,weight,case, forward,output,product,run,unit,video,work,code,direction,disk,display,finish,machine,motor,noise, panel, recognize, service, speed, use, apex
Cannon	body,control,depth,design,display,feel,finish,focus,function,image,learning,look,made,noise,option,print,quality,remote,service,shape,shot,speed,use,weight,zoom
Creative	alarm,appearance,balance,break,build,capacity,case,change,clock,control,cover,creative,deal,design,display,equipment,feature,feel,finding,game,look,looking,manage,memory,music,name,option,panel,pause,play,product,program,quality,recognition,recording,remote,remove,style,support,switch,thing,top,unit,use,value,volume,weight,wheel,work,sorting,navigation
Nikon	construction,control,delay,design,function,image,learn,menu,price,quality,size,software,transfer,use,weight
Nokia	application,background,call,command,construction,design,game,keys,look,memory,message,network,picture,plan,quality,resolution,ring,service,software, sound, speaker, tone, use, voice, work

TABLE 3
DISTRIBUTION OF INFREQUENT FEATURES IN EACH DATASET

Dataset	Total	Frequent	Infrequent	%Infrequent
Apex	110	78	32	29.09090909
Cannon	98	73	25	25.51020408
Creative	179	129	50	27.93296089
Nikon	73	58	15	20.54794521
Nokia	110	84	26	23.63636364

IV. Conclusion

This paper presents a systematic review of unsupervised approaches of opinion target identification from unstructured reviews. This study shows that there two main issued in unsupervised learning of opinion targets from unstruted reviews i.e. Frequent base noun pharse but not target features and Infrequent but target features. Besides a significant improvements in the opinion target identification techniques these two prolmes are still challenging. Our analysis shows that results can be greatly improved with the imrpovement in candidate selection and relevance scroing. We have proposed hybrid patterns based candidate selection that have shown considerable improvement in the true positive. We have also the affect of threshold value on relevance scroing using Likelihood ratio test. It was found that 20 to 30 % infrequent features cannot be detected by the LRT technique due low frequency of the target feature. Hence the recall of the this method low due to high number of flase negative features. This shows that recall can be improved with the selection of infrequent features. Hence future should focus on dependecy patterns and infrequent features for the better improvement in the results.

References

[1] Pang, B. and L. Lee, Opinion Mining and Sentiment Analysis. Foundations and Trends in Information Retrieval, 2008. 2(1-2): p. 135.

[2] Zhang, L. and B. Liu, Identifying noun product features that imply opinions, in Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers - Volume 2. 2011, Association for Computational Linguistics: Portland, Oregon. p. 575-580.

[3] Somprasertsri, G., Mining Feature-Opinion in Online Customer Reviews for Opinion Summarization. . Journal of Universal Computer Science, 2010. vol. 16(6): p. 938-955.

[4] Goujon, B. Text Mining for Opinion Target Detection. in European Intelligence and Security Informatics Conference (EISIC). 2011.

[5] Qiu, G., et al., Domain Specific Opinion Retrieval Information Retrieval, in Fifth Asia Information Retrieval Symposium. 2009, Springer Berlin / Heidelberg: Japan. p. 318-329.

[6] Zhai, Z., et al., Clustering product features for opinion mining, in The fourth ACM international conference on Web search and data mining. 2011, ACM: Hong Kong, China. p. 347-354.

[7] Yi, J., et al. Sentiment analyzer: extracting sentiments about a given topic using natural language processing techniques. in Third IEEE International Conference on Data Mining (ICDM) 2003.

[8] Hu, M. and B. Liu, Mining and summarizing customer reviews, in 10th ACM SIGKDD international conference on Knowledge discovery and data mining. 2004, ACM: Seattle, WA, USA. p. 168-177.

[9] Popescu, A.-M. and O. Etzioni, Extracting product features and opinions from reviews, in Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing. 2005, Association for Computational Linguistics: Vancouver, British Columbia, Canada. p. 339-346.

[10] Holzinger, W., B. Krüpl, and M. Herzog. Using ontologies for extracting product features from web pages. in 5th International Semantic Web Conference, ISWC 2006. 2006. Athens, Georgia, USA.

[11] Ferreira, L., N. Jakob, and I. Gurevych. A Comparative Study of Feature Extraction Algorithms in Customer Reviews. in Semantic Computing, 2008 IEEE International Conference on. 2008.

[12] Wei, C.-P., et al., Understanding what concerns consumers: a semantic approach to product features extraction from consumer reviews. Info Syst E-Bus Management, 2010(8): p. 149-167

[13] Wong, T.-L. and W. Lam, An unsupervised method for joint information extraction and feature mining across different Web sites. Data & Knowledge Engineering, 2009. 68(1): p. 107-125.

[14] Bloom, K., N. Garg, and S. Argamon. Extracting appraisal expressions. in In Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics. 2007. Rochester, New York, USA.

[15] Carenini, G., R.T. Ng, and E. Zwart, Extracting knowledge from evaluative text, in Proceedings of the 3rd international conference on Knowledge capture. 2005, ACM: Banff, Alberta, Canada. p. 11-18.

[16] Ben-David, S., et al., Analysis of representations for domain adaptation. In Advances in Neural Information Processing Systems 2007. vol. 19.

[17] Blitzer, J., M. Dredze, and F. Pereira. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. in 45th Annual Meeting of the Association of Computational Linguistics. 2007. Prague, Czech Republic.

[18] Lu, Y. and C. Zhai. Opinion integration through semi-supervised topic modeling. in 17th International World Wide Web Conference (WWW '08). 2008. Beijing, China.

[19] Agrawal, R. and R. Srikant, Fast Algorithms for Mining Association Rules in Large Databases, in 20th International Conference on Very Large Data Bases. 1994, Morgan Kaufmann Publishers Inc. p. 487-499.

[20] Dunning, T., Accurate methods for the statistics of surprise and coincidence. Comput. Linguist., 1993. 19(1): p. 61-74.

[21] Toutanova, K., et al., Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network, in North American Association for Computational Linguistics (NAACL). 2003. p. 173-180.

[22] Diniz, L., Comparative Review: TextStat 2.5, ANTCONC 3.0, and Compleat Lexical Tutor 4.0. Language Learning & Technology, 2005(Vol 9 Issue 3): p. 22-27.

Khairullah khan received MSc Computer Science Degree from University of Peshawar Pakistan and has PhD degree from Department of Computer and Information Sciences, Universiti Teknologi PETRONAS, Malaysia. He is working as Assistant Professor at University of Science and Technology Bannu Pakistan. His current research interests include NLP, Data Mining, Opinion Mining and Information Retrieval.

Baharum Bin. Baharudin received his Master Degree from Central Michigan University, USA and his PhD degree from University of Bradford, UK. He is currently Associate Professor at the Department of Computer and Information Sciences, Universiti Teknologi PETRONAS Malaysia. His research interests lies in Image Processing, Data Mining and Knowledge Management.

Aurangzeb khan received BS-Degree in Computer Science from Gomal University D.I.Khan, Pakistan, and Master Degree in Information Technology From University of Peshawar, Pakistan, and PhD degree from Department of Computer and Information Sciences, Universiti Teknologi PETRONAS, Malaysia. He is an Assistant Professor at University of Science & Technology Bannu Pakistan. His current research interests include Data Mining, Opinion Mining and Information Retrieval.

A Survey of Word-sense Disambiguation Effective Techniques and Methods for Indian Languages

Shallu
University Institute of Engineering & Technology, Panjab University, Chandigarh, India
Email: shallu.2.chd@gmail.com

Vishal Gupta
University Institute of Engineering & Technology, Panjab University, Chandigarh, India
Email: vishal@pu.ac.in

***Abstract*—Word Sense Disambiguation is a challenging technique in Natural Language Processing. There are some words in the natural languages which can cause ambiguity about the sense of the word.WSD identifies the correct sense of the word in a sentence or a document. The paper summarizes about the history of WSD. We have discussed about the knowledge - based and machine learning – based approaches for WSD. Various supervised learning and unsupervised learning techniques have been discussed. WSD is mainly used in Information Retrieval (IR), Information Extraction (IE), Machine Translation (MT), Content Analysis, Word Processing, Lexicography and Semantic Web. Finally, we have discussed about WSD for Indian languages (Hindi, Malayalam, and Kannada) and other languages (Chinese, Mongolian, Polish, Turkish, English, Myanmar, Arabic, Nepali, Persian, Dutch, and Italian).**

***Index Terms*— Word Sense Disambiguation (WSD), Natural Language Processing (NLP), supervised, unsupervised, knowledge, information retrieval, information extraction, machine translation, context, ambiguity, polysemous words.**

I. INTRODUCTION

There are words in Natural languages which have different meaning for different context but they are spelled same. Those words are called polysemous words. Word sense disambiguation (WSD) is the solution to the problem. Word Sense Disambiguation [1] is a task of automatically assigning a correct sense to the words which are polysemous in a particular context.

Many Natural languages like English, Hindi, Punjabi, French, Chinese, etc. are the languages which have some words whose meaning are different for same spelling in the different context. In English, Words likes Bark, Lie, book, etc. can be considered example of polysemous words. Human beings are blessed with the learning power. They can easily find out what is the correct meaning of a word in a context. But for computer it is a difficult task. So, we need to develop an automatic system which can perform like humans do i.e. the system which can find out the correct meaning of the word in particular context. Context is the text or words which are surrounding to the ambiguous word. Using the context, human can easily sense the correct meaning of the word in that context. So we also need the computer to follow some rules using which the system can evaluate the absolute meaning out of multiple meanings of the word.

If we consider a text T a sequence of words i.e. w_1, w_2, w_3.....w_n. Then, WSD is a task to assign the correct sense for all or some words in the text T.

Two main approaches which are used to WSD are Deep approaches and shallow approaches. Deep approaches uses some kind of knowledge related to the word and shallow approaches see the context in which the word has been used [2]. The other approaches to Word sense Disambiguation are knowledge-based approach, machine learning approach.

The conceptual Model [15] for Word Sense Disambiguation is given below:

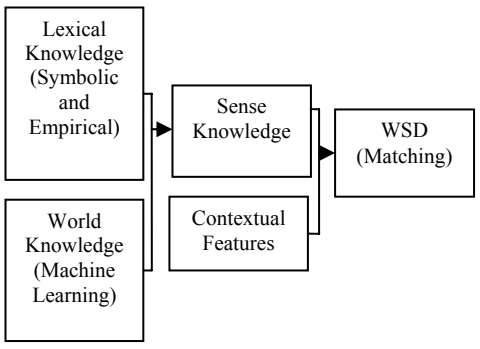

Figure 1 Conceptual Model of WSD [15]

Now, there are so many methods to assign senses, but how to measure which method provide good performance. So, the performance of the WSD can be measured by Precision and recall. Precision is defined as the proportion of correctly identifying senses of those identified, while recall is the proportion of correctly identified senses of total senses.

doi:10.4304/jetwi.5.4.354-360

It is an important and challenging technique for natural language processing (NLP). Many real world applications like machine translation (MT), semantic annotation (SA), semantic mapping (SM), and ontology learning (OL) uses WSD. Information retrieval (IR), information extraction (IE), and speech recognition (SR) are some of the applications in which WSD is used to improve the performance.

The remainder of the paper is organized as follows: in section 2, we mention various approaches for WSD, while in section 3 we present the WSD algorithms for making a word sense disambiguation system. Section 4 covers the applications where WSD is used and in Section 5, we will discuss about WSD for various Indian languages.

II. WSD Approaches

There are two approaches that are followed for Word Sense Disambiguation (WSD): Knowledge Based approach and Machine-Learning Based approach. In Knowledge based approach, it requires external lexical resources like Word Net, dictionary, thesaurus etc. In Machine learning-based approach, systems are trained to perform the task of word sense disambiguation. These two approaches are briefly discussed below

A. Machine Learning Based Approach

It adapts to new circumstances, detects and extrapolates patterns. In machine learning approach, the systems are trained to perform the task of WSD. A classifier is used to learn features and assigns senses to unseen examples. In these approaches, the initial input is the word to be disambiguated called target word, and the text in which it is embedded, called as context. Part-of-Speech tagging is used for processing, in which fixed set of features are extracted which are relevant to the task of learning called linguistic features. These linguistic features can be classified in two classes: collocation features and co-occurrence features. Collocation conceals the information about words that are located to left or right of target word at specific positions. Co-occurrence features contain the data or information about neighboring words. In this approach features are themselves served by the words. The value of feature is the number of times the word occurs in the region surrounding the target word. The region is often a fixed window with target word as center. Three types of techniques of machine learning based approaches are: supervised techniques, unsupervised techniques, and semi-supervised techniques.

Supervised Techniques: The learning here perform in supervision. Let us take the example of the learning process of a small child. The child doesn't know how to read/write. He/she is being taught by the parents at home and then by their teachers in school. The children are trained and modules to recognize the alphabets, numerals, etc. Their each and every action is supervised by the teacher. Actually, a child works on the basis of the output that he/she has to produce. Similarly, a word sense disambiguation system is learned from a representative set of labeled instances drawn from same distribution as test set to be used. Input instances to these approaches are feature encoded along with their appropriate labels. The output of the system is a classifier system capable of assigning labels to new feature encoded inputs. System is informed precisely about what should be emitted as output. In supervised learning, it is assumed that the correct (target) output values are known for each Input. So, actual output is compared with the target output, if there is a difference, an error signal should be generated by the system. This error signal helps the system to learn and reach to the desired or target output.

Unsupervised Technique: In unsupervised learning technique, no supervision is provided. Let us consider an example of a tadpole. Learning is done by itself i.e. child fish learn to swim without any supervision. It is not taught by anyone. Thus its leaning process is independent and not supervised by a teacher. Unsupervised approaches to word sense disambiguation eschew the use of sense tagged data of any kind during the training. In this technique, feature vector representations of unlabeled instances are taken as input and are then grouped into clusters according to a similarity metric. These clusters are then labeled by hand with known word senses. Main disadvantage is that senses are not well defined.

Semi-Supervised Techniques: In semi-supervised learning techniques, the information is present like in supervised but might be less information is given. Here only critic information is available, not the exact information. For example, the system may tell that only particular about of target output is correct and so. The semi-supervised or minimally supervised methods are gaining popularity because of their ability to get by with only a small amount of annotated reference data while often outperforming totally unsupervised methods on large data sets. There are a host of diverse methods and approaches, which learn important characteristics from auxiliary data and cluster or annotate data using the acquired information.

B. Dictionary Based Approach

In this style of approach the dictionary provides both the means of constructing a sense tagger and target senses to be used. An attempt to perform large scale disambiguation has lead to the use of Machine Readable Dictionaries (MRD). In this approach, all the senses of a word that need to be disambiguated are retrieved from the dictionary. These senses are then compared to the dictionary definitions of all the remaining words in context. The sense with highest overlap with these context words is chosen as the correct sense.

For example: consider the phrase 'pine cone' for selecting the correct sense of word cone, following are the definitions for pine and cone:

Pine: kinds of evergreen tree with needle-shaped leaves or waste away through sorrow or illness

Cone: solid body which narrows to a point or something of this shape whether solid or hollow or fruit of certain evergreen trees

In this example, Lesk's [11] method would select cone as the correct sense since two of the words in its entry,

evergreen and tree, overlap with words in the entry for pine.

A major drawback of Dictionary based approaches is the problem of scaling.

III. WSD ALGORITHMS

A. HyperLex

The HyperLex algorithm presented in [14] is entirely corpus-based. Author has used the co-occurrence graphs. All-pair of words in the context are built in the form of co-occurrence graphs. It is a dictionary free method. The nodes in the graph are the words that co-occur with the target word. An edge is used to connect two nodes which concurrent to each other. It uses the properties of small world graph, and has the highly connected components (called hubs) in the graph. These hubs represent the senses produced by the system. These hubs identify the main word used i.e. it identifies the senses of the target word, and is used to perform word sense disambiguation.

In this, first author build the co-occurrence graph using the senses of the target word. The author considers only noun and adjectives. Verbs were also considered by the author but ended up because it was causing a notable degradation in performance. Paragraph is filtered and only nouns and adjectives are considered. All the verbs, prepositions, determiners and stop words are removed from the paragraph. Then a co-occurrence matrix from this filtered set of contexts was generated. Two words appearing in the same paragraph are called co-occur words. The HyperLex for the example used by author [14] is shown below in Fig 2:

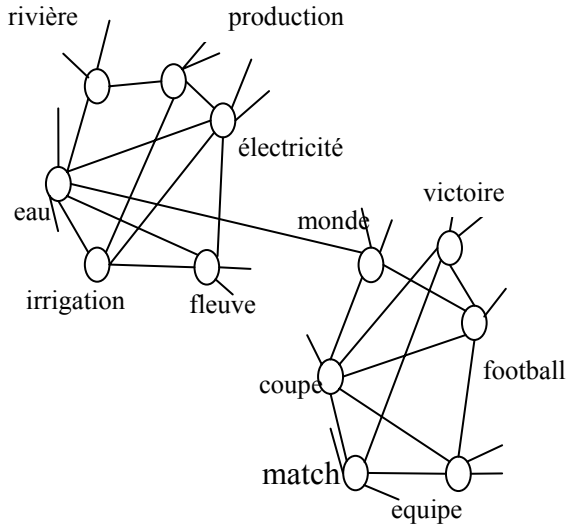

Figure 2 Graph of the co-occurrence of the French word 'barrage' [14]

After this, weights are given to the edges connecting two nodes. The co-occurrence networks are scale-free, so they contain a small number of highly connected hubs and a large number of weakly connected nodes [14]. Co-occurrence graph detects the different uses of a word and thus it amounts to isolate the high-density components. Every high-density component, one of the nodes has a higher degree than the others which is called the root hub of the component. All the root nodes are identified iteratively. For the root node, the node has to have (1) at least 6 specific neighbors (this threshold was determined experimentally), and (2) a weighted clustering coefficient large enough for it to actually be a root hub of a bundle [14]. Then a minimum spanning tree (or MST) is computed over the graph by taking the target word as the root and making its first level having the previously identified root hubs. The complexity of the graph mentioned by author [14] is O (E log (E)), where E is the number of edges in the graph. This MST is then used to construct a disambiguation system, which will tag the target word occurrences in the corpus. Each node v in the tree is assigned a score vector s with dimensions as there exist for the components. HyperLex given by the author provides a tool for domain and lexicon navigation. The results of the HyperLex algorithm were evaluated on the Web page corpus [14]. The best 25 contexts were checked for each of the 50 uses which include 1245 contexts in all. The overall precision obtained was 95.5%.

B. Extended Word Net

In the Lesk algorithm [5], word to disambiguate is given, the dictionary definition or gloss of each of its senses is compared to the glosses of every other word in the phrase. A word is assigned that sense whose gloss shares the largest number of words in common with the glosses of the other words. The algorithm begins a new for each word and does not utilize the senses it previously assigned.

A version of Lesk algorithm in combination with WordNet has been reported for achieving good word sense disambiguation results [13]. In their work, different types of relationships in WordNet have been experimented with. It showed that the best results are obtained when concatenating the descriptions of word senses with the glosses of its first and second-levels hypernyms.

This algorithm is used by Naskar and Bandyopadhyay [7], in which they have used the Word Net lexical database, because it contains different types of relationships between words. They proposed a global approach instead of local approach where all the words in the context window are simultaneously disambiguated in a bid to get the best combination of senses for all the words in the window instead of only the target word. The Lesk algorithm only work for short phrases. But the algorithm proposed by [7] takes the entire sentence under consideration.

The gloss bag is constructed for every sense of every word in the sentence. The gloss-bag is constructed from the POS and sense tagged glosses of synsets, obtained from the Extended Word Net. Once, the gloss-bag creation process is over, the comparison process starts. Each word (say W_i) in the context is compared with each word in the gloss-bag for every sense (say S_k) of every other word (say W_j) in the context. If a match is found, they are checked further for part-of-speech match. If the

words match in part-of speech as well, a score is assigned to both the words: the word being matched (W_i) and the word whose gloss-bag contains the match (W_j). This matching event indicates mutual confidence towards each other, so both words are rewarded for this event. Two two-dimensional vectors are maintained: sense_vote for the word in context, and sense_score for the word in gloss-bag. Once all the comparisons have been made, sense_vote value is added with the sense_score linearly value for each sense of every word to arrive at the combination score for this word-sense pair.

Finally, for any word in the context, the value of sense index that maximizes this sum is declared the assigned sense for this particular word.

Knowledge base used by Naskar and Bandyopadhyay [7] was first 10 Semcor2.0 files. Another approach of knowledge based Disambiguation is using Word Net domains which is used for disambiguate nouns. It follows the unsupervised approach to word sense disambiguation [8] [17]. Domain is defined as set of words which contain the words with the semantic relation. This algorithm use 3 bags for solving ambiguity. Bag 3 contains the target word which we need to disambiguate and bag 3 is compared with bag 1 and bag 2. First, Domain of the word is interpreted and then the sense in that domain is the sense of the target word in bag 3. Precision of the algorithm was 85.9%, and the recall was calculated 62.1%.

C. Improved Unsupervised Learning Probabilistic Model

The purposed algorithm by the authors is a probabilistic model. The probabilistic models have parametric form and parameter estimation [15]**.** It shows an effect of one contextual feature over other contextual features and also, the effect of one contextual feature over the sense of an ambiguous word. The authors have considered the Naïve Bayes form for this purposed model. The posterior probability function, $p(S|F_1, F_2, ..., F_n)$, defined by Bayes Rule given by [15] is:

$p(S|F_1, F_2, ..., F_n)$ = p(F1,F2,…,Fn,S)/p(F1,F2,…,Fn) = p(S)×∏i=1 to n p(Fi ⎪ S))/(∑s p(F1,F2,…,Fn,S)

In this algorithm, first the word net is used. The word net will first annotate the senses of words that have single semantic item. Second step of this algorithm focuses on the part-of-speech ambiguity in which it will remove the ambiguity prior to sense disambiguation. After that, it will check for the words which are ambiguous and those words which are required for disambiguation. In this word net will define all the senses related to ambiguous words. Feature selection is the next important step in which features are selected using the Z-test. It will remove the noise in the disambiguation. Feature selection is used to increase the accuracy of WSD. In the result of this, the efficiency of WSD will also be improved.

In the proposed model [15], if *w* represents an ambiguous word and *wj* represents a contextual word, then their mutual information, *I* (*w, wj)*, is defined as:

I (w, wj) = log2 p(w,wj)/(p(w)p(wj)

After this, authors have estimated the initial parameters values. This algorithm proposes a statistical learning algorithm which will estimate initial parameter values of the model from raw untagged text because it is an unsupervised learning method and unsupervised learning is done strictly based on information obtained from raw untagged text [15]. The Expectation Maximization (EM) algorithm or Gibbs Sampling can be used to estimate the parameters of the probabilistic model [15].

D. Genetic Algorithm for WSD

The genetic algorithm for WSD is provided for the Arabian language due to writing structure [26]. The authors think the genetic algorithm is effective because it is very helpful in solving many NP hard optimization problems. Fig 3 below show the GAWSD prototype purposed by [26]:

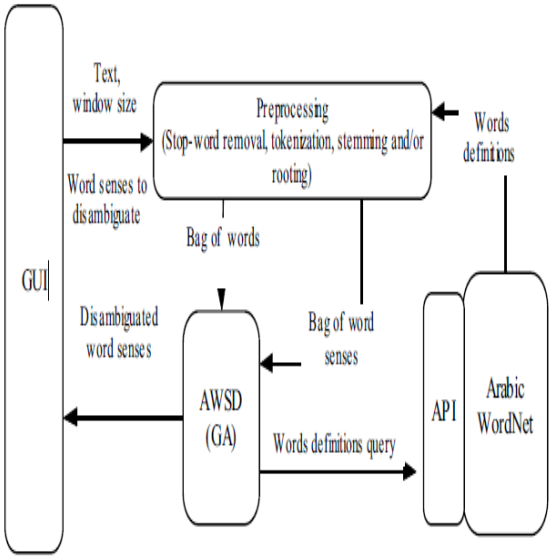

Figure 3 GAWSD prototype [26]

In this algorithm, a text T is passed through the preprocessing phase in which tokenization, stop-word removal, stemming and rooting is done. In preprocessing phase, first tokenization is done to split the text into words. After tokenization, authors have done the stop-word removal to filter out the stop words which are not important words in the text such as prepositions and articles, etc. after removing the stop words authors have performed the stemming on the remaining tokens. In stemming, it will remove the prefixes n suffixes from the word. After stemming, last step is rooting. Rooting will reduce the words to their root. Authors have used Khoja's Stemmer for rooting. The senses of each word are retrieved from Arabic Word net (AWN) as word definitions which are reduced in turn to bags of words. AWSD (GA) is used to find the most appropriate mapping from words to senses retrieved from AWN in the context T. Authors have shown that GA performs better than Naïve Bayes algorithm.

IV. APPLICATIONS

A. Information Extraction (IE)

Information Extraction is used for accurate analysis of text. Tasks like named-entity recognition (NER), acronym expansion (e.g., MP as Member of Parliament or military police), etc., can all be cast as disambiguation problems, although this is still a relatively new area. Another task is metonymy task in which systems are required to associate the appropriate metonymy with target named entities. For example, For instance, in the sentence *the BMW slowed down*, *BMW* is a car company, but here we refer to a specific car instance produced by BMW. Similarly, the Web People Search task [10] required systems to disambiguate people names occurring in Web documents, that is, to determine the occurrence of specific instances of people within texts.

B. Information Retrieval (IR)

Use either SI (MKS) or CGS as primary units. (SI units are encouraged.) If your native language is not English, try to get a native English-speaking colleague to proofread your paper. Do not add page numbers.

C. Machine Translation (MT)

WSD is important for Machine translations. There are words in one language which need to sense so that it could be translated to other language. There are some words which appear same in both language but they have different meaning. Machine translation helps in better understanding of source language and generation of sentences in target language. It also affects lexical choice depending upon the usage context.

D. Text Processing

Word Sense Disambiguation can also be used in Text to Speech translation, i.e., when words need to be pronounced in more than one way depending on their meaning. For example: “lead” can be “in front of” or “type of metal”.

E. Speech Processing and Part of Speech tagging

Speech recognition, i.e., when processing homophones words which are spelled differently but pronounced the same way. For example: “base” and “bass” or “sealing” and “ceiling”.

V. WSD FOR INDIAN LANGUAGES

Various works on WSD can be found in English and other European languages but, less amount of works in Indian languages. Various Indian Languages in which work has done are Manipuri, Tamil, Kannada, Hindi, Malyalam etc.

A. Manipuri

Manipuri is a Tibeto-Burman language, spoken in the valley of Manipur, a North-Eastern state of India. Due to the geographic location, differences in syntactic and semantic structures are noted from other Indian languages. Richard Singh and K. Ghosh[18] has recently given a proposed architecture for Manipuri Language in 2013. No work was done before this. The work presented in this paper is performed at KIIT University. The System performs WSD in two phases: training phase and testing phase. The suggested architecture to develop the Manipuri word sense disambiguation system contains five building blocks:(i) preprocessing, (ii) feature selection and generation and (iii) training, (iv) testing and (v) performance evaluation.

Raw Data is processed in the order to get the features which can be used for training and testing data efficiently. In feature Selection, a total of 6 features are taken to build feature:(i) the focus word for which the sense is to be derived,(ii) the normalized position of the word in the sentence,(iii) the previous word,(iv) the previous-to-previous word,(v) the next word,(vi) the next to next word. A 5-gram window is formed using the pair of the focus word and its context words which forms the context information. A focus word, based on the context may have different senses. Hence, in order to disambiguate the sense of the focused word, the contextual information is very much necessary and helps in predicting the correct one.

In the current study positional feature is suggested because of the lack of other relevant morphological features. As the syntactic and semantic structures of a sentence remain mostly similar for a particular language, this feature contains probable morphological information.

To generate the final input feature vector, from the database mentioned above mentioned six features are collected automatically by using the six above mentioned features and the output sense of the focus word, development of final feature vector takes place. By deriving manually the sense of the focus word, seven entries will be feed to the classifier finally. The classifier will be trained using a specified training algorithm.

During the testing, training algorithm used will be used to predict and compare the features for the test case. For predicting the sense for a test word, trained data is used and the corresponding features are generated and compared. The output generated will be tested for the accuracy and if the focus word is not found then it will be added in the training set. The predictions are later compared with the correct sense tags to perform evaluation of the current system.

B. Malyalam

Malayalam is a Dravidian language used predominantly in the state of Kerala, in southern India. It is one of the 22 official languages of India, and it is used by around 36 million people. [20] has given the first attempt for an automatic WSD in Malayalam. The author used the knowledge based approach. One approach used is based on a hand devised knowledge source and the other is using the concept of conceptual density, by using Malayalam Word Net as the lexical resource. The author has used the Lesk and Walker algorithm. In this algorithm, the author has collect all of the words from the context of a word 'w', which needs to be disambiguated and suppose this collection as 'C'. For each sense of 'w',

collect the bag of words from the Knowledge source. Let it be 'B'. Measure the overlap between 'C' and 'B'. A score of 1 will be added to that sense if any overlap is there. Highest score sense will be selected as the winner.

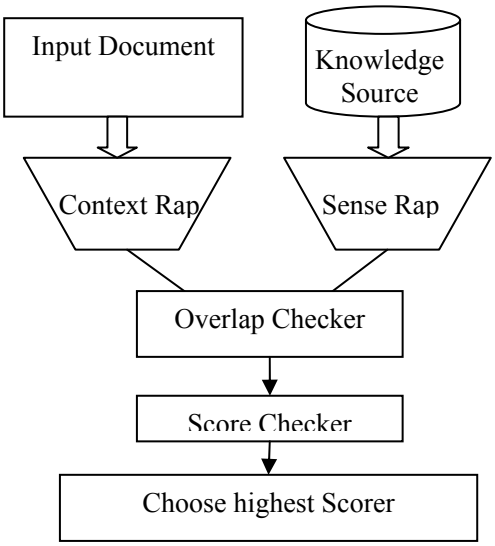

Figure 4 System Design based on Lesk and Walkers from [20]

The Second method is Conceptual density based Algorithm Design semantic relatedness between the words is taken into consideration. Semantic Relatedness can be measure in many ways. 3 metrics can be considered for measuring the semantic similarity of words using word net: Path, Depth and Information content.

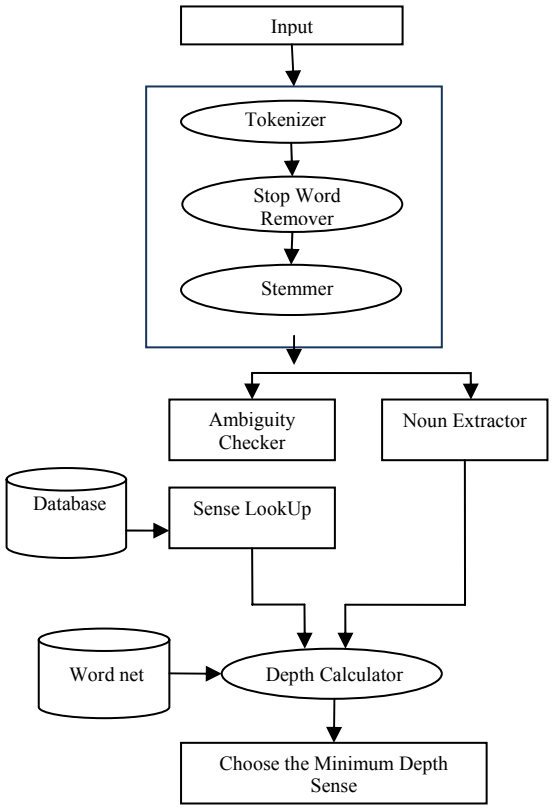

Figure 5 System Design using conceptual density from [20]

In this Algorithm depth is taken as the measurement. For each sentence: Tokenize the sentence, Remove the stop words, Perform stemming, Check for ambiguous words, If ambiguous word occurs, shift that word into one document and sense lookup is performed. Extract the nouns from the sentence and save it as a document.

For each sense in the sense lookup: Calculate the depth with each noun. If there are multiple nouns, depth of each will be added and taken as depth. The sense which results in lower Depth (highest conceptual density)is selected as the correct sense. Fig 5 is showing the system design using conceptual density given by [20].

C. Punjabi

The Punjabi language is morphologically rich. Rakesh and Ravinder [22] have given the WSD algorithm for removing ambiguity from the text document.WSD algorithm used by authors is Modified Lesk's Algorithm. There are two hypothesis that underly this approach. The first is that words appears together in a sentence can be disambiguated by assigning to them the senses that are most closely related to their neighboring words. The second hypothesis is that related senses can be identified by finding overlapping words in their definitions

REFERENCES

[1] Christopher D. Manning, and Hinrich Schutze "Foundations of Statistical Natural Language Processing," MIT Press, Cambridge, Massachusetts London, England 1999

[2] Wikipedia: http://en.wikipedia.org/wiki/Word-sense_disambiguation

[3] E. Agirre, and Philip Edmonds, "Word Sense Disambiguation: Algorithms and Applications (Text, Speech and Language Technology)," Springer-Verlag New York, Inc. Secaucus, NJ, USA 2006

[4] Roberto Navigli "Word Sense Disambiguation: A Survey," Vol. 41, Universita di Roma La Sapienza, ACM Computing Surveys, 2009

[5] M. Lesk "Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from a ice cream cone," In Proceedings of SIGDOC '86, 1986

[6] Daniel Jurafsky and James H. Martin "An Introduction to Natural Language processing, Computational Linguistics, and Speech Recognition," Pearson Education, 2008

[7] S.K. Naskar and S. Bandyopadhyay "word sense disambiguation using extended Word Net," In proceedings of ICCTA'07, 2007

[8] S. G. Kolte and S. G. Bhirud "Word Sense Disambiguation using Word Net Domains". In Proceedings of ICETET'08, 2008

[9] R. Krovetz and W. Bruce Croft "Lexical ambiguity and information retrieval," Information System, vol. 10, pp. 115-141, 1992

[10] J. Artiles, J. Gonzalo and S. Sekine "The Semeval-2007 WEPS evaluation: Establishing a benchmark for the Web people search task," In Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval, Prague, Czech Republic). pp. 64-69, 2007

[11] Ravi Sinha and Rada Mihalcea "Unsupervised Graph-based Word Sense Disambiguation Using Measures of Word Semantic Similarity," In International Conference on Semantic Computing, 2007

[12] G. Ramakrishnan, B. Prithviraj and P. Bhattacharyya "A Gloss Centered Algorithm for Word Sense Disambiguation," In Proceedings of the ACLSENSEVAL'04, Barcelona, Spain, pp. 217-221, 2004

[13] Jean Veronis "Hyperlex lexical cartography for information retrieval," Computer Speech & Language. Vol. 18, pp. 223-252, 2004

[14] Xu Li, Xiuyan Zhao, Fenglong Fan and Bai Liu "An Improved Unsupervised Learning Probabilistic Model Of Word Sense Disambiguation," In Information and Communication Technologies (WICT),World Congress 2012

[15] Francisco Tacoa, Danushka Bollegala and Mitsuru Ishizuka "A Context Expansion Method for Supervised Word Sense Disambiguation," In IEEE Sixth International Conference on Semantic Computing, 2012

[16] Wei Jan Lee and Edwin Mit "Word Sense Disambiguation by Using Domain Knowledge," Electronics Computer Technology (ICECT),3rd International Conference, 2011

[17] Richard L. Singh and Krishnendu Ghosh "Word Sense Disambiguation System in Manipuri Language," In International Conference on Computer Science and Information Technology, Goa, 2013

[18] Prity Bala "Word Sense Disambiguation Using Selectional Restriction". In International Journal of Scientific and Research Publications, 2013

[19] Rosna P. Haroon "Malayalam Word Sense Disambiguation," In Computational Intelligence and Computing Research (ICCIC),IEEE International Conference, 2010

[20] Jingbo Zhu, Huizhen Wang , Benjamin K. Tsou, and Matthew Ma, "Active Learning With Sampling by Uncertainty and Density for Data Annotations," In IEEE Transactions On Audio, Speech, And Language Processing, Vol. 18, pp. 1323-1331, 2010

[21] Rakesh Kumar, and Ravinder Khanna "Natural Language Engineering: The Study of Word Sense Disambiguation in Punjabi," In an International Journal of Engineering, Vol. 1, 2011

[22] Andres Montoyo, Armand Suarez, German Rigau and Manuel Palomar "Combining Knowledge- and Corpus-based Word-Sense-Disambiguation Methods," In Journal of Artificial Intelligence Research, 229-330 2005

[23] Zhang Zheng, and ZHU Shu, "A New Approach to Word Sense Disambiguation in MT System," In World Congress on Computer Science and Information Engineering , 2009

[24] Y.S. Alam "Lexical-Semantic Representation of the Lexicon for Word Sense Disambiguation and Text Understanding," In IEEE International Conference on Semantic Computing, 2009

[25] Jianping YU and Jian ZHANG "Word Sense Disambiguation of the English Modal Verb May by Back Propagation Neural Network," In Natural Language Processing and Knowledge Engineering, NLP-KE '08. International Conference, 2008

[26] Mohamed El Bachir Menai and Wojdan Alsaeedan "Genetic algorithm for Arabic word sense disambiguation," 13th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, 2013

A Survey of Text Summarizers for Indian Languages and Comparison of their Performance

Vishal Gupta
UIET, Panjab University, Chandigarh, India
Email: vishal@pu.ac.in

***Abstract*—Automatic text summarization is technique of compressing the original text into shorter form which will provide same meaning and information as provided by original text. The brief summary produced by summarization system allows readers to quickly and easily understand the content of original documents without having to read each individual document. The overall motive of text summarization is to convey the meaning of text by using less number of words and sentences. Summaries are of two types: Abstractive summaries and Extractive summaries. Extractive summaries involve extracting relevant sentences from the source text in proper order. The relevant sentences are extracted by applying statistical and language dependent features to the input text. On the other hand, abstractive text summaries are made by applying natural language understanding. Human beings usually make summaries in abstractive way. Moreover abstractive summaries can also involve the words or sentences which are not present in the input text. Automatic generation of abstractive summary is more difficult as compared to producing extractive text summary. This paper concentrates on survey and performance analysis of automatic text summarizers for Indian languages.**

***Index Terms*—Indian summarizers, summarizers, text summarization system**

I. Introduction

Automatic text summarization [1] is technique of compressing the original text into shorter form which will provide same meaning and information as provided by original text. The brief summary produced by summarization system allows readers to quickly and easily understand the content of original documents without having to read each individual document. The overall motive of text summarization is to convey the meaning of text by using less number of words and sentences. Text Summaries are of two types: Abstractive summaries and Extractive summaries. Extractive summaries involve extracting relevant sentences from the source text in proper order. The relevant sentences are extracted by applying statistical and language dependent features to the input text. On the other hand, abstractive text summaries are made by applying natural language understanding. Human beings usually make summaries in abstractive way. Moreover abstractive summaries can also involve the words or sentences which are not present in the input text. Automatic generation of abstractive summary is more difficult as compared to producing extractive text summary.

Automatic Text Summary generated by Microsoft Word is type of extractive summary for English language. In some of summarization systems, users can specify percentage of total source text in final summary. Worthiness of lengthy documents can quickly and easily be judged using text summarization. A summary can be labeled as good summary if it is highlighting different topics of input text and it should not have duplicate sentences. For natural language processing, making automatic text summary is largely used application of it.

Abstractive text summarization generates the summary after thoroughly understanding of input text and reconstructing the summary using less number of words and sentences in same manner as human beings usually make the summary. Abstractive text summarization is difficult because as compared to human beings, computers have limited capabilities of language understanding, so alternative methods must be considered. Difficulties of abstractive summary [1] are as: The main difficulty with abstractive summarization is representation. The abilities of automatic systems are limited by the large number of their representations and capability to produce these representation-structures—Abstractive summarizers can not produce summary of the text which their structures cannot represent. Under restricted category it is possible to formulate proper representations, but a general solution is not feasible and is dependent on general-domain semantic representations. It is not possible to build the automatic systems which can fully understand and represent the natural language of human beings.

Extractive text summarization selects the relevant sentences from input text. The relevant sentences are extracted by applying statistical and language dependent features of sentences. In most of cases in the world we prefer to make extractive text summaries due to its ease in generating text summary. Difficulties of extractive text summary [1] [2] are: 1) As compared to average summaries, extractive summaries are normally lengthy because certain sections of text which are not required in summary may also be included in it. 2) In many cases essential information is usually present across different lines, and usually extractive summaries may not collect it unless it is lengthy enough for covering all these lines.

doi:10.4304/jetwi.5.4.361-366

This paper concentrates on survey and performance analysis of automatic text summarizers for various Indian languages.

II. TEXT SUMMARIZERS FOR INDIAN LANGUAGES

Various automatic text summarization systems are commercially or non-commercially available for most of the commonly used natural languages. Most of these text summarization systems are for English and other foreign languages. Moreover, for commercial products the technical documentation is often minimal or even absent. When it comes to Indian languages, automatic text summarization systems are still lacking. Various text summarizers for Indian languages are discussed below:

Islam and Masum (2004) developed corpus oriented text summarization system 'Bhasa' for Bengali language. It is based on scoring the files of corpus in which query words are having highest frequency and then producing the summary of text documents on the basis of query words by applying vector-space-term-weighting. A tokenizer is used for tokenizing the input documents and then ranking of documents is performed with text summarization on these tokenized text documents. Tokenizer is able to determine different terms, abbreviations, tags and boundary of sentences and to denote terms, headings, titles and sentences using markups by semantic and syntactic analysis. Moreover if lines are identified using shallow-linguistic text analysis then some times text summary may have dangling anaphors [3].

Das and Bandyopadhyay (2010) developed Bengali opinion text summarizer based on given topic which can determine the information on sentiments in the input text. Then this information is aggregated for denoting text summary. It applies a model on topic-sentiment for determination and aggregation of sentiments. It is implemented for theme determination at the discourse level. Moreover aggregation is performed by clustering of theme using k-means approach and by applying theme graph representation at relational level which is ultimately applied for selection of relevant sentences in summary by using page rank standard approach. The Precision, Recall and F-Score of this approach is calculated as 72.15%, 67.32% and 69.65% respectively [4].

Sarkar (2012) proposed Bengali text summarization by sentence extraction and has investigated the impact of thematic term feature and position feature on Bengali text summarization. The proposed summarization method is extraction based. It has three major steps: (1) preprocessing (2) sentence ranking (3) summary generation. The preprocessing step includes stop-word removal, stemming and breaking the input document in to a collection of sentences. After an input document is formatted and stemmed, the document is broken into a collection of sentences and the sentences are ranked based on two important features: thematic term and position. The thematic terms are the terms which are related to the main theme of a document and having TF-IDF score above a given threshold. The positional score of a sentence is computed in such a way that the first sentence of a document gets the highest score and the last sentence gets the lowest score. Long sentences are given preference in summary A summary is produced after ranking the sentences based on their scores and selecting K-top ranked sentences, when the value of K is set by the user. To increase the readability of the summary, the sentences in the summary are reordered based on their appearances in the original text [5].

Sarkar (2012) proposed another approach for summarizing Bengali news documents. It describes a system that produces extractive summaries of Bengali news documents. The ultimate objective of produced summaries is defined as helping readers to determine whether they would be interested in reading a particular document. To this end, the summary aims to provide a reader with an idea about the theme of a document without revealing the in-depth detail. The approach presented here has four major steps (1) preprocessing (2) extraction of candidate summary sentences (3) ranking the candidate summary sentences (4) summary generation. The proposed approach defines TF*IDF, position and sentence length feature in more effective way that helps in improving the summarization performance. The experimental results show that this system performs better than the lead baseline and a more sophisticated baseline that uses TF*IDF and position features both [6].

Kumar and Devi (2011) proposed Tamil language summarization system for scoring of sentences in summary using graph theoretic scoring technique. This system uses statistics of frequency of words and a term positional and weight-age calculation by string pattern for scoring of sentences [7].

Kallimani et al. (2010) proposed a text summarizer for Kannada i.e. "AutoSum" a named IR system using Text Summarization of some Regional Language in India. This system processes the input text and then decides which lines are relevant and which lines are not relevant. User interaction in this system is command based interaction. In it, text is summarized on console. The output summary of this system can be produced either in simple text or in hyper text markup language. If hyper text markup language is used in output then relevant lines are highlighted. It begins its summarization task when input text is given by user which is having 03 steps i) Command is given by user on the terminal ii) In further stages, the input moves through the system and summary is produced iii) The resulting text is sent to the terminal after summary is made or the results of summary can be highlighted in the web browser. This system uses adjectives, adverbs and nouns as key terms. The score value of each term in every sentence is determined and summed up to the score of that sentence. Every line is assigned a score based on the key terms in it [8].

Jayashree et al. (2011) proposed a text summarization system for Kannada named "Kannada text Summarizer based on Key terms Extraction". This system takes pre-classified Kannada documents obtained from online web resources and identifies the thematic words from these documents by mixing GSS (Galavotti, Sebastiani, Simi)

coefficients and Inverse-Document-Frequency techniques with Term-Frequency and then apply these extracted keywords for making summary [9].

Jayashree et al. (2012) proposed another pre-classified documents summarizer for Kannada by scoring of sentences which retrieves key terms from Kannada documents, by combining GSS (Galavotti, Sebastiani, Simi) coefficients and Inverse-Document-Frequency techniques with Term Frequency for retrieving key terms and then applies them for summarizing the text. Overall motive of this technique is to give weight-age to every term of a line, the final weight of a line is the addition of weight-age of each term in that line. Finally it selects n sentences based on sentence scores. Database is specially built for this purpose by selecting a document of a given category. Kannada text files are taken from Webdunia. Webdunia is a special web portal in Kannada that is used for latest News, Entertainment News, Sports related news, Jokes and Shopping etc. Summary is produced based on number of sentences in the input given by user. Then summary evaluation is done by comparing the human produced summary with system produced summary. Other motive of this technique is to extract different features after removing the stop words from input text. Moreover for elimination of stop words from input a new approach has been used which identifies structurally similar type of terms in any text document [10].

Patel et al. (2007) proposed a technique to text summarization for English, Hindi, Gujarati and Urdu documents. The algorithm is based on structural and statistical (rather than semantic) factors. The algorithm has been applied on document understanding conference (DUC) data English documents and various newspaper articles for other languages with corresponding stop words list and modified stemmer. To test the language independence of the summaries generated by this summarizer, it has been tested on 70 news articles of Hindi leading dailies, 50 articles of Gujarati literature and 75 new articles of Urdu from BBC web site. In almost every case, it gives degree of representative ness more than 80% [11].

Garain et al. (2006) proposed text summarization of compressed text pictures for Indian language. This system is used to summarize JBIG2 coded text pictures without using optical character recognition. Compressed pictures are decompressed and then sentences and terms are marked. Four features are determined at the level of sentences. These features are (1) Feature1: Length of sentences (2) Feature2: Position of sentences in each paragraph (3) Feature3: Thematic term features (4) Feature4: Title terms. For values of these features, lines are treated as summary lines or non summary lines. Finally this system produces a set of summary sentences. Moreover, within summary sentences are further ranked. In experiments author only considers Indian language text images. The sentence selection efficiency of this approach is 56% calculated against human generated summary [12].

Automatic text summarization software for Hindi [13] text has been commercially developed by CDAC (Centre for development of advance computing) Noida. This system has applied statistical approach, Language based approach along with heuristic approach for developing text summarization system for Hindi. This summarization software includes 1) Features based on Statistics: Term, Pair of Terms, Particular Cue terms, count, determining Value of Threshold, location of sentences and proper location scheme etc. 2) Analyzing language oriented features: Determine noun terms, terms existing together, finding stop-words, terms which are functional in nature. 3) Language oriented Psycho features: Unique or duplicate terms. 4) Feature belonging to Heuristics: sentence belonging to Title, Location, Number of words in a sentence and Table of contents etc. 5) Giving weight-age, ranking of lines etc. [13]

Gupta et al. (2012) proposed of Punjabi text summarizer. It makes extractive summary for Punjabi text by extracting the important lines based on language oriented features and features belonging to statistics of text. Every line of input text is treated as vector of different features like sentence relative length, Punjabi cue terms, Punjabi terms belonging to nouns, terms belonging to common nouns of English and Punjabi, Punjabi named entities, location of lines, Term-Frequency and Inverse-Sentence-Frequency scores for extracting thematic terms, existence of numeric data in lines etc. Duplicate sentences are eliminated in the pre processing phase. Weight-age of sentence-features which are influencing the different lines are calculated by applying regression which is a weight learning method. For each sentence, the score values of all features are calculated and final score values of all sentences are determined using equation of features and weights. Finally Punjabi sentences with top scores are selected in same order as in input text at given CR (compression ratios). In case of Punjabi news articles, Punjabi text summarizer is showing F-measure 97.87%, 95.32 and 94.63% respectively at 10%, 30% and 50% CR (compression ratios) and in case of Punjabi stories, this system shows F-measure 81.78%, 89.32% and 94.21% respectively at 10%, 30% and 50% CR (compression ratios) [14][15][21].

Kallimani et al. (2012) proposed a new technique for summarizing the longer text documents by considering one of the South Indian regional languages (Kannada). It deals with a single document summarization based on statistical approach. The purpose of summary of an article is to facilitate the quick and accurate identification of the topic of the published document. The objective is to save prospective readers' time and effort in finding the useful information in a given huge article. Moreover in case of Kannada summary, the total frequency of terms in system produced summary is more as comparative to summary produced by human and also %age term frequency is more in both the summaries because the size of summary is increased. This is clear that out of 04 lines in the 20 % summary, 75% lines i.e. 03 lines are common, Out of 05 lines in 30% summary, 80% lines i.e. four lines are common and out of 06 lines in the 40 % summary, 83.33% lines i.e. 05 lines are common. It shows that with

increase in percentage of summary size, the number of common lines have also increased [16].

Banu et al. (2007) proposed text summarizer for Tamil documents using technique of semantic graph by identifying Subject Object Predicate from individual lines for making semantic-graph of source text document and its corresponding summary generated by human experts [20].

Banu (2010) proposed another technique for summarizing documents of Tamil by using approach of sub graph for selecting lines from source document treated as text summary or another technique for generating a generic summary of document. In this system, syntax of language neutral, which is the system for representing the natural language lines has been applied for compressing the text documents. It has used syntactic analysis of the source text which makes a analysis of logical form has been used for every line. Triples of subject object predicate are selected from individual lines to generate a semantic graph of source document and its corresponding summary generated by human experts. To triples of SOP Semantic Normalization is used for reducing the frequency of nodes of semantic graph of source document. Classifier has provided training by using leaning technique based on support vector machine learning, for identifying triples of SOP from semantic graph of document which belongs to actual summary. Then this classifier is used to extract automatic summaries from test documents [17].

Keyan (2012) proposed multi-lingual (Tamil and English) multi-document summarization by neural networks. The system involves three steps. In first step, the sentences of the documents are converted into vector form. In the second step weight values are assigned to vector form based on sentence features. Depend on sentence weight value, single document summarization is done. The output of single document summarization is used as an input for multi-document Summarization. Final step is a sentence selection, in which output summary is selected based on the similarity and dissimilarity measures. Sentence similarity and dissimilarity measures are used to compare the sentences. From that, resultant summary is produced. The proposed system can be able to summarize both Tamil and English online news papers. [18]

Islam et al. (2007) proposed text summarizer for Bangla using text extraction based summarization technique and reported average highest score of 8.4 (on 0-10 scale) at 40% compression ratio [19].

III. Performance Comparison in Indian Summarizers

Garain et al. (2006) [12] proposed method for automatic summarization of JBIG2 coded textual images for Bengali text without optical character recognition (OCR) with efficiency of about 56% when judged against summarization generated by human. Islam et al. (2007) [19] proposed text summarizer for Bangla using text extraction based summarization technique and reported average highest score of 8.4 (on 0-10 scale) at 40% compression ratio. Das et al. (2010) [4] proposed topic-Based Bengali Opinion Summarization with Precision of 72.15%, Recall of 67.32% and F-measure of 69.65%. Bengali text summarization by sentence extraction is another Bengali text summarization system developed by kamal sarkar [5] and had investigated the impact of thematic term feature and position feature on Bengali text summarization with Average Unigram based Recall Score 0.4122. Automatic text summarization software for Hindi text [13] had been commercially developed by CDAC (Centre for development of advance computing) Noida. Statistics based technique, language oriented & heuristic technique had been applied for this text summarizer for Hindi. Patel et al. (2007) proposed a language independent approach to multilingual text summarization for English, Hindi, Gujarati and Urdu [11] documents based on structural and statistical (rather than semantic) factors with efficiency of 82%. Regarding Kannada, Text summarization system for Kannada named "Information Retrieval by Text Summarization for an Indian Regional Language" [8] had been proposed in 2010 using keywords extraction by taking nouns, adjectives and adverbs as keywords. Another text summarization system for Kannada named "Document Summarization in Kannada using Keyword Extraction" [9] had been proposed in 2011 using extracted key words from pre-categorized Kannada documents collected from online resources with relevant score of 0.7 for literature, 0.8 for entertainment, 0.8 for astrology and 0.76 for sports documents. Banu et al. (2007) [20] proposed text summarizer for Tamil documents using technique of semantic graph by identifying Subject Object Predicate from individual lines for making semantic-graph of source text document and its corresponding summary generated by human experts. Another Tamil text extraction system for an agglutinative language [7] had been introduced in 2011 by proposing an efficient algorithm for sentence ranking based on a graph theoretic ranking model applied to text summarization task with ROUGE-1 score 0.47. TABLE I shows the comparison of performance of some of existing summarizers for Indian languages [21].

TABLE I.
PERFORMANCE COMPARISON OF EXISTING INDIAN SUMMARIZERS [21]

Summarization systems	Performance comparison of existing summarizers for other Indian Languages	
	Accuracy (In %)	Test used
Punjabi Text Summarization System [14][15] [21]	For Stories: 89.32% (At 30% Compression Ratio) For News Documents: 95.32% (At 30% Compression Ratio)	F-Score
Bengali Summarizer using Textual Images [12]	56%	Efficiency
Bengali Summarizer using Text Extraction [19]	84% (At 40% Compression Ratio)	Efficiency

Topic based Bengali Opinion Summarizer [4]	69.65%	F-Score
Multi Lingual Summarizer for English, Hindi, Gujarati & Urdu [11]	82%	Efficiency
Document Summarizer for Kannada [9]	For Literature: 70% For Entertainment: 80% For Sports: 76%	Efficiency
Summarization from large Kannada documents using a novel approach [10]	At 30% Compression ratio: 80% At 40% Compression Ratio: 83.33%	Efficiency
Tamil text extraction system for an agglutinative language [7]	Score : 0.47	ROUGE-1

IV. CONCLUSIONS

Although various automatic text summarization systems are commercially or non-commercially available for most of the commonly used natural languages for English and other foreign languages, but when it comes to Indian languages, automatic text summarization systems are still lacking. But now days lot of research is going on for Indian regional languages and after comparing the performance of various Indian summarizers for Hindi, Punjabi, Kannada, Tamil, Gujarati and Bengali, we can conclude that they are reasonably performing well over wide range of text dataset including news documents, stories, and documents related to literature, sports and entertainment.

REFERENCE

[1] V. Gupta and G. S. Lehal, "A Survey of Text Summarization Extractive Techniques," *International Journal of Emerging Technologies in Web Intelligence*, vol. 2. , pp. 258-268, 2010.

[2] J. Cheung, "Comparing Abstractive and Extractive Summarization of Evaluative Text : Controversiality and Content Selection," *B. Sc. (Hons.) Thesis,* Department of Computer Science of the Faculty of Science, University of British Columbia, 2008.

[3] T. Islam and S. M. A. Masum, "Bhasa: A Corpus Based Information Retrieval and Summarizer for Bengali Text," *Macquarie University*, Sydney, Australia, 2004.

[4] A. Das and S. Bandyopadhyay, "Topic-Based Bengali Opinion Summarization", *International Conference COILING '10*, Beijing, pp. 232–240, 2010.

[5] K. Sarkar, "Bengali text summarization by sentence extraction," *In Proceedings of International Conference on Business and Information Management(ICBIM-2012)*, NIT Durgapur, pp. 233-245, 2012.

[6] K. Sarkar, "An approach to summarizing Bengali news documents," *In proceedings of the International Conference on Advances in Computing, Communications and Informatics*, ACM, pp. 857-862, 2012.

[7] S. Kumar, V. S. Ram and S. L. Devi, "Text Extraction for an Agglutinative Language," *Proceedings of Journal: Language in India*, pp. 56-59, 2011.

[8] J. S. Kallimani, K.G. Srinivasa and B. R. Eswara, "Information Retrieval by Text Summarization for an Indian Regional Language," *In Proceedings of International Conference on Natural Language Processing and Knowledge Engineering*, pp. 1-4, 2010.

[9] R. Jayashree, K. M. Srikanta and K. Sunny, "Document Summarization in Kannada using Keyword Extraction," *Proceedings of AIAA 2011,CS & IT 03*, pp. 121–127 , 2011.

[10] R. Jayashree, "Categorized Text Document Summarization in the Kannada Language by Sentence Ranking," *Proceedings of 12th International Conference on Intelligent Systems Design and Applications (ISDA)*, pp. 776-781, 2012.

[11] A. Patel, T. Siddiqui and U.S. Tiwary, "A language independent approach to multilingual text summarization," *Proceedings of conference RIAO '07*, Pittsburgh PA, U.S.A., 2007.

[12] U. Garain, A. K. Datta, U Bhattacharya and S.K. Parui, "Summarization of JBIG2 Compressed Indian Textual Images," *Proceeding of 18th International Conference on Pattern Recognition (ICPR'06)*, IEEE, Kolkata, India, 2006.

[13] http://cdacnoida.in/SNLP/digital_library/text_summ.asp

[14] V. Gupta and G. S. Lehal, "Complete Preprocessing Phase of Punjabi Language Text Summarization," *International Conference on Computational Linguistics COLING'12*, IIT Bombay, India, pp. 199-205, 2012.

[15] V. Gupta and G. S. Lehal, "Automatic Punjabi Text Extractive Summarization System," *International Conference on Computational Linguistics COLING '12*, IIT Bombay, India, pp. 191-198, 2012.

[16] J. S. Kallimani, K. G. Srinivasa and B. R. Eswara, "Summarizing News Paper Articles: Experiments with Ontology Based, Customized, Extractive Text Summary and Word Scoring", *Journal of Cybernetics and Information Technologies*, Bulgarian Academy of Sciences, vol. 12, pp. 34-50, 2012.

[17] M. Banu, C. Karthika, P. Sudarmani and T.V. Geetha, "Tamil Document Summarization Using Semantic Graph Method", *International Conference on Computational Intelligence and Multimedia Applications*, IEEE, pp. 128-134, 2007.

[18] M.. K. Keyan and K.G. Srinivasagan, "Multi-Document and Multi-Lingual Summarization using Neural Networks", *Proceedings of International Conference on Recent Trends in Computational Methods, Communication and Controls*, pp. 11-14, 2012.

[19] N. Uddin and S. A. Khan, "A Study on Text Summarization Techniques and Implement Few of Them for Bangla Language", *Proceedings of international conference on Computer and information technology*, IEEE, pp. 1-4, 2007.

[20] M. Banu, C. Karthika, P Sudarmani and T.V. Geetha, "Tamil Document Summarization Using Semantic Graph Method", *Proceedings of International Conference on Computational Intelligence and Multimedia Applications*, pp. 128-134, 2007.

[21] V. Gupta and G.S. Lehal, "Automatic Text Summarization for Punjabi Language," *International Journal of Emerging Technologies in Web Intelligence*, vol. 5, pp. 257-271, 2013.

Dr. Vishal Gupta is Senior Assistant Professor in Computer Science & Engineering department at University Institute of Engineering & Technology, Panjab University Chandigarh. He has done his Ph.D. and M.Tech. in Computer Science & Engineering from Punjabi University Patiala in 2013 and 2005 respectively. He is among University toppers. He secured 82% Marks in M.Tech. Vishal did his BTech. in CSE from S.B.S. College of Engineering & Technology Ferozepur in 2003. He is Young Scientist Award Winner-2013 in Engineering & Technology at Punjab Science Congress. He has written around fifty research papers in international and national journals and conferences. He has developed a number of research projects in field of NLP including topic tracking, keywords extraction, named entity recognition, synonyms detection, automatic question answering and text summarization etc. One of his research paper on Punjabi language text processing was awarded as best research paper by Dr. V. Raja Raman at an International Conference at Panipat. He is also a merit holder in 10^{th} and 12^{th} classes of Punjab School education board.

A Survey on Sentiment Analysis and Opinion Mining Techniques

Amandeep Kaur
University Institute of Engineering and Technology, Chandigarh, India
Email: amandeepk.cql@gmail.com

Vishal Gupta
University Institute of Engineering and Technology, Chandigarh, India
Email: vishal@pu.ac.in

***Abstract*—Sentiment Analysis (SA), an application of Natural Language processing (NLP), has been witnessed a blooming interest over the past decade. It is also known as opinion mining, mood extraction and emotion analysis. The basic in opinion mining is classifying the polarity of text in terms of positive (good), negative (bad) or neutral (surprise). Mood Extraction automates the decision making performed by human. It is the important aspect for capturing public opinion about product preferences, marketing campaigns, political movements, social events and company strategies. In addition to sentiment analysis for English and other European languages, this task is applied on various Indian languages like Bengali, Hindi, Telugu and Malayalam. This paper describes the survey on main approaches for performing sentiment extraction.**

***Index Terms*— Natural Languages processing, Sentiment Analysis, Indian languages.**

I. Introduction

"*Sentiment analysis or opinion mining refers to the application of natural language processing, computational linguistics and text analytics to identify and extract subjective information in source materials*"(Source:Wikipedia). Opinion mining/sentiment analysis is a multidisciplinary and multifaceted Artificial intelligence problem. Its aim is to minimize the gap between human and computer. Thus, it is collection of human intelligence and electronic intelligence for mining the text and classifying user sentiments, likes, dislikes and wishes. The user generated content is available in various forms such as web logs, reviews, news, discussion forums. Web 2.0 & 3.0 has provided a platform to share the feelings and views about the products and services. The basic of this problem can be better explained using the following review by a user about a car:

"*I bought a car a few days ago. It had a comfortable suspension set-up but it did not provide stable and safe feel. It delivers good fuel economy but does feel lethargic engine. It has better quality interiors which look sporty*."
The above review is analyzed for opinion mining and extracted views are visualized in Fig 1(a). Similarly comparison of two or more can also be evaluated as represented in Fig 1(b).

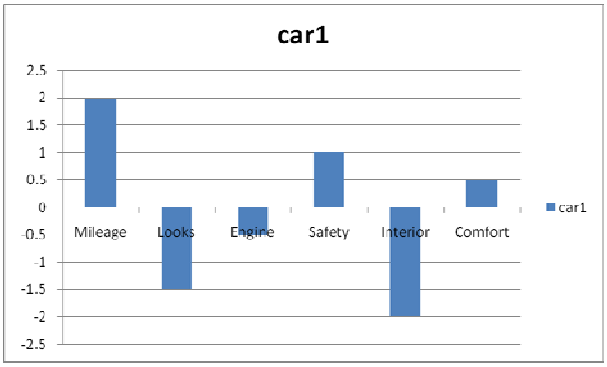

Figure 1(a). Visualization of summary of opinions on a car

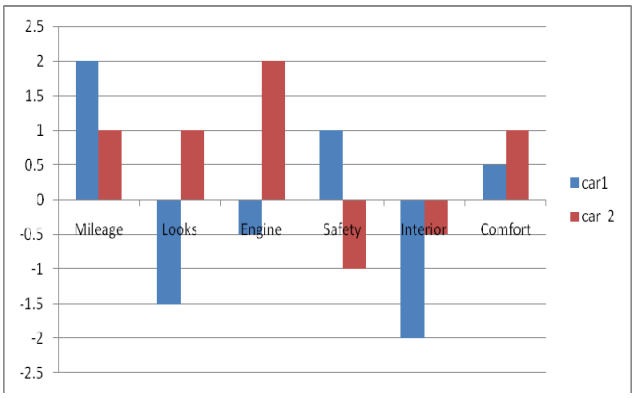

Figure 1(b). Visual comparison of two cars.

Social network revolution plays a crucial role in gathering information containing public opinion. To obtain subjective and factual information from this information, public opinions are extracted. Thus, it is the process to predict hidden information about user's intensions, likeliness and taste. These social networking sites generate enormous data up to terabytes per week. The statistics mentioned in Table I shows the rate at which amount of user generated data is increasing:

doi:10.4304/jetwi.5.4.367-371

TABLE I.
Statistics of user generated data

Facebook	Twitter	Google
850 million users 250 million photos 2.7 billion likes	465 million accounts 175 million tweets	90 million users 675000 users per day

Popular approaches used for sentiment analysis are: Popular approaches used for sentiment analysis are:

- **Subjective lexicon** – is a list of words where each word is assigned a score that indicates nature of word in terms of positive, negative or objective.
- **Using N-Gram modeling-** for given training data , we make a N-Gram model(uni-gram, bi-gram, tri-gram or combination of these)for classification.
- **Machine learning –** perform the supervised or semi- supervised learning by extracting the features from the text and learn the model.

II. Process of Sentiment Analysis for Text

The process of sentiment analysis is divided into five steps [11]: Process of Sentiment Analysis for Text (Lexicon Generation), Subjectivity Detection, Sentiment polarity Detection, Sentiment Structurization, Sentiment Summarization-Visualization-Tracking.

A. Process of Sentiment Analysis for Text (Lexicon Generation)

In this phase, sentiment lexicon is created to acquire the knowledge about sentiments. According to previous studies, prior polarity should be attached at each lexicon level. To develop SentiWordNet(s), Manual and Automated processes have been attempted for multiple languages.

Related Work: (Stone, 1966) Philip Stones developed General Inquirer system, was the first milestone for extracting textual sentiment. It was based on the manual database containing set of positive or negative orientations and the input words are compared with database to identify their class such as positive, negative, feel, pleasure [5].

(Brill, 1994) Brill Tagger depicted the semantic orientation for verbs, adverb, noun and adjective. After extracting these phrases, PMI algorithm (Turney, 2002) is applied to identify their semantic polarity [31][7].

(Hatzivassiloglou et al., 1997) Hatzivassiloglou was the first to develop empirical method of building sentiment lexicon for adjectives. The key point is based on the nature of conjunctive joining the adjectives. A log-linear regression model is provided with 82% accuracy [8].

(Turney, 2002) For the classification of positive and negative opinion, Peter Turney proposed the idea of Thumbs Up and Thumbs Down. For better problem formalization, there was the necessity of an automated system, which could be employed for electronic documents. For consecutive words and their polarity, Turney came up with an algorithm to extract Point wise Mutual Information(PMI).Experiments were conducted on movie review corpus and polarity is referred to as "thumbs up" for positive and "thumbs down" for negative [7].

(Pang et al.,2002) Pang build sentiment lexicon for movie reviews to indicate positive and negative opinion. This system motivated the other machine learning approaches like Support Vector Machine, Maximum Entropy and Naive Bayes[10].

(Kamps et al., 2004) Kamps, Marx, Mikken and Rijke tried to identify subjectivity of adjectives in Word Net. In this research, they classified adjectives into four major classes and used base words (to measure relative distance) depending on the class. For class *Feeling* their base words were “happy” and “sad”, for class *Competition* their base words were “pass” and “fail”, etc. Based on this idea, they gathered a total of 1608 words in all four classes with average accuracy of 67.18% for English [24].

(Gamon et al., 2005) proposed similar method as by (Turney, 2002).Machine Learning based technique is used with input of some seed words. This classifier is based on assumption that the words with same polarity co-occur in one sentence but words with different polarity cannot [11].

(Read, 2005) have stated three different problems in the area of sentiment classification: Time, Domain and Topic dependency of sentiment orientation. It has been experimented that associative polarity score varies with time [12].

(Denecke, 2009) introduced uses of SentiWordNet in terms of prior polarity scores. The author proposed two methods: rule-based and machine learning based. Accuracy of rule-based is 74% which is less than 82% accuracy of machine learning based. Finally, it is concluded that there need more sophisticated techniques of NLP for better accuracy [13].

(Mohammad et al., 2009) proposed a technique to increase the scope of sentiment lexicon. It includes the identification of individual words as well as multi-word expressions with the support of a thesaurus and a list of affixes. The technique can be implemented by two methods: antonymy generation and Thesaurus based. Hand-crafted rules are used for antonymy generation. Thesaurus method is based on the seed word list which means if a paragraph has more negative seed words than the positive ones, then paragraph is marked as negative [14].

(Mohammad and Turney, 2010) developed Amazon Mechanical Turk, an online service by Amazon, to gain human annotation of emotion lexicon. But there was the need of high quality annotations. Various validations are provided so that erroneous and random annotations are rejected, discouraged and re-annotated. Its output provides 2081 tagged words with an average tagging of 4.75 tags per word [10].

B. Subjectivity Detection

Sentiment analysis categorizes the text at the level of subjective and objective nature. Subjectivity means the

text contains opinion and objectivity means text contains no opinion but contains some fact. In precise form, Subjectivity can be explained as the Topical Relevant Opinionated Sentiment [9].Genetic Algorithm (Das, 2011) achieved a good success for the subjectivity detection for Multiple Objective Optimization [27].

Some example-

1. Subjective- *The car is comfortable.* (This sentence expresses the feeling which is an opinion; hence it is of subjective nature)

2. Objective- *Maruti launched a new car*. (This sentence contains a fact).

Related Work: (Wiebe, 2000) defined the concept of subjectivity in an information retrival perspective which explains the two genres subjective and objective [9].

(Aue and Gamon, 2005) told that subjectivity identification is a context dependent and domain dependent problem which replaces the earlier myth of using sentiwordnet or subjectivity word list etc. as prior knowledge database [17].

(Das and Bandyopadhyay, 2009) explained the techniques for subjectivity based on Rule-based, Machine learning and Hybrid phenomenon [2].

The idea of collecting subjectivity clues helped in the subjectivity detection. This collection includes entries of adjectives (Hatzivassiloglou and Mackeown, 1997) and verbs(Wiebe,2000) and n-grams(Dave et.al, 2003) [8][9][18].

The detail of sentiment analysis and subjectivity detection is given by Wiebe in 1990 [16].

Methods of identification of polarity are explained in (Aue and Gamon,2005) [17].

Some algorithms like Support Vector Machine (SVM), Conditional Random Field(CRF) (Zhao et. al,2008) have been used for clustering of opinions of same type[6].

C. Sentiment Polarity Detection

The sentiment polarity detection means classifying the sentiments into semantic classes(Turney et.al.,2002) such as positive, negative or neutral or other emotion classes like anger, sad, happy, surprise [7].

SentiWordNet is most popular to be used as polarity lexicon. Another technique used for polarity detection is Network Overlap Technique [27]. In this, contextual prior polarity is assigned to each sentiment word.

Related Work: Since the last few years, Tweet Feel http://www.tweetfeel.com) and Twitter Sentiment Analysis Tool(http://twittersentiment.appspot.com/) are available. To satisfy the end users, level of research should be raised [19].

(Cambria et al., 2011) developed a new paradigm known as Sentic Computing. This research is based on a common sense and emotion representation. It have been used for short texts to infer emotional states over the web [20].

Concept Net, a semantic network was introduced with approx 10000 concepts and more than 72000 features extracted from Open mind corpus. In the sentic computing, four dimensions are taken as basis to classify the affective states: Sensitivity, Attention, Pleasantness and Aptitude.

D. Sentiment Structurization

Sentiment Analysis explained till now is not sufficient to satisfy the needs of end user, because the latter is not interested in binary output in terms of positive or negative but interested in aspectual sentiment classification. Aspectual can be explained as relative information. For example, a social worker may be interested to know the change in the society before and after implementation of his scheme. So, a sentiment analysis system should be understand and identify the aspectual sentiments present in the text.

For this problem, sentiment structurization technique has been proposed by Das(Das 2010).This technique is based on 5W (Why, Where, When, What, Who).The drawback of 5Ws is that it may lead to label bias problem. To solve this Problem Maximum Entropy Model (MEMM) was introduced.

Related Work: (Bethard et. al., 2006) have introduced the automatic identification of opinions from question answering.

(Bloom et al., 2007) describes Appraisal Theory (Martin and white ,2005).The system classifies the opinions into three types: affect, appreciation or judgment.

(Yi et.al. 2006) introduced e sentiment analyzer for online text documents.

(Zhou et. al., 2006) have introduced the architecture for blogosphere to get the summarized text.

E. Sentiment Summarization-Visualization-Tracking

One of the main needs of end users is the aggregation of data. After the Literature survey, following summarization attempts are found:

- Polarity wise (Hu, 2004),(Yi and Niblack, 2005),(Das and Chen, 2007)
- Topic wise(Yi et al.,2003),(Pang and Lee, 2004) ,(Zhou ,2006)

Visualization and Tracking is the last phase of sentiment analysis which is most important to satisfy the needs of end users. In this phase, visual sentiments are generated which are further tracked with polarity wise graph according to some dimension or combination of dimensions. The final graph for tracking is created with a timeline.

III. Sentiment Analysis for Indian Languages

There is comparatively less research has been done for Indian languages.

(Das and Bandyopadhyay, 2010) suggested a computational technique for developing SentiWordNet(Bengali) using English-Bengali bilingual dictionary and English Sentiment Lexicons [21].

(Das and Bandyopadhyay, 2010)- The author introduced four approaches to predict the polarity of a word. In the First strategy; an interactive game is provided which identify the polarity of the words. In the Second strategy, a bi-lingual dictionary is developed for

English and Indian Languages. In the third strategy, word net expansion is done using antonym and synonym relations. In the fourth approach, a pre-annotated corpus is used for learning [1].

(Das and Bandyopadhyay, 2010)- developed the method for tagging using the Bengali words. Classification of words is performed into six emotion classes (happy, sad, surprise, fear, disgust, anger) according to three categories of intensities (low, general and high) [22].

(Draya et.al., 2009) performed blog sentiment analysis to extract domain specific adjectives [23].

(Joshi, et.al. 2010) used two lexical resources: English-Hindi Word Net Linking and English SentiWordNet and created H-SWN(Hindi-SentiWordNet) [28].

(Kim and Hovy,2004) Kim and Hovy did the research work for Hindi Language but their work is restricted to synonyms [24].

(Narayan, et.al., 2002) Hindi Subjective Lexicon and hindi WordNet is used for the identification of semantic orientation of adjectives and adverbs [25].

(Pang, et.al. 2002) sentiment classification is done at document level using syntactic approach of N-Grams. This method is used to perform machine learning [10].

(Rao and Ravichandran, 2009) performed the classification of bi-polar nature [26].

(Turney, 2002) Turney used semantic mining for binary classification and also did research on part of speech (POS) information. It is document level and review level sentiment analysis [7].

IV. Conclusion and Future Work

A. Conclusion

Sentiment Analysis has lead to development of better products and good business management. This research area has provided more importance to the mass opinion instead of word-of-mouth.

In the conclusion, it has been proved that coverage expansion is good by using automatic processes where as prior polarity assignment is credible by using manual methods. SentiWordNet has been successfully generated for Hindi, Telugu and Bengali and global SentiWordNet has been generated for 57 languages. The 5W structure is more acceptable solution across domains. The success of Genetic Algorithm can be estimated by the fact that the system based on this algorithm has highest performance till date for Bengali and English.

For Indian languages, scarcity of resources has become the biggest issue. Research is going on for building subjective lexicon and datasets for Indian languages.

B. Future Work

As SentiWordNet has been generated for various languages, there can be further research on cross-lingual sentiment sense mapping. It is necessary to update the prior polarity scores according to various dimensions. The future research can be to develop web service API so that latest prior polarity scores can be accessed. The concept of Artificial intelligence can be used for further research that can mimic human biological mechanism. In the future, Event Tracking can also be implemented using the concept of 5W structure.

There are 22 official languages and 13 languages having more than 10 million speakers in India. Research is going on these languages but successful results are obtained in few languages such as Bengali, Hindi and Malayalam. There are many languages which are unexplored. Multilingual dictionary is available for English and 11 Indian languages (Hindi, English, Marathi, Bengali, Gujarati, Oriya, Malayalam, Urdu, Punjabi, Tamil, and Telugu).In future; subjective lexicon can be developed for the unexplored languages which does not have a word net. The basic resources like parsers, named entity recognizers, morphological analyzers, and part of speech tagger need to be improved to reach the state of accuracy.

References

[1] A. Das and S. Bandyopadhay, "SentiWordNet for Indian languages," *Asian Federation for Natural Language Processing*, China, pp. 56–63, August 2010.

[2] A. Das and S. Bandyopadhay, "Subjectivity Detection in English and Bengali: A CRF-based Approach," In Proceedings of the 7th International Conference on Natural Language Processing, Macmillan 2009.

[3] H. Tang, S. Tan and X. Cheng, " A survey on sentiment detection of reviews", In Proceedings of the Expert Systems with Applications 36, Elsevier Ltd., Beijing, 2009

[4] N. Mohandas, J.P. Nair and G.V, " Domain Specific Sentence Level Mood Extraction from Malayalam Text", In Proceedings of the International Conference on advances in Computing and Communications, IEEE, 2012.

[5] P.J. Stone, "The General Inquirer: A Computer Approach to Content Analysis", The MIT Press, 1966.

[6] J. Zhao, K. Liu and G. Wang, "Adding Redundant Features for CRFs- based Sentence Sentiment Classication" In Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp.117-126, 2008.

[7] P. Turney, "Thumbs up or thumbs down? Semantic orientation Applied to Unsupervised Classification of Reviews" In Proceedings of the Association for Computational Linguistics, pp.417-424, Philadelphia, 2002.

[8] V. Hatzivassiloglou and K. R. McKeown, "Predicting the Semantic Orientation of Adjectives", In Proceedings of the 35th Annual Meeting of the ACL and 8th Conference of the European Chapter of the ACL, pp.174-181, Madrid, 1997.

[9] J. M. Wiebe, " Learning Subjective Adjectives from Corpora", In Proceedings of the 17th National Conference on Artificial Intelligence and 12th Conference on Innovative Applications of Artificial Intelligence, pp. 735-740, Mento Park, 2000.

[10] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up? Sentiment Classification using Machine Learning Techniques" In Proceedings of the Empirical Methods on Natural Language Processing, pp. 79-86, Pennsylvania, 2002

[11] M. Gamon, A. Aue, S. Corston-Oliver and E. Ringger, "Pulse: Miming Customer Opinions from Free Text", In Proceedings of the International Symposium on Intelligent Data Analysis, pp. 121-132, 2005.

[12] J. Read, "Using Emotions to Reduce Dependency in Machine Learning Techniques for Sentiment

Classification", In Proceedings of the Student Research Workshop, pp. 43-48, Arbor, 2005.
[13] K. Denecke, "Are SentiWordNet Scores Suited For Multi-Domain Sentiment Classification", In Proceedings of the 4th International Conference on Digital Information Management, pp. 33-38, Ann Arbor, 2009.
[14] S. Mohammad, B. Dorr, and C. Dunne, " Generating High-Coverage Semantic Orientation Lexicons fom Overly Marked Words and a Thesaurus", In Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing,pp. 599-608, 2009.
[15] S. Mohammad and P. Turney, "Emotions Evoked by Common Words and Phrases: Using Mechanical Turk to Create an Emotion Lexicon" In Proceedings of the Workshop on Computational Approaches to Analysis and Generation of Emotion, pp. 26-34, California, 2010.
[16] J. M. Wiebe, "Recognizing Subjective Sentences: A Computational Investigation of Narrative Text", Doctoral Thesis. UMI Order Number: UMI Order No. GAX90-22203., State University of New York, 1990.
[17] M. Gamon and A. Aue, " Automatic Identification of Sentiment Vocabulary: Exploiting Low Association with Known Sentiment Terms", In Proceedings of the Workshop on Feature Engineering for Machine Learning in Natural Language Processing, pp. 57-64, Ann Arbor, 2005.
[18] K. Dave, S. Lawrence and D. M. Pennock, "Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews", In Proceedings of 12th International Conference on World Wide Web, pp. 519-528, Hungary, 2003.
[19] B. Liu, "Sentiment Analysis: A Multi- Faceted Problem", In Proceeding of IEEE Intelligent Systems, pp. 76-80, 2010.
[20] E. Cambria, A. Hussain and C. Eckl, " Taking Refuge in Your Personal Sentic Corner", In Proceeding of Workshop on Sentiment Analysis where AI meets Psychology, pp. 35-43, Thailand, 2011.
[21] A. Das and S. Bandyopadhay, "SentiWordNet for Bangla", 2010.
[22] A. Das and S. Bandyopadhay, "Labeling emotion in Bengali blog corpus a fine grained tagging at sentence level", In Proceeding of 8th Workshop on Asian Language Resources, pp. 47-55, Beijing, 2010.
[23] G. Draya, M. Planti, A. Harb, P. Poncelet, M. Roche and F. Trousset, " Opinion Mining from Blogs", In Proceeding of International Journel of Computer Information Systems and Industrial Management Applications, 2009
[24] S. M. Kim and E. Hovy, "Determining the sentiment of opinions", In Proceeding of COLING, pp. 1367-1373, 2004.
[25] D. Narayan, D. Chakrabarti, P. Pande and P. Bhattacharyya, " An experience in building the indo wordnet -a wordnet for hindi", In Proceeding of First International Conference on Global WordNet, 2002.
[26] D. Rao and D. Ravichandan, " Semi-supervised polarity lexicon induction", In Proceeding of 12 conference of the European Chapter of the Association for Computational Linguistics, pp. 675-682, USA, 2009.
[27] A. Das, "Opinion Extraction and Summarization from Text Documents in Bengali", Doctoral Thesis. UMI Order Number: UMI Order No. GAX95-09398., Jadavpur University, 2011.
[28] A. Joshi, A. R. Balamurali and P. Bhattacharyya, "A Fall-back Strategy for Sentiment Analysis in Hindi: a Case Study", In Proceedings of the 8th ICON, 2010.
[29] Amitava Das, http://www.amitavadas.com/
[30] Janyce M. Wiebe,"Learning Subjective Adjectives from Corpora",http://www.cs.columbia.edu/~vh/courses/LexicalSemantics/Orientation/wiebe-aaai2000.pdf
[31] Brill Tagger, http://www.ling.gu.se/~lager/mogul/brill-tagger/index.html.

Reviewing Soft Computing Approaches for Edge Detection: Hybrid and Non-hybrid

Manisha Kaushal
UIET, Panjab University Chandigarh
Er.manishakaushal@gmail.com

Akashdeep
UIET, Panjab University Chandigarh
akashdeep@pu.ac.in

***Abstract*—Soft Computing is a multifaceted technique comprising of Fuzzy Logic, Neural Network, Genetic algorithms and other Evolutionary computation. These paradigms have found wide variety of applications in the field of image processing. One of the most vital applications of image segmentation is edge detection where edge refers to the boundary between two consistent regions and edge detection is the process of detecting and finding abrupt discontinuities in an image. This paper summarizes hybrid and non-hybrid approaches for edge detection. The objective of this paper is to survey the core issues for soft computing based approaches for edge detection.**

***Index Terms*—Soft computing, Neural Networks, Genetic Algorithm, Fuzzy logic, Hybrid**

I. Introduction

A classical problem in the field of computer vision and image processing is edge detection. This paper covers various edge detection techniques based on soft computing approaches such as Neural network, Fuzzy logic and Genetic Algorithm. Some of the hybrid approaches are also discussed. At first, Artificial Neural network based edge detection approach are presented which are inspired by the working of nervous system. Neural networks have the ability to learn about solution from input data. The Second approach, fuzzy logic based edge detection takes into account vides variety of uncertainties in logical reasoning. The third approach, Genetic algorithm which is basically derived from evolution theory. GA can be applied to various areas related to image processing. Hybrid approaches using combination of two or more of the above mentioned techniques are presented next which is followed by the conclusion and future directions.

II Soft Computing Approaches

Soft computing refers to a bank of techniques that are stretching many fields that come under various categories in Computational Intelligence. They are implanted on field within computer science to surmount problems such as NP-complete problems for which there is no known algorithm that can compute an exact solution in polynomial time. Soft computing varys from conventional (hard) computing based on the concept of precise modeling and analyzing to yield accurate results, human mind is the role model of soft computing [23]. Image segmentation is implemented using following approaches of soft computing (1) Neural Network based (2) Fuzzy Logic based and (3) Genetic Algorithm based.

A. Neural Network Based Approaches

Neural network is the emerging technology which can be used in many applications such as digital image processing. An artificial neuron is the center point in working of neural network. A simple model of neural networks consists of multiple inputs and a single output. Weights are assigned to inputs [17] and different neurons combine these weighted inputs and help the neuron to fire. A neuron fires ie produces a positive input if its combined value is greater than threshold. AN activation function is applied to combined input to calculate its output [16]. Various activation function employed includes binary sigmoid, bipolar etc. The edge detection approaches based on neural network provides better results than the classical edge detection approaches.

Multi-layer feed forward neural network framework with one hidden layer was used by W.E. Blanz et al.[1] for segmentation in gray scale images. Their framework consists of three layers: an input layer, a single hidden layer, and an output layer. The input and output layers are not directly linked and all units of a layer is completely connected with the units in the layer above but there are no links among units within a layer. The Back propagation learning algorithm was applied to convert the image segmentation problem into a pixel classification problem. The input vector for the ANN included vector of features extracted from every pixel, and the output vector was the vector of classes desired for segmentation. They executed image segmentation on two real world applications:

Armando J. Pinho et al. [2] explored another approach to edge detection based on neural network classifiers which uses some properties of the data in order to simplify the design of the disseminate functions. They used feed-forward neural networks, trained with back-

doi:10.4304/jetwi.5.4.372-379

propagation comprising of nine inputs, one output and from two to six hidden units.
They compared their approach with the well known derivate of a Gaussian edge detection filter and found that the neural networks generates less thick and less missing detections compared to the DG Linear filters.

Yasar Becerikli et al. [3] demonstrated that edge detection can be implemented using artificial neural network (ANN) by taking any raw image for edge detection using Laplacian method to produce the edges of the image and neural network in turn applied to learn edges of all images. The proposed artificial neural network architecture consists of 22 cells i.e. nine for input layer, twelve for hidden layer and one for output layer Each node uses a sigmoid function, $f(x) = 1/(1+e^{-x})$ as the activation function. In order to train neural network the Back Propagation method with momentum factor was applied and comparison was made with the laplacian edge detectors.

Leila Fallah Araghi et al. [4] put forward two methods for edge detection. The first method used Neural Network and the second method is Sobel methods based on Wavelet function. The proposed method applied the multi layer perceptron neural network with two layers for edge detection and modified levenberg marquart for learning. Each pixel of image is taken as input and edges are output of neural network. The comparison of the proposed approach with classical edge detection methods such as Canny edge detection method found that results of neural based edge detection approach was very promising.

Jesal Vasavada et al.[5] come up with an algorithm based on Feed forward Neural Network (FNN) technique to detect edges in gray scale images and applied the back propagation learning algorithm in order to decrease the error rate. The training patterns applied by them are Standard deviation and gradient values of the image to be processed which are calculated using Sobel operators. The network is tested for a wide variety of grayscale images. The proposed approach is analyzed against the classical operators such as Prewitt, Roberts, Sobel, LoG and other neural network based method in which binary training patterns are applied and on the basis of visual perception and edge pixels counts.

Sabeur Abid[6] have successfully developed multilayer perceptron (MLP) image edge detection method. The author has applied Back propagation algorithm is in the learning stage in order to determine all learning patterns. The proposed algorithm works well for gray-scale images and can be extended to color images.

Nikša Antišić etal. [7] introduced a novel approach to edge detection based on back propagation for noisy images. This approach basically applied feed-forward multilayer back propagation neural network for detecting edges and comparisons are drawn at the end with classical edge detector such as Sobel operator to show the performance of the technique.

B. Fuzzy Logic Based Approaches

The fuzzy technique has always been put forward as one of the modern methods in variety of processes as an operator to simulate at a mathematical level the compensatory behavior in process of decision making or subjective analysis. Fuzzy image processing is basically assembling of all techniques that understand, represent and process the images, their segments and features as fuzzy sets. Fuzzy image processing has three major steps during processing of an image: image fuzzification, changing of membership values, and, if required, image defuzzification. Images are migrated from intensity level plane to fuzzy level plane known as fuzzification to modify their membership values. A rule based, clustering or fuzzy integration approach might be employed for this purpose. 15]. There are wide varieties of methods for illustrating the advancement in the field of fuzzy logic based edge detections techniques. .

One of the most promising methods by Noor Elaiza et al. [8] is based on the Gaussian shaped membership function to the Rule Based Fuzzy (RBF) image detection. The introduced algorithm improves the detection of periosteal and endosteal edges of hand phantom radiographs. The Mean and median filters are used for preprocessing tools. Both subjectively and statistically they concluded that GRBF i.e Gaussian shaped Rule Based Fuzzy produces better results than the RBF i.e. Rule Based Fuzzy(RBF) image detection.

Sulimani et al. [9] put forward a approach based on the use of a fuzzy classifiers for detecting edges in grayscale images. The primary distinction between proposed method and other identical method is the morphological operation which is applied in order to obtain the accurate i.e. thin edges in images. They concluded that the applied approach has brought out far better results as compared to the other methods. One enhancement of the proposed scheme was obtainment of continuous edges and use of Chord-to-Point Accumulation Technique to perform corner extraction.

Methos of Aborisade et al.[10] classifies edges into three categories based on their strength which were later on fuzzified using Gaussian membership functions. Membership was changed to low, medium or high using Fuzzy if-then rules. Defuzzification used to calculate final output was Mamdani Inference. The efficiency of the introduced method is demonstrated through computer simulation against

TABLE 1 :
SUMMARIZATION OF NEURAL BASED APPROACHES

Author	Network Structure	Training & learning	Types of images	Comparison Drawn	Analysis of given results	Findings
W.E. Blanz et al. 1]	multi-layer Feed forward perceptron	Standard back-propagation (BP) learning algorithm	Gray scale images	---	Better for complex real world applications.	Justification of result is required.

Armando J. Pinho et al. [2]	Feed-forward neural networks,	Back-propagation (BP) learning algorithm	Gray scale images	Gaussian edge detection filter	Less thick and less missing detections compared to the DG Linear filters.	Under adverse conditions the performance degrades gracefully.
Yasar Becerikli et al. [3]	MLP structure, using forward propagation neural network	Supervised learning method with momentum	Gray scale images With Noise	Laplacian based edge detection method.	Provides more desirable results than Laplacian method.	Works only with noise less images
Leila Fallah Araghi et al. [4]	Multi layer perceptron neural network	Modified levenberg marquart used for learning	Gray scale images	Classical methods such as canny and sobel method for edge detection	The proposed approach provides better results than classical methods.	Only Lena image taken for results.
Jesal Vasavada et al. 5]	Feedforward Neural Network (FNN)	Back propagation learning algorithm	Gray scale images	Classical methods such as canny, sobel, prewitt, method for edge detection	The method detects highest edge pixels and performs well in case of noisy images also.	Less Training rules are used.
Sabeur Abid[6]	Multi layer perceptron neural network	Back propagation algorithm	Gray scale images	Classical methods such as canny, sobel, Robert, prewitt, method for edge detection	More detailed edges are found than classical operators.	Can be extended to color images
Nikša Antišić etal. [7]	feed-forward multilayer Neural Network	Back propagation algorithm	Gray scale images With Noise	Compared with the Sobel operator only	Provides more promising results than classical edge detector.	Can be compared with other edge detection operater.

existing classical edge detector such as Sobel and Krisch edge detection operators.

Aijaz Ur Rahman khan et al. [11] explored fuzzy rule base algorithm which provides a method for detecting edges efficiently from the gray scale images. In the proposed approach, Mandani method is employed for defuzzification and Triangular Membership functions is used as membership function. At last the method is compared with the classical edge detector such as prewitt and sobel. In future proposed method can be applied on higher dimension window.

Wafa barkhoda et al. [12] introduced two different methods based on calculation of gradient and standard deviation of pixels to form two set of edges used as inputs for fuzzy system. The fuzzy system includes appropriately defined fuzzy rules and fuzzy membership functions to decide about pixel classification as edge or non-edge. The proposed method demonstrated that the extracted edges when analyzed against classical edge detection methods such as Sobel, Robert, and Prewitt provides better results.

Koushik Mondal etal.[13] formulate a new edge detection technique that is based on fuzzy rules. The main focus of their study is to imply fuzzy set theory for image segmentation technique and quality of the present approach is measured both subjectively and objectively. The proposed approach is capable to produce promising results as compared to their counterparts addressed in literature. RanitaBiswas et al. [14] put forward a novel approach to edge detection that selects effective threshold values using type-2 fuzzy logic to be applied in Canny's edge detector. Experimental
results suggest that the proposed approach is capable one in comparision to the Canny's edge detector.

C. Genetic Algorithm Based Approaches

Genetic algorithm can be used to optimize the functioning of the classical edge detection algorithms. They are the simulation of natural biological progression mechanisms which are used to develop highly adaptive search algorithm Basically, a genetic algorithm are adaptive heuristic search algorithms which consists of three major operations: selection, crossover, and mutation. The selection used to improve average quality of the population by keeping only the fittest ones in the population [18]. The quality of the individual is measured by the fitness function. The crossover operator is a genetic operator that combines two chromosomes to produce a new chromosome. They are classified as One Point, Two Point, uniform, arithmetic and heuristic crossover oprators. The mutation operator changes one or more gene values in a chromosome from its initial state. It helps in maintaining diversity of the population[19].

Mutation is done with small probability and helps to avoid local minima/maxima.

Zhang Jin-Yu et al. [20] put forward an approach to edge detection that automatically determines an optimal threshold. They proposed automatic threshold algorithm for images processing based on

TABLE2:
SUMMARIZATION OF FUZZY BASED APPROACHES

Author	Approach	Types of images	Comparison Drawn	Analysis of given Results	Findings
Noor Elaiza et al. [8]	Gaussian shaped membership function to the Rule Based Fuzzy(RBF) image detection	Hand Phantom Radiograph Images	Rule Based Fuzzy (RBF) image detection.	GRBF provides better results than RBF	The proposed algorithm is far more computationally expensive.
SULIMAN1, et al. [9]	Fuzzy classifiers	Grayscale images	----	Provides thick edges.	The Chord-to-Point Accumulation Technique can be used along with this algorithm to perform corner extraction.
Aborisade, D.O[10],	Gaussian membership functions	Grayscale images	Sobel and Krisch edge detection operator	Better than the classical edge detectors	Better refining algorithm using different membership functions can be developed.
Aijaz Ur Rahman khan and Dr. Kavita Thakur[11]	Fuzzy rule base algorithm	Grayscale images	Sobel and Prewitt edge detection operator	Better than the classical edge detectors	The proposed algorithm avoids detection of spurious edges corresponding to nois.
Wafa barkhoda et al. [12]	Fuzzy Edge Detection Based on Pixel's Gradient and Standard Deviation Values	Grayscale images	classical edge detection methods such as Sobel, Robert, and Prewitt	Better than the classical edge detectors	Compared only with traditional operators.
Koushik Mondal etal.[13]	Mamdani rule base.	Gray scale image with noise	Compared with different threshold methods.	efficient as compared to the other techniques	Only lena image used for drawing comparison
Ranita Biswas etal. [14]	single threshold selection technique from image histogram using type-2 fuzzy logic	Medical images of hand radiography	Compared with the canny edge detector.	minimizes uncertainty involved in thresholding procedure.	Can be extended to colored image domain.

genetic algorithms and improved Sobel operator. It is observed that although the proposed algorithm overcame many shortcomings of classical Sobel edge detection algorithm but detected edges are not fine enough and better algorithm can be developed in future. The advantage of using Genetic algorithm is that it is capable of global search on the data space and the biggest disadvantage of using it is that it cannot use local information effectively.

Zhang Jing et al. [21] introduced evolutionary approach for edge detection which combines the local search capability of the classical edge detection operators with global space capability of the emerging genetic based edge detection approach. The approach can be applied to extract the edges efficiently in wield images. The study concluded that proposed algorithm cannot restrain the noise but can protect edge information effectively.

Wenlong Fu et al. [22] approach takes as input an entire image and pixels are classified as edges or non classical edge detector operator such as Laplacian and Sobel edge detectors and the results suggest that the detectors evolved by GP provides better results than the classical edge detectors.

Wenlong Fu et al.[23] presented a rising genetic programming approach in order to develop detectors with new fitness functions. Fitness function was trained with accuracy of training images. The experimental approach points that fitness functions was able to balance the accuracies across results of detection and joining accuracy of overall pixels along with the accuracy of training images. Results show that proposed method outperforms the previously available classical edge detectors. In future, this approach can be applied using a multi-objective approach for the training images.

Huili Zhao et al. [24] proposed algorithm for the enhancement of Canny operator in pavement image detection. The method works on the principle of Mallat wavelet transform to detect the weak edges and quadratic optimization of genetic algorithm for proper thresholding. But in order to make the edge much more accurate, they are still some problems such as payment crack detection, etc to be improved.

III HYBRID SOFT COMPUTING APPROACHES

Hybrid soft computing approaches are the ones who combine features of multiple approaches to overcome each other shortcomings and look for optimized solutions.

The studies discussed in this section are based on combination of neuro-fuzzy and fuzzy- genetic techniques

TABLE3:
SUMMARIZATION OF GENETIC ALGORITHM BASED APPROACHES

Author	Approach	Operators	Comparison Drawn	Analysis of given Results	Findings
Zhang Jin-Yu et al. [20]	Improved Sobel Operator and Genetic Algorithms	Population Initialization : Produce N individuals randomly with equal probability between 0 to 255. Fitness Function: is the summation of ratio of original gradient and individual's gradient.	Classical operator such as sobel	Better than the classical edge detectors	No crossover and mutation operator is specified and very few images taken.
Zhang Jing et al. [21]	Gradient calculation based on genetic algorithm	Population initialization: Randomly generates 40 chromosomes as the initial population. Fitness function Refers to the maximum classes variance. Crossover operator: Use single-point crossover method with 0.9 probability	----	Edge detection accuracy and noise immunity.	If the mutation probability isachieved much, genetic algorithm will degrade as random search.
Wenlong Fu et al. [22]	Genetic Algorithm and Canny Edge Detector	Mutation operator: inverse mutation operator is used. Fitness function: inverse of Hamilton ring length Crossover operator: Partially mapped crossover operator.	Classical edge detector operator such as Laplacian and Sobel edge detectors	Better than the classical edge detectors	proposed algorithm is very complex.
Wenlong Fu et al.[23]	Genetic Programming via Balancing Individual Training Images	Fitness function: the mean square error or the accuracies for different indicators are employed. Population size:500 Mutation Probability: 0.15 Crossover Probability: 0.80	Compared with various fitness functions	Better results	Can be applied using a multi-objective approach for the training images in future.
Huili Zhao et al. [24]	Mallat wavelet transform	Quadric optimization genetics algorithm is applied	----		Fails at detecting thin edges
A. Saenthon et al.[25]	Edge detection based on the soldier detection	crossover, mutation, and reproduction operators are applied	Classical edge detector operator such as Sobel, Prewitt, and Canny filters.	The accuracy of proposed technique is better than Sobel, Prewitt, and Canny filters.	Computationally Complex Technique

TABLE4:
SUMMARIZATION OF FUZZY- NEURAL BASED APPROACHES

Author	Network Structure	Training & learning	Types of images	Analysis of given results	Findings
M. EminYuksel [26]	Each NF subdetector is a first-order Sugeno type fuzzy inference system with 3-inputs and 1-output.	Training by using simple artificial training images	Gray scale images with impulse noise	Promising results as compared to ones classical operator.	Thick edges
Mohammed Madiafi et al. [27]	Kohonen neural network (ANN)	Unsupervised fuzzy competitive learning	Gray scale images	Better results as compared to traditional ones.	Need to compare with existing approaches
Victor Boskovitz et al. [28]	Multi-layer Feed-forward neural networks and Fuzzy entropy for detecting edge pixels	Unsupervised Learning	Gray scale noisy images	Better than other approach such as histogram clustering : median cut, FCM, etc.	Less Training rules used
H. Farahanirad et al. [29]	neuro-fuzzy network (nf), an adaptive median filter and four identical neural networks (nn).	Back propagation learning algorithm	Gray scale images with salt and pepper Noise	Provides more desired results than the classical edge detector such as canny, sobel and performs well in case of noisy image	Limited test images including Boats and Cameraman
Siwei Lu et al. [30]	three-layer feed-forward fuzzy neural network	Back propagation learning algorithm	Gray scale images	The method detects the highest edge pixels and performs well in case of noisy images	Complex algorithm

A .Neuro-Fuzzy Algorithms

The approach of artificial intelligence founded on fuzzy logic and neural networks are put together to amalgamate these two soft computing techniques in order to reduce the limitations and difficulties of each isolated technique. Usually, when they are applied in a combined way, they are known as Neuro-Fuzzy Systems.

Fuzzy systems are suitable for industrial application but the problem of determining the membership function and appropriate rules is generally exhausting process of attempt and error. This provides to the direction of exercising learning algorithms to the Fuzzy System. The neural network presents computational characteristics of learning algorithms that act as substitute to support the advancement of fuzzy systems. This type of arrangement is characterised by a fuzzy system where fuzzy sets and fuzzy rules area arranged using input output patterns. Therefore, the limitations of the fuzzy systems are removed by the capacities of the neural networks.They combine techniques of both areas so these approaches are complementary, which justifies its use together.

M. EminYuksel [26] put forward a recent neural fuzzy operator that was employed for extraction of edges in impulse noise images. A number of NF sub detectors were combined with a post processor to develop the implemented operator. The comparison of the proposed approach with classical edge detection methods such as Sobel and Canny edge detection method found that results of fuzzy neural based edge detection approach was very promising.

M. Madiafi et al. [27] has presented a neuro fuzzy approach for automatic recognition of facial images. This approach is build on a Kohonen neural network (ANN), which was trained in unsupervised way, using a fuzzy competitive learning algorithm previously developed, applied and well tried on real images. The experimental approach points the effectiveness of the techniques by demonstrating through examples using a test dataset used by other researchers.

V. Boskovitz et al. [28] has proposed architecture comprising of a multilayer perceptron (MLP) like network that carry out image segmentation by adaptive thresholding of the input image using labels. A fuzzy clustering technique was used to automatically pre-select these labels. A feed-forward neural network with unsupervised learning is used whose output is described as a fuzzy set. Segmentation errors and likely edge pixels were measured by fuzzy entropy. The results presented indicate that study is good when tested on noisy images.

H. Farahanirad et al. [29] come up with a neuro-fuzzy edge detection algorithm in situations where the image is corrupted by Salt and Pepper noise. The presented approach is very simple and combination of a neuro-fuzzy network (nf), an adaptive median filter and four identical neural networks (nn). The improvement of the proposed work is that it provides better results than the several edge detection methods such as Roberts,Prewitt, Sobel, Zero-crossing, Canny, etc under the noisy conditions.

Fuzzy-neuro system presented by Siwei Lu et al.[30] was employed for both edge detection and enhancement. A three-layer feed-forward fuzzy neural network was used for edge detection and a modified Hopfield network was used for enhancement. Sample patterns were first fuzzified and used to train proposed fuzzy neural network and trained network was able to determine the edge elements. The authors had also compared their approach with standard edge detection operators.

B. Fuzzy-Genetic Edge Detection Algorithm

Fuzzy Logic has turned out to be a very efficient tool for describing human knowledge with the help of mathematical expressions. The advantage of fuzzy system is that they are rule based systems which can easily represent the imprecise information. So, prototyping can be made very effortlessly but most of the time involved in changing the involved parameters and that can turn out to be worse with growing difficulty of the system. Thus in order to optimize the fuzzy rule set based on the training data, genetic algorithms (GA) is proposed which is also known as fuzzy-genetic algorithm. The fuzzy-genetic algorithm belongs to a newly developed class of hybrid knowledge which combines the main features of the fuzzy and genetic paradigms and these approaches are very promising for noisy images. FGA works considerably better than other well-known conventional edge detection methods in the literature.

A. JUBAI et.al [31] put forward a novel algorithm which integrates uncertainty principles of fuzzy with evolutionary concepts of genetic algorithms for detection of oil spilled on sea. Authors have tried to improve Pal-king fuzzy edge detection method with the help of genetic algorithms. The method has a problem as the calculations were complex and thresholds value was fixed which was not useful for problems like oil in sea. The proposed algorithm presents the hybrid approach which consists of improved version of fuzzy enhancement algorithm to simply the Pal-king fuzzy edge detection method and a genetic algorithm with which threshold value can be easily determined for image processing. The presented approach have been put up in a complemtary fashion to adjust some of the undesirable features.The processing results are compared with Pal-King algorithm for the purpose of experimental evaluation.

Somya Jain [32] presented a novel approach for the edge detection which combines fuzzy logic derivatives and evolutionary learning techniques such as genetic algorithm. The combinations of both the approaches will lead to simplify the edge detection process. In the introduced approach, rule set is defined by fuzzy logic which is further optimized by using genetic algorithm. The simulation and implementation process of the proposed approach is carried by comparing it in terms of kappa value and entropy measurement with the other standard edge detection operators in literature

Janne Koljonen et.al [33] proposed a theory of optimizing fuzzy pattern matching using genetic algorithm. The authors have taken the advantages of Genetic algorithms in the field of pattern matching. The developed algorithm firstly perform fuzzy segmentation

and further membership function and template are optimized by stochastic search strategy of genetic algorithm The accuracy of the algorithm is measured by comparing them with traditional pattern matching algorithms in literature in terms of Pearson correlation coefficient.

M.Abdulghafour [34] has developed image segmentation approach that utilizes fuzzy logic and genetic algorithms. The author has taken the help of genetic algorithms at every step of image segmentation in order to develop the membership functions that are required to categorize the presence of image features through Fuzzy logic. The effectiveness of the proposed approach for producing successful results was demonstrated. The proposed approach makes novel improvement when it is compared with its ideal counterparts.

IV. Analysis and Inferences

In previous sections work related to the edge detection with application of soft computing techniques is presented. In this section, inferences from review of previous sections are presented. Most results have been taken on grey scale images where as only few studies are available on application of soft computing techniques that directly work on true colored images. Since substantial loss of information occurs when we transform image to grey scale therefore studies could focus in this direction. Comparisons shall be drawn in context to present trends where as almost every paper has considered classical operators for sake of comparisons. Work on reducing the computational time of soft computing approaches may be another area for exploration. Hybrid techniques such as Neuro-fuzzy, Fuzzy –GA had been utilized recently but studies on neural and GA are still limited. Recent Approaches in Soft computing like Ant Colony Optimization and its various variants like Max-Min ant system, Rank based Ant-system, continuous orthogonal ant colony and ant colony with fuzzy logic may also be explored. There are few papers in this direction. Edge detection approaches as an application in medical field for detection of diseases and satellite images can be an effective tool.

V. Conclusion

The paper has tried to expend encroachment of soft computing techniques to edge detection accessible in literature. Both hybrid and non-hybrid approaches discussed had sound foundation in field of Edge detection as noteworthy literature is available. The paper has also listed down areas where improvements and explorations can be made by researchers in this unending field. This can be accomplished that even vast amount of literature available is not any hindrance to scope of improvements that can be made in this direction.

References

[1] W.E. Blanz, S.L. Gish, “A connectionist classifier architecture applied to image segmentation”,*Proceedings of 10th International Conference of Pattern Recognition*, pp.272-277, 1998.

[2] A. J. Pinho_ and Lu´ıs B. Almeida, “Neural network classifiers for edge detection” *Published in Pattern Recognition and Image Analysis*, *Proceedings of NSPRIA’99*, Barcelona, Spain, April 1999, pp. 371–376 (Vol. I)

[3] Y. Becerikli and H. E. Demiray,“Alternative Neural Network Based Edge Detection”, *Neural Information Processing- Letters and Reviews* Vol. 10, Nos. 8-9, Aug.-Sept. 2006.

[4] L. F. Araghi, M.R. Arvan, “An Implementation Image Edge and Feature Detection Using Neural Network”, *Proceedings of the International Multi-Conference of Engineers and Computer Scientists* 2009 Vol I IMECS 2009, March 18 - 20, 2009, Hong Kong.

[5] J. Vasavada and S. Tiwari, “An Edge Detection Method for Grayscale Images based on BP Feedforward Neural Network”, *International Journal of Computer Applications* (0975 – 8887) Volume 67– No.2, April 2013.

[6] S. Abid and F. Fnaiech,” A novel neural network approach for image edge detection”, *Electrical Engineering and Software Applications (ICEESA)*, 2013 International Conference.

[7] N. Antišić, M. Bonković and B.Džaja, “Neural network as edge detector”, *Software, Telecommunications and Computer Networks* (SoftCOM), 2012 20th International Conference

[8] N. Elaiza, A. Khalid, M. Manaf , M. E. Aziz, N. M. Noor and N. Zainol, “Gaussian Rule Based Fuzzy (GRBF) Membership Edge Detection on Hand Phantom Radiograph Images”, *Fifth International Conference on Computer Graphics, Imaging and Visualization* 978-0-7695-3359-9/08 2008 IEEE, DOI 10.1109/CGIV.2008.70.

[9] C.Sulimani, C. Boldisori, R Bazavan and F. Moldoveanu, “A Fuzzy Logic based Method for Edge Detection”, Bulletin of the Transilvania University of Braşov, Series I: Engineering Sciences, Vol. 4 (53) No. 1 – 2011.

[10] Aborisade and L. Akintola,”Novel Fuzzy logic Based Edge Detection Technique”, *International Journal of Advanced Science and Technology,* Vol. 29, April, 2011.

[11] A. U. R. Khan and Dr. K. Thakur, “An Efficient Fuzzy Logic Based Edge Detection Algorithm for Gray Scale Image”, *International Journal of Emerging Technology and Advanced Engineering,* (ISSN 2250-2459, Volume 2, Issue 8, August 2012).

[12] W. Barkhoda, F. A. Tab, O.K. Shahryari, “Fuzzy Edge Detection Based on Pixel’s Gradient and Standard Deviation Values”, *Proceedings of the International Multiconference on Computer Science and Information Technology*, pp. 7 – 10, ISSN 1896-7094.

[13] K. Mondal, P. Dutta and S. Bhattacharyya, “Fuzzy Logic Based Gray Image Extraction and Segmentation” *International Journal of Scientific & Engineering Research*, Volume 3, Issue 4, April-2012 1 ISSN 2229-5518.

[14] R. Biswas and J.Sil, “An Improved Canny Edge Detection Algorithm Based on Type-2 Fuzzy Sets”, *2nd International Conference on Computer, Communication, Control and Information Technology(C3IT-2012)* on February 25 - 26, 2012.

[15] A. A. Alshennawy, and A.A. Aly, “Edge Detection in Digital Images Using Fuzzy Logic Technique”, *World Academy of Science, Engineering and Technology, 2009.*

[16] M. K. Sharma and V. Shrivastav, “Feature Extraction from web data using Artificial Neural Networks (ANN)”, *International Journal of Computer Trends and Technology*- volume3 Issue5- 2012

[17] S.Haykin, “Neural Networks and Learning Machines”, *4th Edition, 2010, Prentice Hall of India*. ISBN 9788131740156

[18] A. Borji, and M. Hamidi, ”Evolving a Fuzzy Rule-Base for Image Segmentation”, *International Journal of Intelligent Systems and Technologies*, 2007, vol2 issue 7pp.471-476.

[19] M. Abdulghafour, ”Image segmentation using Fuzzy logic and genetic algorithms”, *Journal of WSCG*, vol. 11,no.1, 2003. ISSN 1213-6972

[20] Z. J.Yu1, Chen Yan and Huang Xian-Xiang, “Edge Detection of Images Based on Improved Sobel Operator and Genetic Algorithms”,
http://ieeexplore.ieee.org/stamp/stamp.jsp? number=0505 4605

[21] Z. Jing and J. Yan-xia, “Genetic Algorithm for Weld Image Edge Detection”, *Proceeding of International Conference on Computer Application and System Modeling (ICCASM 2010)*, pp 512-515.

[22] W. Fu, M. Johnston and M. Zhang, ”Genetic Programming for Edge Detection: A Global Approach”,http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=05949626

[23] W. Fu; Johnston, M.; M. Zhang, "Genetic programming for edge detection via balancing individual training images," *IEEE Congress on Evolutionary Computation (CEC)*, 2012, vol.1, no.8, pp. 10-15 June 2012

[24] H. Zhao, G. Qin and X. Wang, “Improvement of Canny Algorithm Based on Pavement Edge Detection*”, 3rd International Congress on Image and Signal Processing* 2010 pp 964-967

[25] A. Saenthon & S. Kaitwanidvilai, “Development of new edge-detection filter based on genetic algorithm: an application to a soldering joint inspection”, *International Journal of Adancement Manuf Technol* ogy(2010) 46:1009–1019DOI 10.1007/s00170-009-2157-x.

[[26] M.EminYukse,l “Edge detection in noisy images by neuro-fuzzy processing”, *Int. J. Electron. Commun. (AEU)* 61 (2007) 82 – 89.

[27] M. Madiafi and A. Bouroumi, “A Neuro-Fuzzy Approach forAutomatic Face Recognition”, *Applied Mathematical Sciences*, Vol. 6, 2012, no. 40, 1991 – 1996.

[28] V. Boskovitz and H. Guterman, “An Adaptive Neuro-Fuzzy System for Automatic Image Segmentation and Edge Detection*”, IEEE TRANSACTIONS ON FUZZY SYSTEMS*, VOL. 10, NO. 2, APRIL 2002

[29] H. Farahanirad, J. Shanbehzadeh, M. M. Pedram, A.Sarrafzadeh, “ Fuzzy Neural Networks for Edge Detection”, *IEEE Canadian Conference on Electrical and Computer Engineering* Vol. 2, pp. 446 – 449,1917

[30] S. Lua, Z. Wanga and J. Shenbe, “Neuro-fuzzy synergism to the intelligent system for edge detection and enhancement”,0031-3203/03/$30.00 ? 2003 Pattern Recognition Society. Published by Elsevier Ltd.

[31] A. JUBAI, B. JING and J. YANG, “Combining fuzzy theory and a genetic algorithm for satellite image edge Detection, International Journal of Remote Sensing ISSN 0143-1161 print/ISSN 1366-5901 online # 2006 Taylor & Francis ISSN 0143-1161 print/ISSN 1366-5901 online # 2006 Taylor & Francis.

[32] S. Jain, “Edge Detection Using Fuzzy Derivative and Evolutionary Algorithms”, *IUP Journal of Information Technology*, Vol. IX, No. 2, June 2013, pp. 7-35

[33] J. Koljonen and J. T. Alander, “Genetic algorithm for optimizing fuzzy image pattern matching”.

[34] M.Abdulghafour, “Image Segmentation using fuzzy logic and genetic algorithms”, *Journal of WSCG*, Vol.11, No.1 ISSN 1213-672, WSCG2003, February 3-7, 2003, Plzen, Czech Republic.

Size of Training Set Vis-à-vis Recognition Accuracy of Handwritten Character Recognition System

Munish Kumar
Computer Science Department
Panjab University Constituent College
Muktsar, Punjab, India

R. K. Sharma
School of Mathematics and Computer Applications
Thapar University
Patiala, Punjab, India

M. K. Jindal
Department of Computer Science and Applications
Panjab University Regional Centre
Muktsar, Punjab, India

***Abstract*—Support Vector Machines (SVMs) have successfully been used for character recognition. In the present study, we have shown how the recognition accuracy of a SVM classifier varies with variation in the training set size. The training set for this work is taken from samples of offline handwritten Gurmukhi characters. For recognition of a handwritten Gurmukhi character, we have used curvature features extracted from the skeletonized image of each Gurmukhi character. Features of a character have been computed based on statistical measures of distribution of points on the bitmap image of character. To extract these features, the image of each Gurmukhi character is first segmented into few zones and then the curvature shape is computed within each of these zones. Considering all the zones, a feature set is formed for representation of each image pattern and a database of 3500 isolated handwritten Gurmukhi characters has been used for the same. The results of investigation presented in this paper show that the size of training set has a significant effect on the accuracy of offline handwritten Gurmukhi script recognition system.**

***Index Terms*—Feature extraction; curve fitting; handwritten character recognition; SVM.**

I Introduction

Handwritten Character Recognition, usually abbreviated as HCR, is the process of converting handwritten text into machine processable format. HCR is the field of research in pattern recognition and artificial intelligence. Handwriting recognition provides a methodology for improving the interface between user and computer as it enables computers to read and process handwritten documents which are currently being processed manually. A good number of researchers have already worked on the recognition problem of offline printed characters. For example, a printed Gurmukhi script recognition system has been proposed by Lehal and Singh [3]. Wen et al. [4] have proposed handwritten Bangla numerals recognition system for automatic letter sorting machine. Swethalakshmi et al. [5] have proposed handwritten Devanagri and Telugu character recognition system using SVM. The input to their recognition system consists of features of the stroke information in each character and SVM based stroke information module has been considered for generalization capability. Pal et al. [6] have presented a technique for off-line Bangla handwritten compound characters recognition. They have used modified quadratic discriminant function for feature extraction. They have also used curvature features for recognizing Oriya characters. Chaudhary and Pal [7] have proposed recognition system for two Indian scripts, Bangla and Devanagari. They have used tree classifier for character recognition. Hanmandlu et al. [8] have reported grid based features for handwritten Hindi numerals recognition. They have divided the input image into 24 zones. After that, they compute the vector distance for each pixel position in the grid from the bottom left corner and normalize these distances to [0, 1] in order to obtain the features. Bansal and Sinha [9] have provided a complete OCR system for printed Devanagari script. Kumar [10] has proposed a technique for recognition of handwritten Devanagri characters. He has used an AI approach to integrate information from sources and a fuzzy logic concept to handle uncertainties and imprecise information. In order to tackle the problem related to selection of a proper dataset for training a SVM, different strategies have been considered in this work. Chaudhury et al. [11] has been presented a scheme using a syntactic method for connected Bangla handwritten numerals recognition. In 2006, Roy and Pal have presented an automatic scheme, for word-wise identification of hand-written Roman and Oriya scripts for Indian postal automation [12]. In 2008, Sharma et al. [13] have proposed a system based on elastic matching for online Gurmukhi script recognition. Here, we have analysed the recognition performance of the SVM with variations in the training set size. We have used parabola curve and power curve based features for representation of handwritten Gurmukhi characters in feature space. In doing so, a skeletonized image of handwritten Gurmukhi character is segmented into the zones of equal size and the shape of curve in each zone is determined. This shape defines the features of the zone. This paper is organized into six sections. Introduction to Gurmukhi script and data collection for this work is described in Section 2.

doi:10.4304/jetwi.5.4.380-384

Section 3 presents a method of feature extraction for handwritten character recognition system. Classification is described in Section 4. Section 5 shows some experimental results to prove the usefulness of this approach. Conclusions are finally included in Section 6.

II Gurmukhi Script and Data Collection

Gurmukhi script is the script used for writing Punjabi language. The word Gurmukhi has been derived from the Punjabi term "Guramukhi", which means "from the mouth of the Guru". Gurmukhi script is the 12th most widely used script in the world. Gurmukhi script has three vowel bearers, thirty two consonants, six additional consonants, nine vowel modifiers, three auxiliary signs and three half characters. The character set of Gurmukhi script is given in Figure 1. For the present work, we have collected data from 100 different writers. These writers were requested to write each Gurmukhi character. All these characters are scanned at 300 dpi resolution with HP-1400 scanner A sample of handwritten characters by 5 different writers (W1, W2, …, W5) is given in Figure 2.

The Consonants

ਸ ਹ ਕ ਖ ਗ ਘ ਙ ਚ ਛ ਜ ਝ ਞ ਟ ਠ ਡ ਢ ਣ ਤ ਥ

ਦ ਧ ਨ ਪ ਫ ਬ ਭ ਮ ਯ ਰ ਲ ਵ ੜ

The Vowel Bearers

ਉ ਅ ਏ

The Additional; Consonants (Multi Component Characters)

ਸ਼ ਜ਼ ਖ਼ ਫ਼ ਗ਼ ਲ਼

The Vowel Modifiers

ੳ ੰ ੇ ੈ ਿ ੀ ਾ ੁ ੂ

Auxiliary Signs

ੱ ੰ ਂ

The Half Characters

ੵ ੍ਰ ੍ਵ

Figure 1. Gurmukhi script character set.

Script Character	W1	W2	W3	W4	W5
ੳ					
ਅ					
ੲ					
ਸ					
ਹ					

Figure 2. Samples of handwritten Gurmukhi characters.

III Handwritten Gurmukhi Script Recognition System

The proposed recognition system consists of the phases, namely, digitization, preprocessing, feature extraction and classification. The block diagram of proposed recognition system is given in Figure 3.

3.1 Digitization

Digitization is the process of converting the paper based handwritten document into electronic form. The electronic conversion is accomplished using a process whereby a document is scanned and an electronic representation of the original document, in the form of a bitmap image, is produced. We have used HP-1400 scanner for digitization Digitization produces the digital image, which is fed to the pre-processing phase.

3.2 Preprocessing

Preprocessing is a series of operations performed on the digital image. Preprocessing is the initial stage of character recognition. In this phase, the character image is normalized into a window of size 100×100. After normalization, we produce bitmap image of normalized image. Now, the bitmap image is transformed into a contour image.

Feature extraction and classification phases are discussed in next sections.

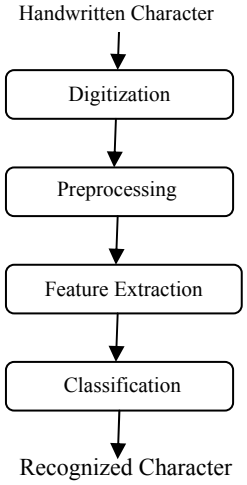

Figure 3. Block diagram of handwritten character recognition system.

IV Feature Extraction

In this phase, the features of input character are extracted. The performance of handwritten character recognition system depends on features, which are being extracted. The extracted features should be able to uniquely classify a character. The performance of recognition system greatly depends on features that are being extracted. We have used two feature extraction techniques, namely; parabola curve fitting and power curve fitting in order to find the feature sets for a given character. The skeletonized image of a handwritten Gurmukhi character is segmented into 100 zones of equal size (10×10). The shape of the curve in each zone is then

estimated by fitting a parabola and also by fitting a power curve using least square estimation. The coefficients of these curves represent the handwritten Gurmukhi character into feature space.

4.1 Parabola Curve Fitting based Feature Extraction

The skeletonized image of a character is divided into *n* (=100) zones as illustrated in Figure 4. A parabola is fitted to the series of *ON* pixels in every zone using least square method. A parabola $y = a + bx + cx^2$ is uniquely defined by three parameters: *a, b* and *c*. As such, this will give 3*n* features for a given character.

The steps that have been used to extract these features are given below.

Step I: Divide the skeletonized image into *n* (=100) number of equal sized zones.

Step II: For each zone, fit a parabola using least square method and calculate the values of *a, b* and *c*.

Step III: Corresponding to the zones that do not have a foreground pixel, take the values of *a, b* and *c* as zero.

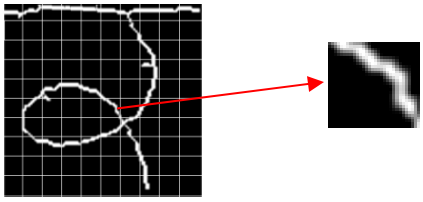

Figure 4. Parabola curve fitting based feature extraction.

4.2 Power Curve Fitting based Feature Extraction

The skeletonized image of a character is again divided into *n* (=100) zones as illustrated in Figure 3. A power curve is fitted to the series of *ON* pixels in every zone using least square method. A power curve of the form $y = ax^b$ is uniquely defined by two parameters: *a* and *b*. This will thus give 2*n* features for a given character.

The steps that have been used to extract these features are given below.

Step I: Divide the skeletonized image into *n* (=100) number of equal sized zones.

Step II: In each zone, fit a power curve using least square method and calculate the values of *a* and *b*.

Step III: Corresponding to the zones that do not have a foreground pixel, take the value of *a* and *b* as zero.

V CLASSIFICATION

In this work, we have used Support Vector Machine (SVM) classifier for recognition. The SVM is a learning machine, which has been widely applied in pattern recognition. SVMs are based on statistical learning theory that uses supervised learning. In supervised learning, a machine is trained instead of programmed to perform a given task on a number of inputs/outputs pairs. SVM classifier has been considered with three different kernels, namely, linear kernel, polynomial kernel and RBF kernel.

VI RESULTS AND DISCUSSION

As stated earlier, we have performed experiments on different training sets sizes while using the SVM as a classifier. The total number of samples in the database is 3500. We have divided the data set using partitioning strategies as depicted in Table 1.

TABLE 1.
PARTITIONING STRATEGIES OF TRAINING AND TESTING DATA

Strategy	*a*	*b*	*c*	*d*	*e*	*f*	*g*	*h*	*i*	*j*	*k*
Training Data	50 %	55 %	60 %	65 %	70 %	75 %	80 %	85 %	90 %	95 %	99 %
Testing Data	50 %	45 %	40 %	35 %	30 %	25 %	20 %	15 %	10 %	5 %	1 %

6.1 Parabola Curve Fitting based Feature Extraction

In this sub-section, we have presented recognition performance results of different training set strategies (*a, b,…, k*) based on the parabola curve fitting based features (Table 2). Using this approach, we have achieved a maximum recognition accuracy of 97.14% when we use strategy *k* and SVM with linear kernel. These results are depicted in Figure 5 graphically.

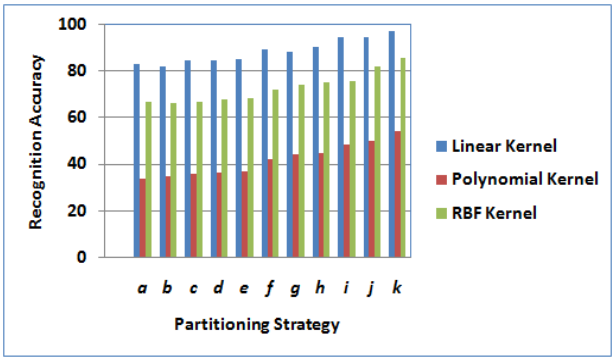

Figure 5. Effect of training set sizes on recognition performance with parabola curve fitting based features.

TABLE 2:
EFFECT OF TRAINING SET SIZES ON RECOGNITION PERFORMANCE WITH PARABOLA CURVE FITTING BASED FEATURES.

Strategy	Linear Kernel	Polynomial Kernel	RBF Kernel
a	82.86%	33.66%	66.97%
b	81.84%	34.79%	66.09%

c	84.43%	36.07%	66.79%
d	84.65%	36.16%	67.59%
e	84.86%	36.95%	68.24%
f	89.02%	42.29%	71.77%
g	88.14%	44.14%	74.20%
h	90.47%	44.57%	74.87%
i	94.57%	48.57%	75.71%
j	94.29%	49.71%	82.14%
k	97.14%	54.29%	85.71%

6.2 Parabola Curve Fitting based Feature Extraction

In this sub-section, recognition results of training set strategies (*a*, *b*,…, *k*) based on power curve fitting features using SVM with three kernels are presented (Table 3). Maximum accuracy achieved here is 89.12% when strategy *k* and SVM with linear kernel is considered. The minimum accuracy achieved is 72% when strategy *a* and SVM with linear kernel again, is considered. These results are also depicted in Figure 6.

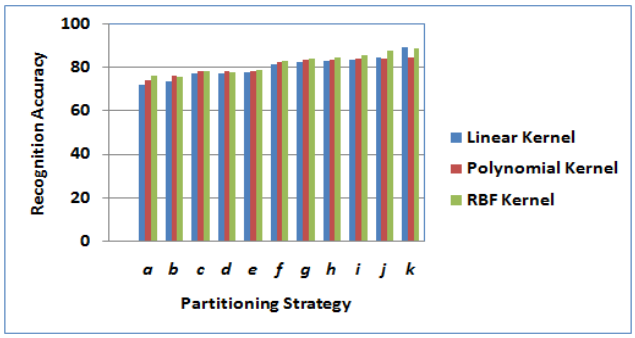

Figure 5. Effect of training set sizes on recognition performance with power curve fitting based features.

TABLE 3:

EFFECT OF TRAINING SET SIZES ON RECOGNITION PERFORMANCE WITH POWER CURVE FITTING BASED FEATURES.

Strategy	Linear Kernel	Polynomial Kernel	RBF Kernel
a	72.00%	74.24%	76.00%
b	73.36%	76.29%	75.49%
c	77.14%	78.07%	78.21%
d	77.39%	78.12%	77.79%
e	77.68%	78.14%	78.67%
f	81.28%	82.29%	83.20%
g	82.47%	83.40%	84.00%
h	83.09%	83.67%	84.38%
i	83.67%	83.92%	85.71%
j	84.28%	84.07%	87.43%
k	89.12%	84.64%	88.57%

CONCLUSION

The work presented in this paper is a study on variation of the recognition performance of the SVM classifier vis-à-vis the variation in the training set size. Eleven training strategies have been explored in this paper in order to recognize the performance of an offline handwritten Gurmukhi script recognition system. This has been noticed that irrespective of the features, the SVM classifier performs increasingly better if we increase the numbers of samples in the training data set. This claim shall be verified in our subsequent work by increasing the size of samples that as of now is 3500.

REFERENCES

[1] Bansal, V., and Sinha, R. M. K. Sinha. 2000. Integrating Knowledge Sources in Devanagari Text Recognition System. *IEEE Transactions on Systems, Man and Cybernetics- Part A: Systems and Humans*, 30, 4 (2000), 500-505.

[2] Bansal, V., and Sinha, R. M. K. 2002. Segmentation of touching and fused Devanagri characters. *Pattern Recognition*, 35, 4 (2002), 875-893.

[3] Chaudhary, B. B., and Pal, U. 1997. An OCR System to Read Two Indian Languages Scripts: Bangla and Devanagri (Hindi). In Proceedings of International conference on Document Analysis and Recognition (ICDAR), 2 (1997), 1011-1015.

[4] Hanmandlu, M., Grover, J., Madasu, V. K., and Vasikarla, S. 2007. Input fuzzy for the recognition of handwritten Hindi numeral, In Proceedings of ITNG, 2007, 208-213.

[5] M. K. Jindal, "Degraded Text Recognition of Gurmukhi Script", PhD Thesis, Thapar University, Patiala, India, 2008

[6] Kumar, D. 1991. AI Approach to Hand Written Devnagari Script Recognition, In Proceedings of IEEE Region 10th International conference on EC3-Energy, Computer, Communication and Control Systems, 1991, 229-237.

[7] Kumar, M., Sharma, R. K., and Jindal, M. K. 2010. Lines and words segmentation of offline handwritten Gurmukhi script documents. In Proceedings of International conference on IITM, 2010, 25-28.

[8] Kumar, M., Sharma, R. K., and Jindal, M. K. 2011. SVM based offline handwritten Gurmukhi character recognition, In proceedings of SCAKD, Vol. 758 (2011), 51-62.

[9] Lehal, G. S., and Singh, C. 2000. A Gurmukhi script recognition system. In the Proceedings of 15th ICPR, 2 (2000), 557-560.

[10] Lorigo, L. M. and Govindaraju, V. 2006. Offline Arabic handwriting recognition: a survey. *IEEE Transactions on PAMI*, 28, 5 (2006), 712-724.

[11] Parui, S. K., Chaudhuri, B. B., and Majumder, D. D. 1982. A procedure for recognition of connected hand written numerals. *International Journal Systems Sciences,* 13 (1982), 1019-1029.

[12] Roy, K., and Pal, U. 2006. Word-wise Hand-written Script Separation for Indian Postal automation. In the Proceedings of 10th IWFHR, 2006, 521-526.

[13] Sharma, A., Kumar, R., and Sharma, R. K. 2008. Online handwritten Gurmukhi character recognition using elastic matching. *International Journal of Congress on Image and Signal Processing*, 2 (2008), 391-396.

[14] Pal, U., and Chaudhary, B. B. 2000. Automatic recognition of Unconstrained Offline Bangla Handwritten Numerals, In the Proceedings of Advances in Multimodal Interfaces, 2000, 371-378.

[15] Pal, U., Wakabayashi, T., and Kimura, F. 2007. A system for off-line Oriya handwritten character recognition using curvature feature. In the proceedings of 10th ICIT, 2007, 227-229.

[16] Pal, U., Wakabayashi, T., and Kimura, F. 2007. "Handwritten Bangla Compound Character Recognition using Gradient Feature. In the Proceedings of 10th ICIT, 2007, 208-213.

[17] Plamondon, R. and Srihari, S. N., 2000. On-line and off-line handwritten character recognition: A comprehensive survey, *IEEE Transactions on PAMI*, 22, 1, (2000), 63-84.

[18] Rajashekararadhya, S. V., and Ranjan, S. V. 2009. Zone based Feature Extraction algorithm for Handwritten Numeral Recognition of Kannada Script. In the proceedings of IACC, 2009, 525-528.

[19] Swethalakshmi, H., Jayaraman, A., Chakravarthy, V. S., and Sekhar, C. C. 2006. Online handwritten character recognition of Devanagari and Telugu characters using support vector machine. In the proceedings of 10th International workshop on Frontiers in Handwriting Recognition (IWFHR), 2006, 367-372.

[20] Wen, Y., Lu, Y. and Shi, P. 2007. Handwritten Bangla numeral recognition system and its application to postal automation, *Pattern Recognition*, 40 (2007), 99-107.

Modeling Future Generation E-Mail Communication Model for Improving Quality of Service

M. Milton Joe
Assistant Professor, Department of Computer Application,
St. Jerome's College, Nagercoil, Tamilnadu, India.
m.miltonjoe@gmail.com

Dr. B. Ramakrishnan
Associate Professor, Department of Computer Science and Research centre,
S.T.Hindu College, Nagercoil, Tamilnadu, India.
ramsthc@gmail.com

Dr. R. S. Shaji
Professor, Department of IT, Noorul Islam University, Nagercoil, Tamilnadu, India.
shajiswaram@yahoo.com

***Abstract*— Most widely used communication medium over the internet is E-mail. E-mail is a web based application, which is used to transmit the message from one location to another location through the internet. This E-mail communication medium is used by all the people all over the world for their day to day life. Especially this web based application is mostly used in all the corporate companies, industries, small offices and educational institutions and so on. Various researches have been carried out in the E-mail web based application, to enhance the security mechanism to keep the security constraints in consistent state. However no researches have been taken to improve the Quality of Service (QoS) in E-mail application. In this paper, the concentration has been made on Quality of Service and modeled an E-mail application, which improves the Quality of Service (Qos) in the existing E-mail message handling service. The proposed model notifies to the user, whether sent E-mail reached the destination and whether the E-mail is read by the receiver, i.e. the delivery status and mail reading status are updated to the sender. The proposed model is evaluated and implemented successfully and the quality of service is obviously improved in the E-mail application model.**

***Index Terms*— E-Mail, Internet, Communication, Quality of Service, Web, Protocols, CPM.**

I. INTRODUCTION

Communication plays vital role over the internet. The main objective of internet users is to share and exchange their resources with one another. There are many communication medium available in the internet: one among them and most widely used communication is E-Mail system. E-Mail is the process of exchanging digital messages from one author to one or more recipients and in the early days both the sender and receiver must be online to send and receive the E-Mail messages [1]. The recent research development created a model called store-and-forward, in which server stores the mail and delivers it to the intended author [1].

This operation mode made neither the sender nor the receiver to be online simultaneously [1]. E-Mail communication model attracted many internet users because its communication is easy and fast [2]. Every E-Mail packet consists of body and header. The body of the E-Mail contains the original message sent by the sender and it is included with Hyper Text Markup Language (HTML) and Multi-purpose Internet Mail Extensions (MIME) encoded attachments sent by the sender [3, 4]. The header part is very important, for it serves as the envelope to forward the message from the source to destination [4]. The header part consists of fields such as 'From', 'To', 'Subject', 'Date', 'CC', 'BCC' and etc., all these fields help to forward the mail to the intended recipients [4]. The following components are needed to configure an electronic email system to work properly [5].

- Mail User Agents
- Mail Delivery Agents
- Mail Alias Files
- The Mail Queue
- Networking Topographies
- MIME Applications

Mail User Agents are software programs that run on the user machine to send and read the E-Mail messages [5]. Mail Delivery Agents run in the background and it is responsible to route and deliver the E-Mail messages and this Mail Delivery Agents are the core part of the electronic mail system [5]. Mail Alias Files used for mapping the real world names to the user login names [5]. The Mail Queue is a place used to hold the messages

doi:10.4304/jetwi.5.4.385-394

until they are sent on their way [5]. Networking Topographies used to connect group of computers for configuring electronic mail service system [5]. The MIME Applications are used to send variety of data over the E-Mail such as audio, video and other graphics in different formats. This can be achieved with MIME (Multi-purpose Internet Mail Extensions [5]. Sending and receiving of electronic mail over the internet can be done with the help of some protocols. The protocols SMTP and POP3 are the widely used protocols to send and receive the electronic mails. The following diagram depicts how the E-Mail message is routed from one source to other destination. The figure 1 consists of three components namely E-Mail Client [Sender & Receiver], Internet and Public mail servers. As shown, the sender enters the mail address and forwards the mail to the SMTP server through the internet. SMTP server forwards the mail to the POP3 server associated with the receiver's address [6]. Then receiver retrieves the mail form the POP3 server to his local machine [6].

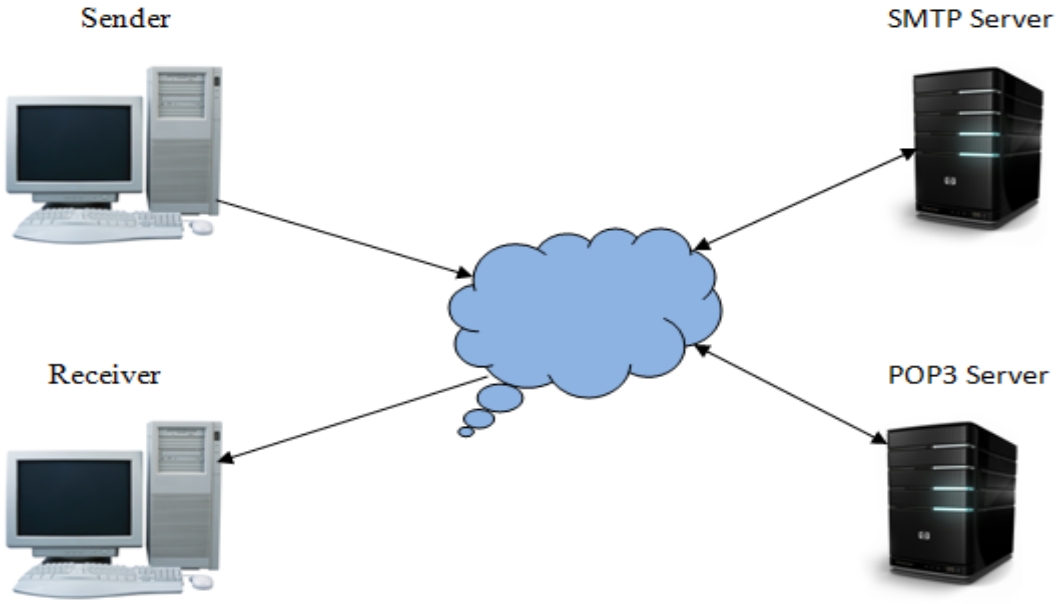

Figure 1 E-Mail Processing System

When the internet users are growing, Quality of Service (QoS) becomes the key point in the network communication. Quality of service (QoS) has a direct impact on user satisfaction with the service provided by Internet Service Providers (ISPs) [7]. E-Mail service is the one of the earliest service emerged on the internet to exchange the message from one recipient to one or more other recipient over the internet [7]. This E-Mail communication model is almost used by every one for their day to day life; however providing quality of service to all the E-Mail users is the key parameter in E-Mail message handling service. In this paper, QoS is chosen as the research area and novel mechanism is presented to improve the Quality of Service (QoS) in the E-Mail application.

II. Related Work

Every network communication must consider the following two key factors: Security and Quality of service. These two factors should be maintained in stable status to obtain better performance forever. Similarly electronic mail network communication should keep these two factors in stable condition to provide, best sort of communication over the internet. Research has been carried out in electronic mail communication especially on the two factors in various ways. The two factors security and quality of service are interconnected with each another. That is, when security is improved that should not affect the quality of service and similarly when quality of service is improved that should not affect the security constraints. Whenever research is carried out in any network communication these two factors must be keenly observed to obtain better performance. The E-mail usage is long been considered as a major facility in enabling employees to gain quick and easy access to the sources of information and contacting people [8, 9, 10, 11]. The relationship between information and work performance has been addressed with two district factors: Technological approach and Human focus approach. The technological approach deals with what new technologies can be offered [8, 12] and the human focus approach focuses on social aspect claiming that information is essential for both individuals and groups of users [8, 13, 14]. The above survey shows the use of E-Mail in organizations level only and it does not speak about quality of service in E-Mail system.

Privacy is the major issue in electronic mail services. When the user deletes the mail from the mailbox, the copy of the mail is maintained at the E-Mail Service Providers (ESPs), which does not define the E-Mail privacy policy [4, 15]. The sender in E-Mail

communication model can easily provide fake identities and even other details, this kind of faking is called spoofing [4, 16]. Spoofing is the process of changing the original sender address with the fake address and this leads to the intended user will not get the reply from the receiver side. This spoofing result in security issues and causes financial losses through forwarding like spread of virus, worms and denial of services and etc [4, 17]. Several protocols such as SMTP and S/MIME and authentication of sender domain by digital signatures are developed to prevent the spoofing and keep the security and privacy of the E-mail system in consistent state. Here, importance is given to improve the privacy and security systems in E-Mail and no measures taken to improve the quality of service in the E-Mail communication system.

Spam mail is another issue in E-Mail communication model. Unsolicited Bulk E-Mail (UBE) and Unsolicited Commercial E-Mail (UCE) are known as spam [18]. Various anti- spam techniques have been proposed to overcome the spam but that was not successful enough [19]. Three techniques have been provided to prevent the anti- spam such as: misuse detection system, anomaly detection system and social network based methods. Spamassian uses a signature based anti-spam system, which applies a prior of spam signatures [20]. These signatures are manually developed by monitoring the previous spam mails by the experts. Whenever new mail arrives the signature is matched with the new E-Mail, if mismatch found the mail is blocked as spam [18]. Observing all the previous studies once again the research has been carried out mostly on the security side of the E-Mail system.

However the security is one the factors of network communication the other factor quality of service also must be considered. Every network communication must provide user satisfaction to obtain better performance of the communication model. As everyone knows, E-Mail communication system is widely used by all the people of the world. So the E-Mail message handling system must provide good quality of service to its users. This paper concentrates on the quality of service on E-Mail system and develops the novel E-Mail communication model for improving the quality of service. The rest of the paper is divided into following sections: Section 3 gives the details of System Model (both existing and proposed system model) and proposed data flow model. The section 4 describes the details of Implementation of the proposed model. This section discusses performance evaluation, details and working principles of the proposed algorithm and results obtained by the proposed algorithm. Next section 5 is the conclusion of the proposed system model.

III. System Model

A. Existing System Model

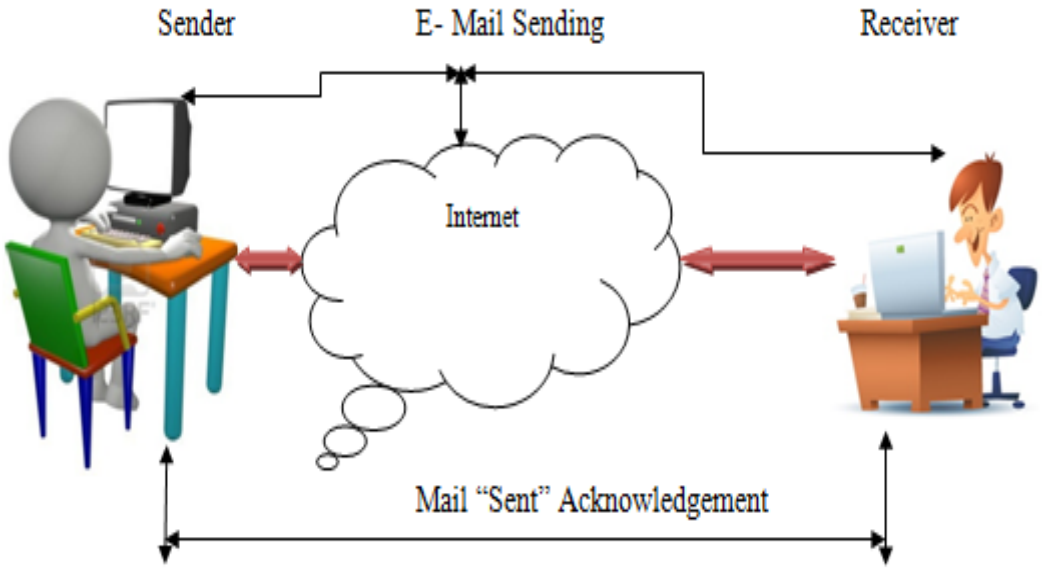

Figure 2 Existing System Model

The above figure 2 represents the existing system model of electronic mail communication. As indicated in the figure 2 both the sender and receiver are connected via internet and they can communicate with each other. This figure gives the simple representation of E-Mail message forwarding from the sender to receiver and vice versa. However, while forwarding the E-Mail message, the existing E-Mail protocols are used to transmit the message to the intended user. Namely the SMTP protocol is used to receive the mail from the sender and store the mail in the SMTP server. Then the SMTP server forwards the mail to the POP3 server associated with the intended receiver address. Later the receiver retrieves the message from the POP3 server with the help of POP3

protocol. However the existing model provides the communication without any trouble to the both sender and receiver but the existing communication model still lacks in quality of service.

B. Problem Statement

User satisfaction plays vital role in electronic mail system. The existing system can be improved to provide higher satisfaction and quality of service to the users. The key factor of network communication is Quality of Service (QoS), which is considered in this paper. In the existing system model the sender sends an E-Mail to the receiver and gets the acknowledgement of message has been sent successfully. However the user will never know, whether sent E-Mail has been delivered to the destination or not. User has to wait until the receiver sends back the response to the sender. Let's consider the case, when sender sends an E-Mail, the message is sent from the sender but the message packet is lost on the way, while forwarding to the destination. In this situation, the sender will think that the message has been successfully forwarded and sender will wait for the response from the receiver. The receiver will never send the response back to the sender because the receiver does not receive any mail from the other end. Actually the message packet is dropped somewhere. This scenario leads to poor quality of service in this existing E-Mail communication model. Quality of Service must be improved for the ease of use to the E-Mail users. This paper proposes an alternative methodology to improve the quality of service, which will obtain higher satisfaction from the users.

C. Proposed System Model

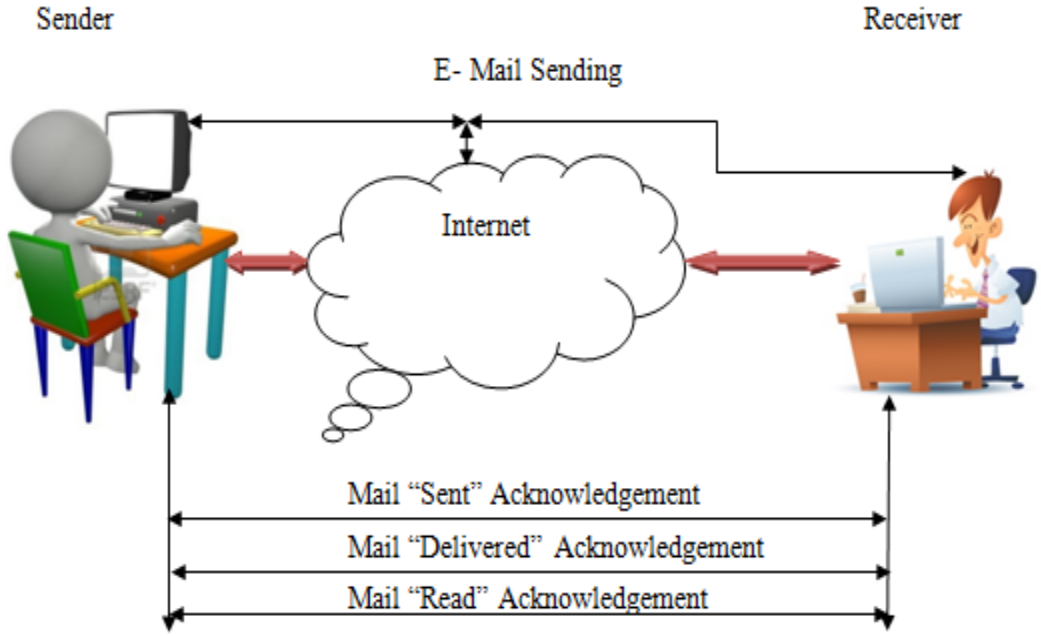

Figure 3 Proposed System Model

Figure 3 describes the entire concept of the proposed model. This proposed system overcomes limitations exists in the existing system model. As indicated in figure 3 both the sender and receiver are connected through internet to exchange E-Mail messages. In the existing model, when the sender sends an E-Mail will get only "Sent" acknowledgement and sender will never know, whether sent E-Mail reached its destination or not. This issue is cleared in the proposed model. Here, when the sender sends an E-Mail, the sender will get three acknowledgements such as message "Sent", message "Delivered" and message "Read" as shown in the figure 3. All the E-Mails sent are stored in the sent box of the user. In the proposed model, the sent box of the user will be updated with the recent status for each E-Mail is sent by the user. Initially once the E-Mail is sent the sent box is updated with the status "Sent", once the E-Mail is delivered, the sent box is updated with the status "Delivered" and once the mail is read by the receiver the sent box of the sender is updated with the status "Read". Since the proposed model clearly notifies to the users with the recent status updates the proposed system model will improve the quality of service compared to the existing model.

D. Proposed Data Flow Model

The Figure 4 shows the working data flow diagram of the proposed architecture. As shown in the Figure 4, when the mail is sent form the sender the sent box is updated as "Sent". Once the mail is delivered to the intended receiver, the sent box of the sender is updated as "Delivered". Similarly the sent box is updated with the status as "Read", once the mail is read by the receiver.

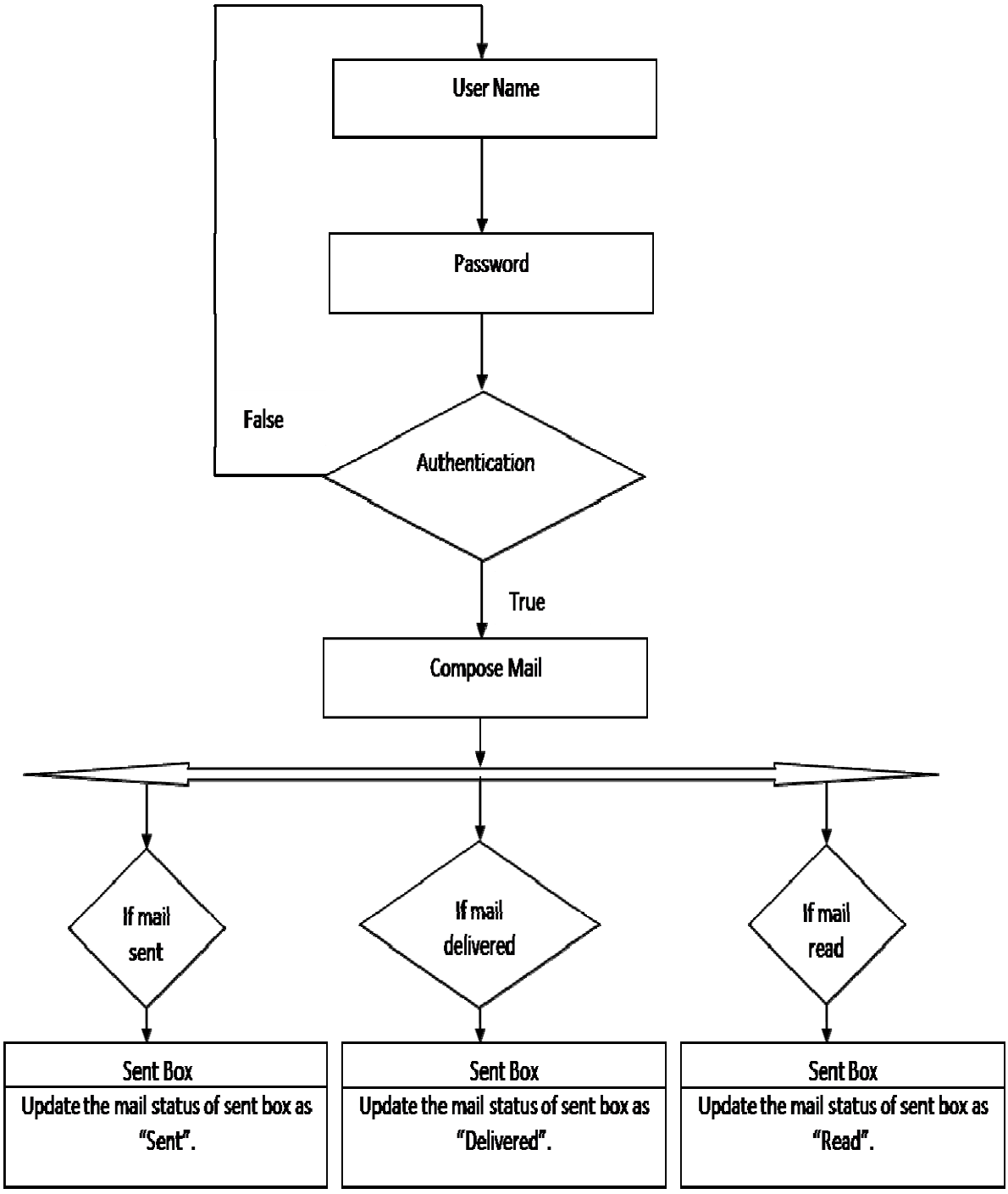

Figure 4 Proposed Data Flow Model

IV Implementation and Experimental Results

A. Performance Evaluation

The performance of the proposed E-Mail communication model can be evaluated in the following ways.

- Case 1: Updating the status field of the sent box as "Sent".
- Case 2: Updating the status field of the sent box as "Delivered".
- Case 3: Updating the status field of the sent box as "Read".

B. Case 1: Updating the Status Field of the Sent Box as "Sent"

Whenever an E-Mail is sent from the sender to receiver in the proposed model, the sent box of the sender is updated with the status "Sent" immediately.

The following algorithm represents the working principle of the updating the status field of the sent box as “Sent”.

```
Get User name;
Get Password;
If (authentication success)
{
Sign in;
}
Else
{
User name & Password mismatch;
}
Compose Mail (CPM);
Integer count=0, count1=0;
On click “Send”
{
Count=count+1;
Count1=count1+1;
Get Sender, Receiver, Subject and Message from CPM;
Forward the mail to the Server’s inbox;
INSERT Sender, Receiver, Subject, Message and count to the server’s sent box;
UPDATE the Status field of Sender’s sent box as “Sent”;
}
```

Algorithm 1: Updating the status field of the sent box as “Sent

In Algorithm 1, after composing the mail the sender enters the receiver’s mail address and press “Send” button to send the mail to its destination. As soon as the mail is sent from the sender’s local machine the status field of the sender’s sent box is updated with the status “Sent”. The following experimental result shows the working nature of the proposed algorithm.

Figure 5 Updating the status field of the sent box as “Sent”

The figure 5 shows the sent box of the user and it consists of three fields namely "To" notifies to whom the mail is being sent, "Status" notifies the current status of the mail that is being sent and "Sub" gives the subject of the mail, which has been sent. In the proposed model the "Status" field added newly to improve the quality of service in the E-Mail communication system. Whenever a mail is sent the status field is updated as "Sent" as shown in the figure 5.

C. Case 2: Updating the Status Field of the Sent Box as "Delivered"

The second metric of the proposed system updates the status field of the sent box as "Delivered" as soon as sent mail reached its destination. The following algorithm describes the entire concept of the second metric that is updating the status field of the sent box as "Delivered".

```
Get User name;
Get Password;
If (authentication success)
{
    Sign in;
}
Else
{
    User name & Password mismatch;
}
Compose CPM;
Integer count=0, count1=0;
String s=" ";
On click "Send"
{
    Count=count+1;
    Count1=count1+1;
    Get Sender, Receiver, Message and count from CPM;
    Forward the mail to the Server's inbox;
    INSERT the Sender, Receiver, Message and Count to the server's sent box;
    UPDATE the Status field of Sender's sent box as "Sent";
    SELECT sender from inbox where count=count1 and store it in s;
    UPDATE the Status field of Sender's sent box as "Delivered" WHERE count=count1 AND sender=s;
}
```

Algorithm 2: Updating the status field of the sent box as "Delivered"

The Algorithm 2 is used to update the status field of the sent box as "Delivered". When the user compose a mail and press the button "Send" to send the mail, at once the mail is sent from the local machine of the user and the status field of the sent box of the user is updated as "sent". The sent mail is routed to its intended mail address with the help of the existing protocols SMTP and POP3. Once the mail reached its correct destination, the status field of the sent box of the sender is updated with the current status as "Delivered". This could be done because the mail is delivered to its destination.

The figure 6 shows the status field of the sent box is updated with the recent status as "Delivered" when the message reached its intended destination.

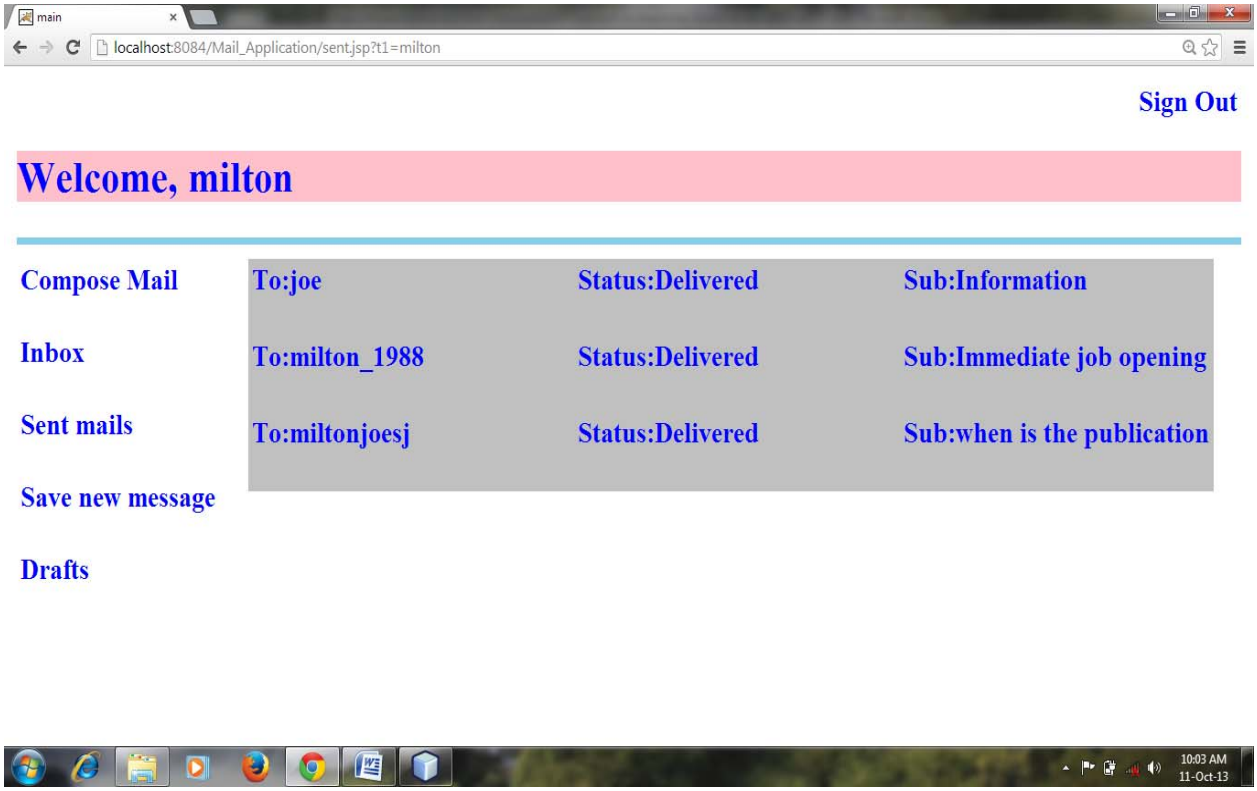

Figure 6 Updating the status field of the sent box as "Delivered"

D. Case 3: Updating the Status Field of the Sent Box as "Read"

The third metric of the proposed model will provide the reading status of the mail that has been sent. That is if the mail has been read at the receiver end, then the status field of the sent box of the sender will be updated as "Read". The following algorithm is used to obtain the third metric successfully.

```
Get User name;
Get Password;
If (authentication success)
{
    Sign in;
}
Else
{
        User name & Password mismatch;
}
String sen=" ", sub= "", msg=" ";
List of mails are displayed in the inbox;
When clicking subject of the mail the message is displayed;
On clicking the subject of the mail
{
    SELECT sender, subject, and message from inbox and store it;
    Sen=sender;
    Sub=subject;
    Msg=message;
    UPDATE the status field of the sender's sent box as "Read" WHERE sender=Sen AND subject=Sub
    AND message=Msg;
}
```

Algorithm 3: Updating the status field of the sent box as "Read"

The Algorithm 3 describes the working principle of the third metric. When the user log into his/her mail id, the inbox is displayed with the list of mails received. When the user clicks the subject of the mail, the content of the mail will be displayed to the user for reading. Using Algorithm 3, as soon the subject of the mail is clicked the sender of the particular mail, subject and the message of the mail is identified. Immediately the sender's sent box status field is updated with the recent status as "Read", since the mail has been read at the receiver end.

Figure 7 Updating the status field of sent box as "Read"

The above the figure 7 shows the sender's sent box and the status field of the sent box is updated as "Read" since the messages have been read by the receivers respectively.

Figure 8 Updating the states field with various status updates

Figure 8 shows the status updates with the combination of "Sent", "Delivered", and "Read". The message has been sent is updated as "Sent", the message that has been delivered to the intended address is updated as "Delivered" and the message that has been read at the receiver end is updated as "Read". Thus, the proposed E-Mail communication model is evaluated based on the three metrics and the experimental results show clearly, that the proposed model gives good performance. Using the proposed method, Quality of Service (QoS) is improved in the E-Mail web based application.

IV. CONCLUSION

The characteristics of the E-Mail application is completely analyzed in this paper and by observing the previous research works, it is clear that all the works have been done in the security constraints of the E-Mail communication. However Quality of Service (QoS) plays vital role in the E-Mail services. So this paper concentrated on QoS and a new E-Mail communication approach is developed. In the proposed model, all the metrics are obtained successfully, which improves the efficiency of the E-Mail service. These metrics are updating the status field of the sent box as "Sent", "Delivered" and "Read". In the existing E-Mail service, the user can identify only whether the mail has been sent or not. In the proposed model, the user can verify whether the mail has been sent, delivered and read by the receiver.

REFERENCES

[1] http://en.wikipedia.org/wiki/Email
[2] Tiago A. Almeida, Akebo Yamakami "Facing the spammers: A very effective approach to avoid junk e-mails", Expert Systems with Applications 39 (2012) 6557-6561.
[3] Resnick P, editor. Internet message format. Internet Engineering Task Force (IETF); 2001. RFC 2822.
[4] M. Tariq Banday, Farooq A. Mir, Jameel A. Qadri, Nisar A. Shah "Analyzing Internet e-mail date-spoofing", Digital Investigation 7 (2011) 145-153.
[5] http://docstore.mik.ua/manuals/hp-ux/en/5992-4607/ch07s01.html
[6] http://siis.cse.psu.edu/jpmail/jpmaildetails.html
[7] LI Yong-zhi, LIU Feng, LEI Zhen-ming, CHEN Ling, "User-perceived e-mail service QoS parameters and measurement" June 2010, 17(3): 85-90.
[8] Rita S. Mano, Gustavo S. Mesch, "E-mail characteristics, work performance and distress", Computers in Human Behavior 26 (2010) 61–69.
[9] Daft, R., & Lengel, R. (1986). Organizational information requirements, media richness and structural design. Management Science, 32, 554–557.
[10] Drucker, P. (1999). The post-capitalist society. Oxford UK: Butterworth, Heinemann.
[11] Hogg, C. (2000). Internet and E-mail Use and Abuse, Short Run, Exeter.
[12] Boisot, M. H. (1998). Knowledge assets: Securing Competitive Advantage in the Information Economy. NY: Oxford University Press.
[13] McGregor, D. (1960). The human side of enterprise. New York, NY: McGraw-Hill.
[14] Olson, M., & Lucas, H. C. (1982). The impact of office automation on the organization: Some implications for research and practice. ACM Communication, 25(11), 838–847.
[15] Oppliger R. Certified mail: the next challenge for secure messaging. Communications of ACM 2004; 47(8):75-9.
[16] Jakobsson M, Myers S, editors. Phishing and countermeasures: understanding the increasing problem of electronic identity theft, adobe e-Book. Wiley Publication, ISBN 978-0-470-08609- 4; 2006.
[17] Surmacz TR. Reliability of e-mail delivery in the era of spam. International Conference on Dependability of Computer Systems, DepCoS-RELCOMEX'07; 2007. p. 198-204.
[18] Ching-Hao Mao, Hahn-Ming Lee, Che-Fu Yeh, "Adaptive e-mails intention finding system based on words social networks", Journal of Network and Computer Applications 34 (2011) 1615–1622.
[19] Ahmed S, Mithun F. Word stemming to enhance spam filtering. In: Proceedings of the first conference on email and anti-spam; July 2004. p. 123–456.
[20] http://spamassassin.apache.org/publiccorpusS SPAMASSASSIN, TheSpa- mAssassin corpus.

Mr. M. Milton Joe received his B.Sc Computer Science degree from Bharathidasan University, India and MCA degree from Anna University, India. Presently he is working as Assistant Professor at St. Jerome's College in Nagercoil, India. He has two years of research experience and authored six research papers in reputed international journals. His research interests include Network Security, Network Communication, Vehicular Network and Social Networks.

Dr. B. Ramakrishnan is currently working as Associate Professor in the Department of Computer Science and research Centre in S.T. Hindu College, Nagercoil. He received his M.Sc degree from Madurai Kamaraj University, Madurai and received Mphil (Comp. Sc.) from Alagappa University Karikudi. He earned his Doctorate degree in the field of Computer Science from Manonmaniam Sundaranar University, Tirunelveli. He has a teaching experience of 26 years. He has twelve years of research experience and published more than twenty five international journals. His research interests lie in the field of Vehicular networks, mobile network and communication, Cloud computing, Green computing, Ad-hoc networks and Network security.

Dr. R.S. Shaji received his M.Tech in Computer Science and Engineering from Pondicherry University and PhD from Manonmaniam Sundaranar University. Presently he is working as a Professor in Noorul Islam University. He has eight years of research experience and published more than twenty five international journals. His research interests include Mobile and pervasive Network.

Multi-View Learning for Web Spam Detection

Ali Hadian* and Behrouz Minaei-Bidgoli
Department of Computer Engineering, Iran University of Science and Technology, Tehran, Iran
Email: hadian@comp.iust.ac.ir, b_minaei@iust.ac.ir

***Abstract*—Spam pages are designed to maliciously appear among the top search results by excessive usage of popular terms. Therefore, spam pages should be removed using an effective and efficient spam detection system. Previous methods for web spam classification used several features from various information sources (page contents, web graph, access logs, etc.) to detect web spam. In this paper, we follow page-level classification approach to build fast and scalable spam filters. We show that each web page can be classified with satisfactory accuracy using only its own HTML content. In order to design a multi-view classification system, we used state-of-the-art spam classification methods with distinct feature sets (views) as the base classifiers. Then, a fusion model is learned to combine the output of the base classifiers and make final prediction. Results on our Persian web spam dataset show that multi-view learning significantly improves the classification performance, namely AUC by 22%, while providing linear speedup for parallel execution.**

***Index Terms*—Web Spam, Content Spam, Machine Learning, Multi-View Learning**

I. Introduction

Web search engines are the most important tool for finding required information from the web. Processing the excessive amounts of data, even the text contents of web pages in the simplest case, is a very challenging task. Search engines should not only do this process in a few milliseconds for each query, but also have to provide high-quality and relevant results in the best order. As a part of this task, malicious and spam pages should be removed from the results. Web spam can be defined as "an unjustifiably favorable relevance or importance score for some web page, considering the pages' true value" [1]. More specifically, spam pages are defined as the pages that boost the ranking algorithm, and thus appear among the top results of the target queries. They usually deceive ranking algorithms by *keyword stuffing* of popular query terms [2].

Studies on recent web corpora have shown that spam pages, if not nullified effectively, are likely to become the majority of search results. Specifically, For queries that are frequently searched, almost all of the results are likely to be spam [3]. The ideal solution is to use ranking algorithms that are robust to spam. Unfortunately, spam-aware ranking methods are not strong enough to resist against spammers [4]. Therefore, spam pages should be removed before the ranking using complementary spam detection methods, i.e. machine learning methods. This will prevent spam pages from competing with other pages in the ranking phase.

Designing a spam detection system is a continuing process. As the spam filters evolve, new spamming techniques will be invented and new efforts will be required to detect them [4]. Web spam detection methods are still unable to reach satisfactory performance for search engines. Besides, most of the previous works propose features that are computationally expensive, such as temporal or host-level features [5]. Moreover, host-level features requires the search engine to dedicate considerable amount of resources for storing and retrieving host features.

In this work, we implemented a page-level spam detection system. We show that feature sets extracted from web pages, without using host information or any external source, can still reach satisfactory performance. The main advantage of this approach is that each page can be processed independently, which yields a linearly scalable system. In spite of the poor classification results for previous page-level methods, we show that a two stage classification with multi-view learning strategy can significantly improve the performance. Multi-view learning is a classifier fusion method in which the results of the predictive functions learned from different views on the same instances are combined to make final prediction. Three feature sets from the state-of-the art are selected as the base classification models, each one representing a different view for learning the target concept. Then, a fusion model is trained for aggregating the predictions of the base classifiers.

The remainder of the paper is structured as follows. Section 2 describes the previous work on web spam detection. In Section 3 we describe the construction of our real-world dataset. The proposed spam detection framework, including the base and meta classifiers and their results, is described in section 4. Finally, section 5 concludes our work and presents future directions.

II. Related Works

Web spam pages are created to deceive ranking algorithms of a search engine. Even though search engines use complex ranking algorithms and several information sources, ranking of web pages is based a simple rule: A web page is possibly relevant to a query if 1)its contents are similar to the query and 2)the importance of web page in the web, according to web-graph or visit rate, is relatively high. Based on this rule, spam pages are boosted by two general techniques:

doi:10.4304/jetwi.5.4.395-400

1) *Content spamming* by boosting content-based ranking algorithms, e.g. deliberate usage of keywords that are frequently searched. Using several popular keywords in the contents of the HTML pages will increase the chance of becoming relevant to more user queries. Moreover, because content-based ranking algorithms pay considerable attention to frequency of query terms in the result pages, a spam site owner may repeat popular keywords in the pages of his site. In this case, the spam pages are more likely to be relevant with popular queries, and subsequently, gain higher ranking in the query results [1, 6]

2) *Link spamming* through creating large group of pages that link to target (boosted) pages. For example, one can create a large number of automatically created web pages, all of which are linking to a target page. This will induce the link-based ranking algorithms, e.g. PageRank, to consider a high importance for the target page(s) [1].

It would be ideal if the ranking algorithm is robust against spamming. Otherwise, the ranking system should be supported by a spam filter. Link-based methods, i.e. algorithms that process the web-graph, are rich in both spam-aware ranking and spam filtering. Considerable effort been directed towards robust and spam-resilient link-based ranking, such as TrustRank [7], Anti-TrustRank [8], DiffusionRank [9], and many more [10-12]. Most of the these methods are based on trust propagation in graph, which is implemented through iteration over the web-graph. However, trust propagation algorithms are offline, and cannot rapidly investigate newly crawled web pages.

On the other hand, despite few recent efforts[13], content-based ranking methods are still defenceless against content spam [4]. Therefore, spam pages should be detected by standalone spam classification methods. Popular approaches for content-based spam detection are very similar to document classification techniques. Content-spam detection is considered as a hard classification task, because the spammers try to deceive the spam detector by manipulating the features of spam pages to make them similar to normal web pages. Nevertheless, a content-based filter is more straightforward to implement, and can be used in any search engine architecture.

From another point of view, spam detection methods can be classified to host-level and page-level methods, according to their granularity level. During the last decade, various method have been proposed for detecting web spam pages. However, research have been almost biased to host-level approach, mainly because the only publicly available spam datasets, namely *UK2006, Castillo2006* and its newer version *UK-2007* are collected with host labels. Erdelyi et al. suggest that content-based features can be very efficient for spam classification and created a simple classifier fusion system to classify host-level instances on UK2007. [14]. Similar work was done by Cormack et al. on *ClueWeb* dataset [3] with overlapping n-gram features. However, none of the above have investigated several features sets for ensembling.

III. Dataset

A. Choice of Granularity

Comparing to page-based spam classification, designing host-based classifiers is not straightforward, and has higher computational cost and more complexity for feature extraction and system integration. Aggregating page features into host features requires batch processing, which limits its scalability [15]. Moreover, it is difficult to measure the effectiveness of a host-level filter in a search engine, because some hosts, are more likely to appear in the results list, and some other hosts are less retrieved, or their contents are rarely searched by users at all.

In contrast, page-level spam filtering is simpler and more flexible. Page-level filters can be used either in crawl-time, just after fetching each page, without the need to wait for other pages from the same host, or at index-generation time. The labelling process is much easier, because the experts only need to judge a simple page rather than surfing the whole host for finding spam signatures [16]. Few research have been done on page-level filters [3, 16], and each work have followed different approaches for collecting data, labelling instances, learning method, and evaluation function.

B. Dataset Description

Web spam detection is usually considered as a two-class classification task. To create a dataset for this task, a number of instances should be assessed by users. To our knowledge, all of the works following the page-level spam classification used datasets from commercial search engines [16, 17]. The only exception is the ClueWeb09 spam scores [3] for which the original set of training labels was not published. In this work, we adopted a collection of 50 million web pages from commercial Persian search engines.

The set of labelled instances can be used as the dataset for learning and verifying the spam detection model. In order to estimate the performance of the spam detector, the dataset should be ideally an i.i.d sample of the pages retrieved by the system. The sampling strategy is quite simple: If a web page have been more often showed in the results of the queries, we would more likely consider this page in the dataset, because correctly classifying this page is more important for the system. A number of queries should be selected according to their probability of being searched. Therefore, we collected a list of frequently searched queries. This log consists of a batch of query terms searched by the users. By aggregating repeated queries, the log can be represented as the list of unique queries and the frequency of each query, which means how many times a query is searched. Let $Q = q_1,...,q_n$ be the set of unique queries, and f_i the frequency of q_i. Assuming enough size for the query log, the estimated probability of each query is

$$\hat{p}_i = \frac{f_i}{\sum_{q_i \in Q} q_i}. \quad (1)$$

After sampling a number of queries, N pages from the top results of each query are labeled by the evaluators (N=10). A group of human adjudicators evaluated the pages, and the majority vote is used as the final label. The dataset contains 5512 unique labelled instances.

Note that we could use the instance selection based on each single web page, and select each web page according to its probability of being observed among the search results of all queries. However, this will significantly degrade the classification performance, because it is likely that we have a few number of pages for each query. For instance, if we select only a single page from the results of the query "Nicole Kidman" being spam, a context-sensitive classifier will probably learn the rule that "Any page containing the terms Nicole and Kidman is spam". This will yield to biased classification models with poor generalization. Hence we have selected the queries based on their probabilities, and then evaluated all top N results of each query.

In order to quantify the agreement between raters, we used kappa statistic:

$$\kappa = \frac{P - P_e}{1 - P_e}. \quad (2)$$

where P is the agreement probability and P_e is the probability of chance agreement. Cohen's coefficient is a statistical measure for analyzing inter-rater reliability. The key advantage of kappa statistic over simple "percent agreement" is that kappa takes chance agreements into account. In our dataset, we observed $\kappa = 0.57$, which is similar to the *UK-2006* dataset ($\kappa = 0.56$) and much better than the dataset described in [18] ($\kappa = 0.45$)

IV. Spam Detection System

The system is composed of two major components. At first, the pages are classified by several content-based spam detection algorithms. Results of this classifiers will be processed as a second classifier, which makes final prediction according to the results of the base classifiers. Figure 1 shows the overall structure of our spam detection system.

A. Base Classifiers

Three distinct feature sets were used as the base classifiers. These feature sets have been have shown relatively better performance among others, as well as being fast and effective.

1) High Level Features

The classical method for spam filtering is to extract specific features from HTML structure of web pages, initially proposed by Ntoulas et al. [19]. Some of these features have shown to be very effective, namely the number of words in a page, the compression ratio of the page, average word length, title size, fraction of visible text, PageRank, and number of outlinks [19, 20]. As suggested by [14], Computationally expensive features (e.g. most of link-based features) are skipped for the sake of efficiency. Totally 30 features were selected for classification.

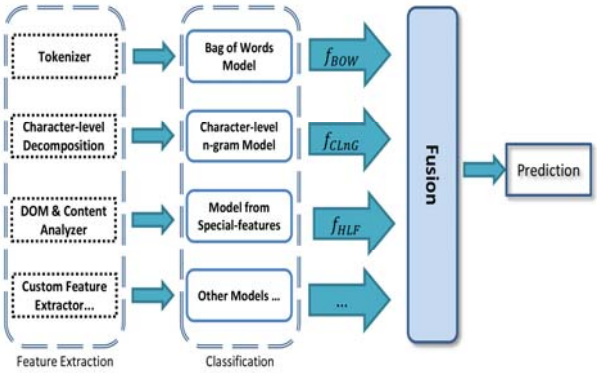

Figure 1: Web spam detection framework

Among several classification algorithms, decision tree, random forest, and KNN are reported to have the best discrimination [14, 19, 20]. Because we prefer classifiers with generative outputs, we selected KNN with K=7, using euclidean distance and inverse distance weighting. In order to accelerate the search for neighbor finding, we used KD-Tree space-partitioning data structure [21].

2) Bag-of-words

The bag-of-words model represents the document as a set of unordered terms. Each page p_i is represented as a term-frequency vector in the vector space model:

$$p_i = (TF(\omega_1), TF(\omega_2), ..., TF(\omega_N)). \quad (3)$$

where $TF_i(\omega_j)$ is the term frequency of the term ω_j in page p_i, and N is the total number of distinct terms in the corpus. To reduce the dimensionality, terms with very low frequencies were pruned, and 39,488 features remained. In the text-classification literature, three classification algorithms are mostly used for this high-dimensional learning task, namely naive-bayes, logistic regression, and linear support vector machines. While naive-bayes and (online) logistic regression are much faster for training, they suffer from the curse of dimensionality when number of training instances is small. SVM, on the other hand, is more consistent with high-dimensional datasets. The discriminating function of SVM is a hyperplane in the feature space which separates positive and negative instances. The margin between hyperplane and classifying instances is maximized by the following criteria:

$$\begin{aligned} &\min \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i, \\ &s.t. y_i(w \cdot x_i - b) \geq 1 - \xi_i \qquad \forall i \end{aligned}. \quad (4)$$

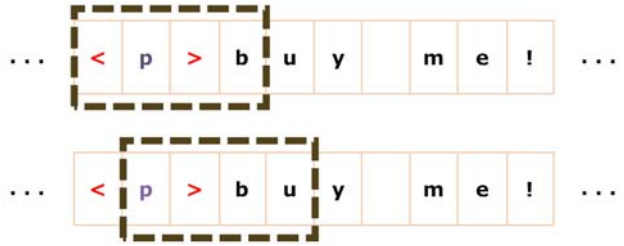

Figure 2: Character-level 4-gram extraction

where w denotes the normal vector of the discriminating hyperplane, b is a scalar vector, and C is a trade-off between accuracy and margin size. The discriminant function is $w \cdot x - b$ for an incoming feature vector x. We used popular SVM implementation from SVMLight[1] project, which is very fast for high-dimensional data [22].

3) Character-level 4-grams

In this model, any sequence of 4 characters represents a feature. It was primarily used for email spam detection task [23], and further proposed for classifying web spam [3, 17]. To extract the features, a 4-character width window starts from the beginning of the page, sliding one character per step. For instance, as visualized in figure 2, the extracted 4-grams for "< p> buy me!" are "< p> b", "p> bu", "> buy", "buy!", "uy! ", "y! m", and "! me". Then, each 4-character string is converted to its equivalent memory representation as a byte sequence, and the byte sequence is treated as a long number. This number can be used as the ID of the feature. However, because of the extremely high dimension of such feature set, features are reduced to a 10^6 dimensional space by simply dividing their ID by 10^6, and using the remainder as the ID in the new feature space [3]. Again, we used linear SVM for classification.

Table 1 shows the performance of the base classifier. To measure our performance, we performed five-fold cross validation for all of the experiments.

B. The Meta-classifier

In this step, outputs of the base classifiers are combined to make the prediction, such that

$$\hat{c} = f_{meta}(f_{HLF}(x), f_{BOW}(x), f_{CL4G}(x)). \quad (5)$$

If all of the three classifiers could generate probability or log-odds estimates, we would be able to use simple fusion operators, such as naive Bayes combination. However, because we used discriminative SVM classifiers for a couple of feature sets, we can not use probabilistic methods for combination. As shown in figure 3 , the prediction of f_{HLF} (KNN) is probabilistic estimate, while the two other classifier output discriminative values. This complex feature space requires a proper model for effective classification. In this case, we use outputs of the three classifiers as a feature set for training the meta-classifier.

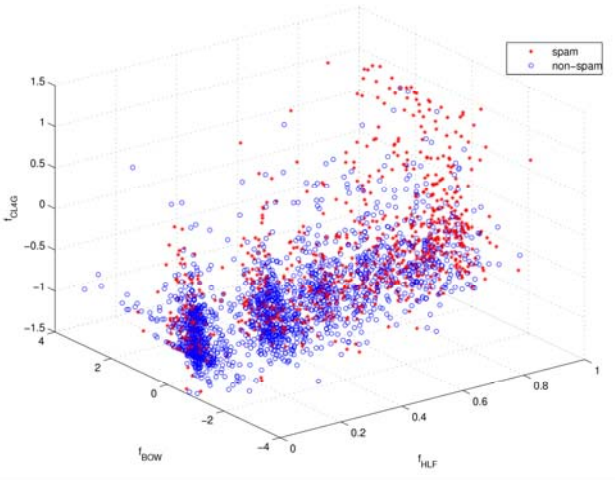

Figure 3: Feature space of meta-classifier

Several classification methods are tested for combining the classifier outputs. Again, we used 10-fold cross validation for validating the performance of meta-classifier. The results are given in Table 2. Except for recall, random forest classifier outperforms other methods. Comparing to the best results of base classifiers, the meta-classifier has increased the performance, both in terms of AUC and F-measure, by 22% and 15%, respectively. The result is not surprising, as it was previously shown that boosting-based classification methods, such as random forest, are robust and effective for adversarial environments [24].

The great advantage of multi-view learning is its capability to use the best possible classification algorithm independently for each of the discriminant feature sets. For instance, we benefited from SVM's ability to learn in high-dimensional feature sets, while using different models, i.e. KNN, for smaller-sized feature sets. However, if any of the base models is updated, the meta-model should be retrained.

C. Parallelization

Total classification time per page is 31 milliseconds in average. Despite the host-level classification models in

TABLE I.
PERFORMACE OF BASE CLASSIFIERS

Feature set	Classifier	Precision	Recall	F-measure	AUC
HTML features	KNN (N=7)	0.4944	**0.5636**	**0.5268**	**0.6789**
Bag-of-Words	SVM	0.4769	0.2097	0.2913	0.6467
Character-level n-grams	SVM	**0.6087**	0.1245	0.2068	0.6114

TABLE II.
PERFORMANCE OF CLASSIFICATION ALGORITHMS FOR ENSEMBLING

Classifier	Precision	Recall	F-measure	AUC
Random Forest	**0.892**	0.552	**0.682**	**0.901**
Decision Tree (C4.5)	0.71	0.484	0.575	0.793
KNN (Best K=5)	0.704	**0.652**	0.677	0.804
Weighted Naive Bayes	0.565	0.473	0.515	0.708
Weighted Logistic Regression	0.511	0.624	0.562	0.716
Bayes Net	0.523	0.612	0.564	0.737

which concurrent access to the web graph is a speedup bottleneck, the proposed method is linearly scalable. Multiple spam detection systems can be run in parallel, each for processing a batch of web pages, with no interrelation. In our industrial experiment, using three 12-core machines which classified the crawled pages in a pipeline, we reached our desired throughput, that is roughly a thousand pages per second.

VI. CONCLUSIONS AND FUTURE WORK

In this paper, we proposed an ensemble learning method for classifying spam web pages. Taking a page-level approach allows efficient and scalable classification of web pages, because the system only requires the page content to predict its spamness. Therefore, multiple instances of the spam detection module can be used in parallel without synchronization, which brings linear speedup. We used three state-of-the-art solutions for spam classification, and trained a meta-classifier to enhance their performance. Results show that this ensembling method successfully outperforms its base classifiers by a large margin. We are currently extending the algorithm with effective feature sets and more robust classification models, and preliminary results are encouraging.

ACKNOWLEDGMENT

The authors benefited from discussions with Azadeh Shakery, Ali Mohammad Zareh-Bidoki, Carlos Castillo, Gordon Cormack, Brian Davison, and Saeed Shahrivari. They also wish to thank Hadi Sharifi, Amin Nikookaran, and Ali Shirvani for their collaboration.

REFERENCES

[1] Z. Gyongyi and H. Garcia-Molina, "Web spam taxonomy," in *Proceedings of the first international workshop on adversarial information retrieval on the web*, 2005.

[2] C. Castillo and B. D. Davison, "Adversarial web search," *Information Retrieval*, vol. 4, no. 5, pp. 377–486, 2010.

[3] G. V. Cormack, M. D. Smucker, and C. L. A. Clarke, "Efficient and effective spam filtering and re-ranking for large web datasets," *Information Retrieval*, pp. 1–25, 2010.

[4] F. Raiber, "Adversarial content manipulation effects," in *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, ACM, 2012, p. 993.

[5] M. Erdélyi and A. A. Benczúr, "Temporal Analysis for Web Spam Detection: An Overview," in *Temporal Web Analytics Workshop TWAW 2011*, 2011, p. 17.

[6] Z. Gyongyi and H. Garcia-Molina, "Spam: It's not just for inboxes anymore," *IEEE Computer Magazine*, vol. 38, no. 10, pp. 28–34, 2005.

[7] Z. Gyongyi, H. Garcia-Molina, and J. Pedersen, "Combating web spam with trustrank," in *Proceedings of the 30th International Conference on Very Large Data Bases*, Morgan Kaufmann, 2004, pp. 576–587.

[8] V. Krishnan and R. Raj, "Web spam detection with Anti-Trust Rank," in *Proceedings of the 2nd International Workshop on Adversarial Information Retrieval on the Web (AIRWeb)*.pp. 37–40, 2006.

[9] H. Yang, I. King, and M. R. Lyu, "Diffusionrank: a possible penicillin for web spamming," in *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2007, pp. 431–438.

[10] R. Baeza-Yates, P. Boldi, and C. Castillo, "Generalizing pagerank: Damping functions for link-based ranking algorithms," in *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2006, pp. 308–315.

[11] L. Becchetti, C. Castillo, D. Donato, R. Baeza-Yates, and S. Leonardi, "Link analysis for web spam detection," *ACM Transactions on the Web (TWEB)*, vol. 2, no. 1, pp. 1–42, 2008.

[12] Y.-j. Chung, M. Toyoda, and M. Kitsuregawa, "Identifying spam link generators for monitoring emerging web spam," in *Proceedings of the 4th workshop on Information credibility*. ACM, 2010, pp. 51–58.

[13] M. Bendersky, W. B. Croft, and Y. Diao, "Quality-biased ranking of web documents," in *Proceedings of the fourth ACM international conference on Web search and data mining*. ACM, 2011, pp. 95–104.

[14] M. Erdélyi, A. Garzó, and A. A. Benczúr, "Web spam classification: a few features worth more," in *Proceedings of the 2011 Joint WICOW/AIRWeb Workshop on Web Quality*, ACM, 2011, pp. 27–34.

[15] X. Geng, X. B. Jin, and D. Zhang, "Evaluating web content quality via multi-scale features," in *Proceedings of the ECML/PKDD 2010 discovery challenge*, 2010.

[16] K. M. Svore, Q. Wu, C. J. C. Burges, and A. Raman, "Improving web spam classification using rank-time features," in *Proceedings of the 3rd international workshop on Adversarial information retrieval on the web*. ACM, 2007, pp. 9–16.

[17] Y. Liu, F. Chen, W. Kong, H. Yu, M. Zhang, S. Ma, and L. Ru, "Identifying Web Spam with the Wisdom of the Crowds," *ACM Transactions on the Web (TWEB)*, vol. 6, no. 1, pp. 1–30, 2012.

[18] A. A. Benczúr, K. Csalogány, T. Sarlós, and M. Uher, "SpamRank: Fully Automatic Link Spam Detection Work in progress," *Proceedings of the First International Workshop on Adversarial Information Retrieval on the Web (AIRWeb)*, 2005.

[19] A. Ntoulas, M. Najork, M. Manasse, and D. Fetterly, "Detecting spam web pages through content analysis," in *Proceedings of the 15th International Conference on the World Wide Web*. Edinburgh, Scotland: ACM, 2006, pp. 83–92.

[20] D. Fetterly, M. Manasse, and M. Najork, "Spam, damn spam, and statistics," in *Proceedings of the 7th International Workshop on the Web and Databases*, 2004.

[21] J. H. Friedman, J. L. Bentley, and R. A. Finkel, "An algorithm for finding best matches in logarithmic expected time," *ACM Transactions on Mathematical Software (TOMS)*, vol. 3, no. 3, pp. 209–226, 1977.

[22] T. Joachims, "Making large scale SVM learning practical," *Advances in Kernel Methods - Support Vector Learning*, 1999.

[23] G. V. Cormack, "University of waterloo participation in the TREC 2007 spam track," in *Sixteenth Text REtrieval Conference (TREC-2007)*, vol. 100, 2007.

[24] B. Biggio, G. Fumera, and F. Roli, "Multiple classifier systems for robust classifier design in adversarial environments," *International Journal of Machine Learning and Cybernetics*, vol. 1, no. 1-4, pp. 27–41, 2010.

LSI Based Relevance Computation for Topical Web Crawler

Gurmeen Minhas
UIET, Panjab University, Chandigarh, India
Email: gurmeenminhas@gmail.com

Mukesh Kumar
UIET, Panjab University, Chandigarh, India
Email: mukesh_rai9@yahoo.com

***Abstract*—Today, size of the web is exceptionally large. And this size is increasing rapidly. Huge number of web pages and web sites are being added each day. Hence, results which are effective, factual and authentic are needed. A simple crawler cannot cover each web page as it would take polynomial time to do so. In order to overcome such issues, this paper proposes an algorithm to develop an efficient, focused, domain specific crawler using LSI (Latent Semantic Indexing). This algorithm makes the crawler highly efficient in downloading relevant documents, thus, avoiding over-heads and resource wastage, and also increases the precision and recall values of the IR system developed on it.**

***Index Terms*— Crawling, focused crawler, latent semantic indexing, domain specific crawler.**

I. Introduction

The World Wide Web Worm (WWWW) is one of the first web engines. According to a survey in 1994, WWWW had an index of 110,000 web pages and web documents [9]. However creating a search engine which fulfils the present day web requirements is a challenging task. Today we need a fast crawling technology to gather the web documents and keep them up to date. A web search engine is designed to search for information on the World Wide Web [9]. A search engine basically has three steps, crawling the web, indexing and searching.

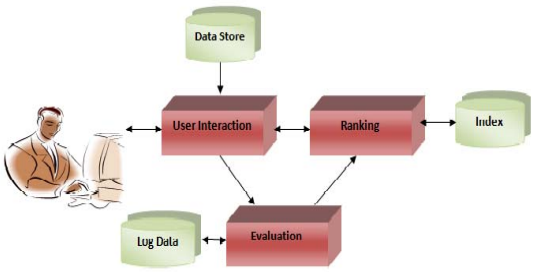

Figure 1. General Searching Process

Crawling the web means to download the documents present on the web which can be used for later queries. This task is performed by a web crawler. A software or a computer program that browses the World Wide web in a procedural, mechanized manner or in an orderly form is known as a web crawler. Other terms for web crawlers are ants, automatic indexers, bots , web spiders, web robots. The process is called web crawling or spidering [4]. Search engines use crawling as a part of its process to store and provide up-to-date data. Main work of a web crawler is to create a copy of all the pages it visits, for later processing by the search engine. Index is created in search engines so that the data is organized and quick results can be given to the user for their queries. Indexes are built based on the number of instances and position of particular words and then efficient ranking is implemented. The ranking of web pages are usually based on various factors such as number of times a word is being used in a document or the semantic structure of the content etc. Some ranking algorithms form the basis to calculate the score of the documents. The documents are ranked so that more relevant results are returned to the user in response to the user's query. The query input is taken from the user through the user interface of a search engine.

Focused search engines are domain-specific search engines which reduces the search margin which somehow increases the search accuracy. A focused search engine has a focused crawler at its heart, which gathers and updates information from the web [34]. A focused crawler is also known as topical crawler. Topical crawlers move over all web pages which are related to a particular subject, beginning from some relevant seed pages. The topical crawlers while travelling the web will analyze each hyperlink and try to figure out which link may be relevant to the subject. The relevant links are chosen and the irrelevant ones are abandoned [34]. Therefore, a focused crawler is the one which attempts to download only those web pages which are relevant to a pre-defined topic. A Focused Crawler is described as a mechanism which seeks, acquires, indexes, and maintains pages on a specific set of topics that represent a relatively narrow segment of the Web [6].

II. Related Work

doi:10.4304/jetwi.5.4.401-406

Due to the broad size of the web and the general crawling and indexing mechanism, results achieved are of low precision. As a result of this, the present day scenario demands specialized and focused crawlers. This section discusses some of the methods that have been used for the purpose of information retrieval and for constructing focused crawlers.

A. Backward and Forward Link Count

The idea that the number of times a paper is cited, has an impact on the importance of that particular paper, and this forms the basis for the concept of link counts [16]. So it is commonly regarded, that, a page that is linked by many other pages on the web will be more useful as compared to the page that is linked by a lesser number of other pages that is, a page which is referred scarcely is considered less important.

Suppose that we have a web page say P, and I(P) is the measure of importance of page P. So according to backward-link count metric, the importance I(P) of page (P), will be measured by the number of other pages on the web that have links pointing to page P, as shown in figure 2.

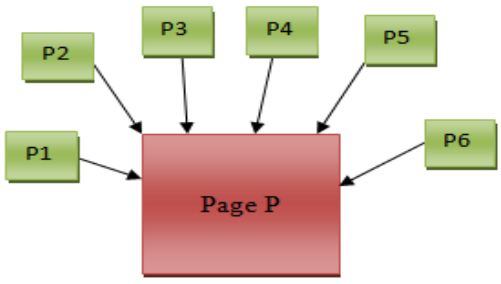

Figure 2. Backward-Link count

The other metric is the forward-link count shown in figure 3. According to this metric, a page that contains many outgoing links is treated important, since it may be a web directory or a web resource depository.

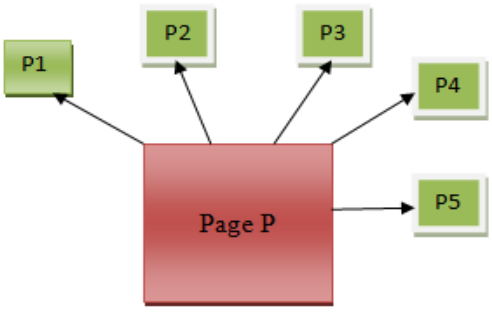

Figure 3. Forward-Link count

B. Page Rank

The original page rank metric was described by Sergey Brin and Lawrence Page [9]. To estimate the importance of a page, this metric makes use of the link structure of the page. Suppose

I(P)= page rank value of page P

I(Q)= page rank value of any page Q

B(P)= set of all pages that have a link pointing to P

c(Q)= number of all links in page Q

$I(P) = \sum_{Q \in B(P)} I(Q) / c(Q)$

Therefore, page rank of page P, that is, I(P), depends on the page rank value of page Q, that is, I(Q), where Q belongs to the set B(P), divided by number of links in page Q.

C. HITS(Hyperlink-Induced Topic Search)

Hyperlink-Induced Topic Search (HITS) is also known as hubs and authorities. HITS makes use of the link structure of the web, in order to discover and rank pages relevant for a particular topic. This algorithm was originally developed by Jon Kleinberg[10].

According to Jon Kleinberg [10] , the way a human user searches a document is very contrary and complicated than the process of just matching a list of query words with a list of documents. In this scheme, every web page is assigned two scores- hub score and authority score. So corresponding to a query two ranked lists are made [10]. The ranking of one list is based on hub scores and ranking of the other list is based on authority scores. Let us imagine that we have a query, "facts about formula 1 racing car ". The official page of formula 1 would be the authoritative source of information on the topic. Such pages will be called authorities. On the other hand there must be many web pages which would be containing lists of links to the authoritative web pages on a particular topic. These are known as hub pages. These hub pages are not sources of topic specific information but accumulation of links on that topic. A good hub page is the one that points to many good authorities, and a good authority page is one that is pointed to by many good hub pages. Suppose we have a set of web pages which are good hubs and authorities and these are having hyperlinks among them. The hub score and authority score for every web page is calculated. We have a web page v , hub score h(v) , authority score a(v) , $v \rightarrow y$ means a hyperlink exist from v to y. We have the following equations[1]

$h(v) \leftarrow \sum_{v \rightarrow y} a(y)$

$a(v) \leftarrow \sum_{y \rightarrow v} h(y)$

According to the first equation hub score of page v is equal to the sum of authority scores of the pages it links to. So if v links to pages with high authority scores then its hub score will be high. According to second equation if page v is linked to by good hubs then its authority score will be high.

D. Text Categorization

Text categorization is also known as text classification, or topic spotting. It is the task of automatically sorting a set of documents into categories or classes or topics from a predefined set.

Text categorization is a supervised learning task in which pre-defined category labels are assigned to new documents based on a training set of labelled documents [11]. Yaug and Liu in [11] have discussed five categorization models which are Support vector machines(SVM), K-NearestNeighbour classification, Linear Least Squares Fit (LLSF), Naive Bayes Classifier(NB), andNeural Network Techniques(NNet).

E. Boolean Queries

Among the traditional methods of information retrieval is Boolean Retrieval model. The Boolean retrieval method uses Boolean operators and is the most straightforward technique of retrieval. The basis of Boolean model is set theory and Boolean algebra. Documents are expressed by the terms extracted from documents and queries are expressed as Boolean expressions. This model consists of a query which is just as a set of words. The queries usually consists of AND, OR, NOT. This model is an exact-match retrieval model which means that the query should be clear-cut and a document either matches the query or it does not, that is, result is 0 or 1.

Joon et all[15] have proposed a ranking algorithm which is thesaurus based that measures the relevance of documents and return the top ranking documents. This algorithm called as E-relevance algorithm gives the similarity score between a query and document[15].

F. Vector Space Model

Vector Space Model represents text documents as vectors. It is different from Boolean retrieval. Boolean retrieval method assigns binary weights 0 or 1, whereas, vector space model assigns non-binary weights to terms in documents and queries. Depending on these weights, the degree of similarity or the inter-relationship between a query and document is found out. According to this model, a space is created, in which both documents as well as queries are represented as vectors. The dimension of the vector is equal to the number of unique terms present in the document. Weights are assigned to terms and this weight is usually based on the number of occurrences of a term in a document, and, this is known as term-frequency. The other weighing scheme mostly used is the tf-idf, where idf is inverse document frequency .

G. Ontology based retrieval

An ontology represents knowledge as a set of concepts and relationships between those concepts for a specific domain. Ontology is a semantic based retrieval technique which understand the meaning of the concept of the user query. Ontologies are arranged in a taxonomy of concepts. Ontology includes description of concepts and its properties. It also describes various features and aspects of the concept.

H. Latent Semantic Indexing

Generally when we retrieve information it is based on exact matching, that is, the terms in the query are matched to those in the document. But sometimes we have certain documents which are relevant to the query but does not contain the exact words as present in the query. So in such cases it is advisable to use a mechanism that helps us to retrieve documents on the basis of conceptual meaning of the query and document. For this we use the concept of Latent Semantic Indexing. Latent Semantic Indexing is also known as Latent Semantic analysis. LSI is a technique that enables us to analyse relationships between terms and concepts occurring in a text. LSI uses a mathematical technique called Singular Value Decomposition (SVD). The main element of LSI is its ability to extract the conceptual content of text by building associations between the terms that have similar contexts. This technique is so called because it has the ability to relate terms that are semantically similar in some text. It uncovers the latent semantic structure of words in a text corpus. When a query is issued on a set of documents on which LSI has been applied, the results that we get will be the ones which are conceptually similar to the query even if the results do not contain same specific words. Latent Semantic Indexing starts with a term by document matrix. Then, Singular Value Decomposition (SVD) is used to decompose the term by document matrix into three matrices: T, a term by dimension matrix, S, a singular value matrix (dimension by dimension), and D, a document by dimension matrix. The number of dimensions is r, that is the rank of the term by document matrix. The original matrix can be obtained, through matrix multiplication of TSDt. In an LSI system, the T, S and D matrices are truncated to k dimensions [19].

III. IMPLEMENTATION AND GRAPHS

The objective of our work includes the development of a term corpus specific to CAD (Computer Aided Design) domain. After which a crawling algorithm is developed which works on the scoring system based on LSI(Latent Semantic Indexing). Finally we evaluate performance by comparing anchor and document scores relevant for crawling to simple breadth-first algorithm (algorithm 1) and keyword based approach (algorithm 2). Our focused crawler builds its corpus , which is specific to CAD domain. Therefore, it is a model that works on the principle of selecting only those web documents, from whom as per algorithm 3 (LSI) , it can gain information with respect to CAD domain only. In this process it is intuitively reducing the uncertainty about the category of a document item being selected for crawling provided by knowing the value of feature Y. Here item Y are the seed keywords or URLs or future hyperlinks or the titles.

Since the ultimate goal of algorithm 3 or our focused web crawler is to build a dataset that would provide a high information gain when used by a search engine or query engine , the selection of URLs and keywords is very important as it would lead to burning of less resources. We are taking the advantage of highly optimized anchor tags , and also taking the advantage of vector semantic model in algorithm 3. By doing the above process , we thus improve the recall and precision of our overall system.

It is apparent from the graph for recall analysis, Figure 4 that the recall value varies from 28% to 40.5%, which reflects the completeness or sensitivity of our algorithm 3. The recall value here means less number of crawl jobs that are false negative in nature , or in simple words , crawling less number of web documents that were selected erroneously or those web URLs which were supposed to be rejected but got selected in URL crawl priority queue.

However, it can also be seen from the precision graph, Figure 5 that value remains around 59.4% and 66.03% which is otherwise difficult to obtain had not the algorithm 3 been implemented , because normally if recall value increases (in our case it is moderate) the precision often decreases, as it gets harder to precise when the sample space increases. But , in our result we can see that precision remains moderate , that means around 60% crawls are true positive in nature , or in simple words, the web documents which were supposed to be in priority queue were correctly selected.

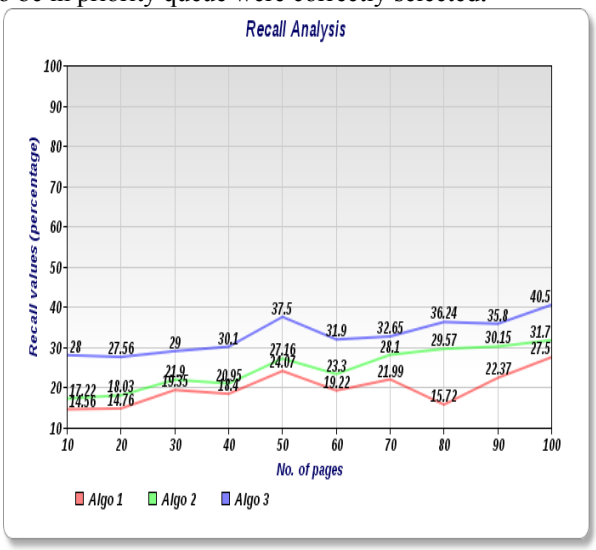

Figure 4. Recall Analysis Line Graph

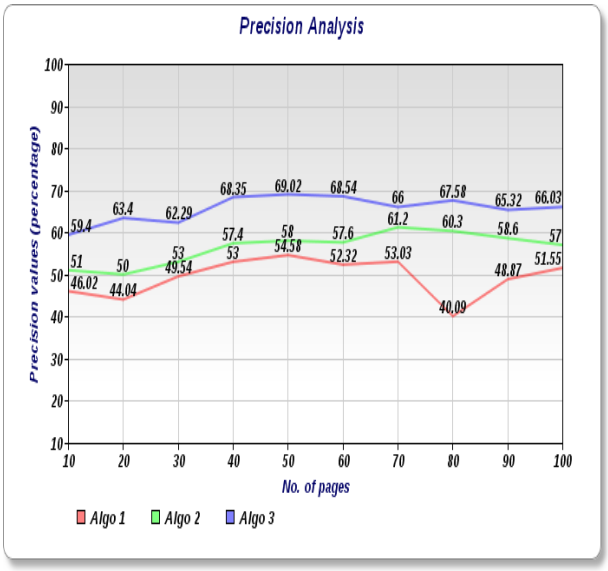

Figure 5. Precision Analysis Line Graph

IV. Conclusion

In this paper a domain specific focused crawler has been implemented. A domain specific crawler is useful for saving time and other resources since it is concerned with a particular domain. Hence we obtain highly relevant data which leads to high information gain and less resource wastage. Being in the field of engineering , computer aided design has been chosen as a domain to work on. Various methods of information retrieval have been studied and reviewed, and based on this literature survey, it was found that there is a requirement to build a crawler that takes into account the context of the words or phrases being searched for. LSI(Latent semantic Indexing) model is one such promising model in the field of information retrieval. LSI uses a mathematical technique known as Singular Value Decomposition. This model has the ability to extract the conceptual content of a body of text by looking for relationships between the terms of the text. The evaluation of the work has been done by using the recall and precision values. The values of my crawler has been compared to the recall and precision values of two other crawlers, which are, breadth first crawler and keyword based crawler. A breadth first crawler crawls the pages in the order they are encountered, without taking into account the relevancy and importance, as it continues in the direction wherever it finds the next link. A keyword based crawler makes use of the keywords supplied to the crawler. If the fetched pages contain 20% of these keywords, then that page is considered as relevant otherwise not. As apparent from the graph, precision values of a simple breadth first crawler is the least , then comes the keyword based crawler and finally the LSI based crawler. Hence it is clear that the performance of LSI based crawler is the most superior.

V. Future Scope

These days many information retrieval systems are being created based on taxonomies, ontologies, knowledge bases. The users want information based on

particular domains which would help them save time and effort and would help them retrieve more relevant and useful results. However there is still lot to do in the field of domain specific web crawlers. Creation of more domain based crawlers in future is suggested in various areas such as chemistry , biology , medicine , etc. We can also add other machine learning algorithms like probabilistic algorithms , neural network etc which may result in even better precision.

REFERENCES

[1] Christopher D. Manning, Prabhakar Raghavan and Hinrich Schütze, "Introduction to Information Retrieval", Cambridge University Press. 2008.

[2] http://en.wikipedia.org/wiki/Information_retrieval.

[3] Dagobert Soergel "'Information Retrieval The scope of IR" , HCL Encyclopedia .

[4] http://en.wikipedia.org/wiki/Web_crawler.

[5] S.S. Dhenakaran and K. Thirugnana Sambanthan, " Web crawler - an overview", International Journal of Computer Science and Communication Vol. 2, No. 1, January-June 2011, pp. 265-267.

[6] Soumen Chakrabarti , Martin van den Berg , Byron Dom, " Focused crawling: a new approach to topic-specific Web resource discovery", Computer Networks 31 (1999) 1623–1640 (Elsevier).

[7] Qu Cheng, Xiamen Univ, Xiamen Wang Beizhan , Wei Pianpian, " Efficient focused crawling strategy using combination of link structure and content similarity ", IEEE International Symposium on IT in Medicine and Education, 2008.

[8] J. Cho, Hector Garcia Molina, Lawrence page, "Efficient Crawling through URL Ordering", paper presented at 7th international WWW Conference. April 1998. Brisbane, Australia.

[9] Sergey Brin , Lawrence Page, "The anatomy of a large-scale hypertextual Web search engine", Computer Networks and ISDN Systems 30 (1998) 107- 117 (Elsevier).

[10] J. Kleinberg, Authoritative sources in a hyperlinked environment, in: Proceedings of the Ninth Annual ACM-SIAM Symposium, Discrete Algorithms, January 1998, pp. 668-677.

[11] Yiming Yang, Xin Liu, " A re-examination of text categorization methods", Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval 1999.

[12] M,Deligenti, F Coetzel , S Lawrence , C Leegiles , M Gori , "Focused Crawling using Context Graphs" , presented in 26th International Conference on Very Large Databases,Cairo , Egypt,2000.

[13] Fan Wu, Ching-Chi Hsu, "Topic-specific crawling on the Web with the measurements of the relevancy context graph", Information Systems Volume 31 Issue 4-5, June, 2006. (Elsevier).

[14] D. Gibson, J. Kleinberg , "Inferring web communities from link topology" , Proceedings of the 9th ACM Conference on Hypertext and Hypermedia (HYPER-98), 1998, pp. 225–234.

[15] Joon ho lee, myoung ho kim, and yoon joon lee , "Ranking documents in thesaurus-based boolean retrieval systems", Information Processing and Management Vol. 30, No. 1. PP. 79-91. 1994.

[16] Wenlei Mao, Wesley W. Chu, " The phrase-based vector space model for automatic retrieval of free-text medical documents" , Data & Knowledge Engineering Volume 61 Issue 1, April, 2007 Pages 76-92 (Elsevier).

[17] Jibran Mustafa, Sharifullah Khan, Khalid Latif, "Ontology Based Semantic Information Retrieval", 4th International IEEE Conference on Intelligent Systems, 2008.

[18] Alex Thomo, "Latent Semantic analysis Tutorial" , http://www.engr.uvic.ca/~seng474/svd.pdf.

[19] April Kontostathis a and William M. Pottenger b, A Framework for Understanding Latent Semantic Indexing (LSI) Performance, International journal on Information Processing and Management, Volume 42, january 2006 (Elsevier).

[20] Deerwester, S. C., Dumais, S. T., Landauer, T. K., Furnas, G. W., & Harshman, R. A. , "Indexing by latent semantic analysis" in Journal of the American Society of Information Science,1999.

[21] Mohsen Jamali, Hassan Sayyadi, Babak Bagheri Hariri and Hassan Abolhassani, " A Method for Focused Crawling Using Combination of Link Structure and Content Similarity", Proceedings of the 2006 IEEE International Conference on Web Intelligence.

[22] Hongfei Yan Jianyong , Hongfei Yan , Jianyong Wang , Xiaoming Li , Lin Guo, "Architectural design and evaluation of an efficient Web-crawling system ", The Journal of Systems and Software 60 (2002) 185–193 (Elsevier).

[23] Anshika Pal, Deepak Singh Tomar, S.C. Shrivastava, " Effective Focused Crawling Based on Content and Link Structure Analysis", International Journal of Computer Science and Information Security, Vol. 2, No. 1, June 2009.

[24] Knut Magne Risvik *, Rolf Michelsen, " Search engines and Web dynamics", Computer Networks 39 (2002) 289–302 (Elsevier).

[25] Yajun Du ; Zhanshen Li , " The New Clustering Strategy and Algorithm Based on Latent Semantic Indexing", Natural Computation, 2008 . IEEE Fourth International Conference.

[26] A. Rungsawang, N. Angkawattanawit, " Learnable topic-specific web crawler", Journal of Network and Computer Applications 28 (2005) 97–114 (Elsevier).

[27] Zhumin Chen, Jun Ma, Jingsheng Lei, Bo Yuan, Li Lian , Ling Song, " A cross-language focused crawling algorithm based on multiple relevance prediction strategies", Computers and Mathematics with Applications 57 (2009) 1057_1072 (Elsevier).

[28] Robert C. Miller, Krishna Bharat, " SPHINX: a framework for creating personal, site-specific Web crawlers" Computer Networks and ISDN Systems 30 (1998) I I9- I30.

[29] Alexandros Batzios , Christos Dimou, Andreas L. Symeonidis, Pericles A. Mitkas, "BioCrawler: An intelligent crawler for the semantic web", Expert Systems with Applications 35 (2008) 524–530 (Elsevier).

[30] Cioara T. , Anghel I. , Salomie I. , Dinsoreanu M. , "A context-based Semantically Enhanced Information Retrieval Model", IEEE 5th International Conference on Intelligent Computer Communication and Processing, 2009.

[31] Aidan Hogan, Andreas Harth, Jürgen Umbrich, Sheila Kinsella, Axel Polleres, Stefan Decker ,"Searching and browsing Linked Data with SWSE: The Semantic Web search Engine", Web Semantics: Science, Services and Agents on the World Wide Web, Volume 9, Issue 4, December 2011 (Elsevier).

[32] G. Almpanidis, C. Kotropoulos, I. Pitas, "Combining text and link analysis for focused crawling—An application for vertical search engines" , Information Systems 32 (2007) 886–908 (Elsevier).

[33] Pooja Gupta , Ashok Sharma , J.P. Gupta, "A Novel Framework for Context Based Distributed Focused Crawler (CBDFC)", Int. J. Computer and Communication Technology, Vol. 1, No. 1, 2009.

[34] Hong-Wei Hao , Cui-Xia Mu, Xu-Cheng Yin , Shen Li, Zhi-Bin Wang, "An Improved Topic Relevance Algorithm for Focused Crawling", in IEEE International Conference on Systems, Man, and Cybernetics (SMC) , 2011.

[35] Hai Dong Hussain, F.K. Chang, E., "A survey in semantic web technologies-inspired focused crawlers", IEEE 3rd International Conference on Digital Information Management, 2008.

[36] Knut Magne Risvik, Rolf Michelsen, "Search engines and Web dynamics", International journal Computer Networks 39 (2002) 289–302 (Elsevier).

[37] Sotiris Batsakis, Euripides G.M. Petrakis, Evangelos Milios, "Improving the performance of focused web crawlers", Data & Knowledge Engineering 68 (2009) 1001–1013 (Elsevier).

[38] http://en.wikipedia.org/wiki/.NET_Framework.

[39] Todd A. Letsche , "Large-Scale Information Retrieval with Latent Semantic Indexing" ,

[40] Ritendra Datta , Dhiraj Joshi, Jia Li, James Z. Wang , " Image retrieval: Ideas, influences, and trends of the new age " , ACM Comput. Surv., Vol. 40, No. 2. , May 2008.

[41] http://www.creighton.edu/fileadmin/user/HSL/docs/ref/Searching_-_Recall_Precision.pdf

Developed an Intelligent Knowledge Representation Technique Using Semantic Web Technology

M.Samsuzzaman
Universiti Kebangsaan Malaysia, Malaysia
Email: sobuzcse@eng.ukm.my

M. Rahman[1], M.T Islam[2],T. Rahman[1], S. Kabir[1], R.I. Faruque[2]
[1]Patuakhali Science and Technology University, Bangladesh
[2]Universiti Kebangsaan Malaysia, Malaysia
Email: {hera_cse, esty_cc@yahoo.com, tariqul,rashed@ukm.my, summa.cse@gmail.com}

***Abstract*—Semantic web offers a smarter web service which synchronizes and arranges all the data over the web in a disciplined pattern. In data mining over the web, accuracy of selecting necessary data as user demand and pick them for output counts as a major key challenge from long ago. Our approach contributes a complete and automatic mapping of data over web3. 0 through ontology and accesses them by intelligent web agent. The agent offers all possible output related to user request, from which user could find desired information. When a user has insufficient data parameters to search, they can gain knowledge from the relational outputs provided by the agent and thus semantic web mining enables unknown knowledge acquisition or discovery. Here, in this paper we briefly illustrate and discuss the architecture of semantic web, then propose a model for web mining to discover knowledge under a framework of agent, and finally discusses the ways, how agent finds out user query related nodes from ontology.**

***Index Terms*—Semantic Web, Intelligent Agent, Web Mining, Ontology, Knowledge Representation.**

I. Introduction

Ongoing rapid progress and extensive application of the internet, there is a massive amount of information resources distributed on the web. The conventional string based search often misses extremely relevant pages and feedbacks a lot of irrelevant pages for user request. A common major problem for a user that "Everything is on the web, but we just cannot find what we need [1]" is partially true. Because, most of the data over the web is scattered, unstructured, often inconsistent and insufficient. Data sets are not interlinked with each other which makes mining more difficult. There are huge examples where users get bored because required information was not given on the web or they were lost. Another case is when users have very insufficient data parameters to search. To discover unknown knowledge is almost impossible in web2. 0, because there are no relationships among data sets which makes traditional web mining results almost unsatisfactory. For better performance, people are now faced toward web3. 0 which is an extension of current web2. 0. Here, information is presented on the web in a well-defined and structured manner, enable machines and human to work cooperatively. Data in the semantic web is interlinked among each other through ontology which makes effective discovery, mechanization and assimilation possible. These data has a major key factor that they are machine readable and can be shared and processed by automated tools as well as people. Intelligent agent [2] facility enables users to find desired results for all possible related terms with respect to requirements. Our work has focused on how an agent detects all possible entities from ontology during web mining [3] related to a user query request on its own in an automated manner which enables the user to discover unknown knowledge.

II. Semantic Web Architecture

The semantic web architecture constructed by seven different levels which is organized of a layered architecture [4-5], according to Tim Berners-Lee (the inventor of semantic web). The starting layer URI and Unicode are the base for the structure of the whole system. Unicode provides a unique encoding system for processing resources. It is the universal standard encoding system for computer character representation and recognition. Before Unicode [11] computer character representation, there were several different encoding systems which made a massacre to the combination and communication across machines. Now it's so much easier.

doi:10.4304/jetwi.5.4.407-412

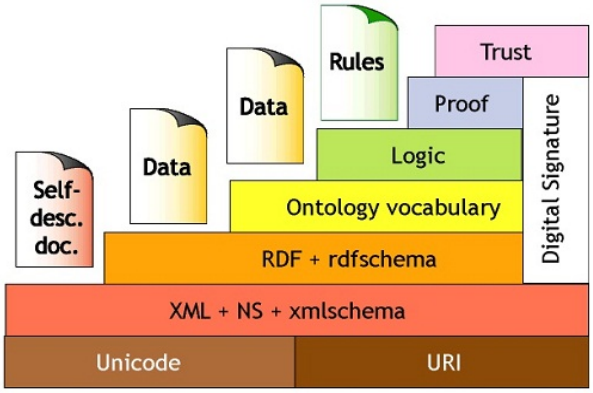

Figure 1. Semantic web architecture [4-5].

URI (Uniform Resource Identifier) provides resource documentation, which allows accurate reclamation of information possible. A resource can be anything that has an identity such as a web site,web page, a document, an image , a audio and vedio file and a person. URLs (Uniform Resource Locator), a widely used type of URIs, are very commonly used in the web, which contains the resource address. The Second layer consists of three parts- XML (Extensible Markup Language), NS (Namespace), XML Schema [8]. This layer represents linguistic data through a standard web format. XML provides a standard way for better sharing of composing information. On the other hand, it improves the freedom to structure. XML Schema pronounces the structure of XML documents. Namespaces [12] provide numerous ways to be eligible the tags and attributes in an XML document with URIs which makes them truly unique on the Web and thus, Universal (among other things). This is really important because every resource on semantic web must be identified uniquely. XML Query is a standardized language for conjoining documents, databases, web pages and almost anything else. It is very widely employed, powerful, and easy to learn. On the other hand, Namespaces [12] permits the combination of different vocabularies. For example, if a document is not marked-up and then each machine may display the documentation its own way which makes document exchange extremely challenging. XML is a mark-up language that tracks certain rules and if all documents are marked-up using XML then there is uniform representation and presentation of used documents. This is one of the significant progresses of the WWW(World Wide Web). The third layer RDF [8] (Resource Description Framework) and RDF Schema offers a semantic model used to describe the information on the Web and type. SPARQL is an RDF query language - it can be used to query any RDF-based data (i.e. including the statements involving RDFS and OWL). Querying language is essential to regain information for semantic web applications. The fourth layer Ontology [8] vocabulary layer is accountable for the definition of shared knowledge and describes the semantic relationships between the different kinds of information to disclose the semantic web between the information itself and information. Ontology is considered the pillar [12, 14] for the semantic web architecture affords a machine-processable semantics and a sharable domain which can facilitate communication between people and different applications. The rule consents proof without full logic machinery. Similar rules are those used by the production systems offered in the corresponding knowledge representation subsection. They imprisonment dynamic knowledge as a set of conditions that must be fulfilled in order to accomplish the set of consequences of the rule. The Semantic Web technology for this layer is the Semantic Web Rule Language (SWRL) The fifth layer logic [8] layer is answerable for providing axioms and inference principles to deliver the basis for intelligent services. The sixth layer proof and the seventh layer trust are liable for providing authentication [9] and trust mechanisms. In the semantic web, each object must be reformed with the corresponding change in the real world. So, for detecting any false change or attacks (may be passive or active), digital signature [9] and encryption [9] techniques are used.

Trust to consequent statements will be supported by (a) verifying that the premises come from trusted sources and by (b) relying on formal logic during deriving new information. Cryptography [10] is significant to ensure and verify that semantic web statements are approaching from a trusted source. This can be accomplished by the appropriate digital signature of RDF statements.

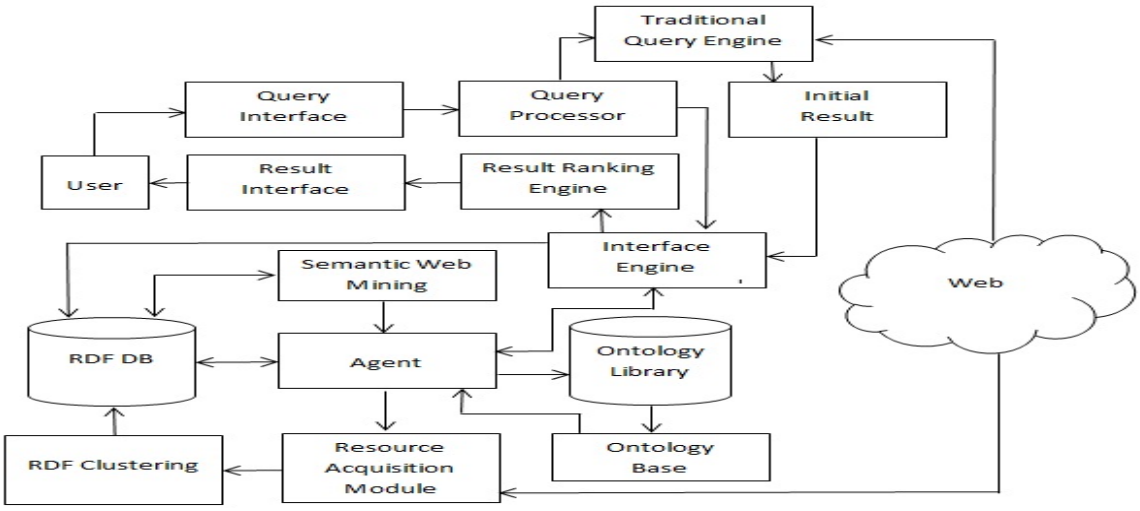

Figure 2. Web mining model under semantic agent framework

III. Agent Based Web Mining

The Semantic Web assurances to change the way of agents navigate, harvest and utilize information on the web. By providing a structured representation for articulating concepts and relationships defined by multiple ontologies, it is now possible for agents to realize users need in a better way and determine knowledge intelligently.

A. What is an Agent?

Agent [8] is an intellectual software being, which is able to complete a particular task instinctively and able for an agent to agent communications under certain state of affairs. Agents are able to perform smart analytical tasks according to the semantic information on the web which develop the accuracy of information retrieval. Rather than doing everything for a user, the agents would find possible ways, to meet user needs and offer the user choices for their achievement. Knowledge discovery from the web instead of data mining is a very intelligent process where semantic agent technology is being used.

B. Proposed Model for Web Mining by Semantic Agent

Though most of the data over the web is unstructured, it is really tough to combine or accumulate them under a common structure (construct a structured web by a night is impossible). So, we combine both traditional web mining model and semantic web mining model facilitated by a semantic agent for superior combination between well-structured semantic network and unstructured real world network situation which are shown in figure 2. The working procedure of this model is described below:

1) *The First Step:* In the beginning, the user query request is being sent to the query processor through a query interface. The query processor is the subcomponent of the data server that processes user requests.

2) *The Second Step:* The query processor calls various traditional query engines and side by side intelligent agent through the interface engine with user request parameters. Interface stop controller enables the user to shut down mining immediately if desired. A query engine [13] is a service that takes a explanation of a search request, evaluates and executes the request, and returns the results back to the caller. This service acts as an in-between layer between clients and the underlying data sources by interpreting search requests and shielding the clients from details on how to access the data sources. Traditional query engines return initial results to interface engine and results are sent to RDF database. RDF database store results in a structured way.

3) *The Third Step:* For agent based searching, an initial ontology should build and to construct this initial ontology various concepts about the objects of the web need to be gathered together. In most of the cases, specialized clustering algorithm [8] is used to gather data from the web. Ontology model merges knowledge of experts [8] in the environment to build an initial ontology. The ontology level will be warehoused in the ontology library system [8] for future levels usage.

4) *The Fourth Step:* When user request parameters are received by the agentfrom thee queryprocessorr through interface engine, agentchecks thee RDFdatabasee. If RDF database contains desired results by caching, agent directly sent results to user through interface engine. On the other hand, agent seeks out all possible relationships between user request and other web entities from ontology library and builds an ontology base with relational entities if desired results are not found in RDF database.

5) *The Fifth Step:* Ontology base contains all possible nodes related to user request collected by the agent and by acquiring knowledge from ontology base; resource acquisition module collects task related information from the web. But during the acquisition of data from the web a crucial problem arises that is arriving of irrelevant information because most of the data over the web is unstructured. The total model performance is mostly dependent on these data acquisition performance.

6) *The Sixth Step:* Resource nodes of the closest characteristics are detected and collected by resource acquisition module [8]. These nodes are being stored in the RDF database [8].

7) *The Seventh Step:* Semantic web mining module [8] mines the data in RDF database for better output and outputs is being sent to agent.

8) *The Eighth Step:* To increase the relevance of result agent performs various filtering process over the outputs of the semantic web mining module.

9) *The Ninth Step:* In this final step, all the relational results will be sent to interface engine from RDF database by the agent. Result ranking engine used for ranking the results and after ranking results will be shown to the user by a result interface. The result is given to user exhibits all possible relational aspects from which user could get desired knowledge may be known or unknown. This process is very efficient when users don't have a sufficient amount of data parameters to find desired output from the web. Instead of data mining, semantic web enables knowledge mining (knowledge acquisition) over the web. This is the main difference between web2. 0 web mining and semantic web mining.

IV. Ontology Access by Semantic Agent

Ontology level contains all conceptual knowledge about the objects in the field and stores them into an

ontology library. When a user calls agent with some data parameters, agent starts to search the ontology to find all possible nodes related to user given parameters. This inquiry becomes possible because all the datasets are interlinked with each other and well defined in the semantic web. The agent gives the user a broad range of ability to choose what exactly he/she requires. Thus, users feel more comfortable to be facilitated by semantic web agent than web2. 0 search engine.

A. A Simple Ontology Building

We can construct any kind of ontology by using *Protégé* [7]. Here, we build a simple ontology named relation. Owl shown in figure 3 which has a class named people and under people some subclasses named Norbert, Jack, John, Irina and Edward.

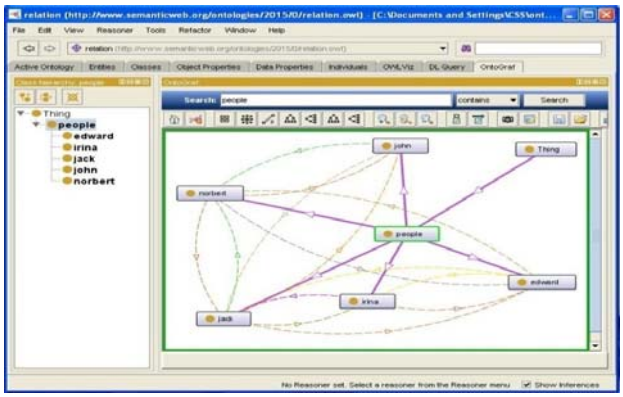

Figure 3. Relationships between entities shown in relation.Owl.

The relationship between these five people is constructed by using various object properties.

B. Agent Query

Ontology relation. The owl can be accessed by two simple queries-

```
// Query for some property restrictions
$query1= 'SELECT ?x ?y ?z ?a  WHERE
(?x, rdfs:subClassOf,    ?y),
(?y, owl:onProperty,     ?z),
(?y, owl:someValuesFrom, ?a)';
// Query for all property restrictions
$query1= 'SELECT ?x ?y ?z ?a  WHERE
(?x, rdfs:subClassOf,    ?y),
(?y, owl:onProperty,     ?z),
(?y, owl:allValuesFrom, ?a)';
```

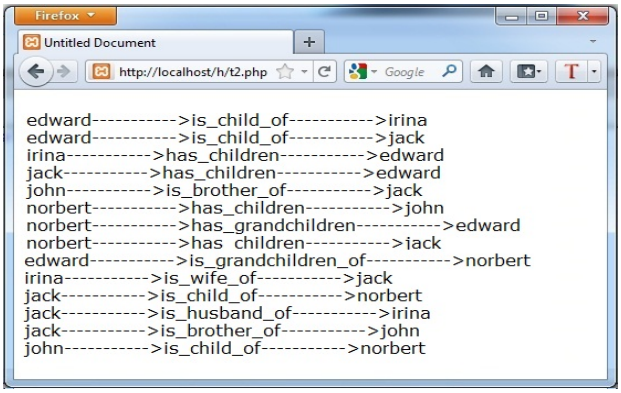

Figure 4. Relationships among all entities accessed by an agent from relation.Owl.

By executing these queries we can figure out all relationships among five people as shown in Fig 4. These relationships have always remained in RDF Triple format (Subject+ Relationship+ Object). Now, two types of user request could be found-

1) *Simple Knowledge Acquisition:* User requires total information about Edward. Relationships with all possible nodes related to Edward could be detected by SPARQL filter query shown in Fig. 6.

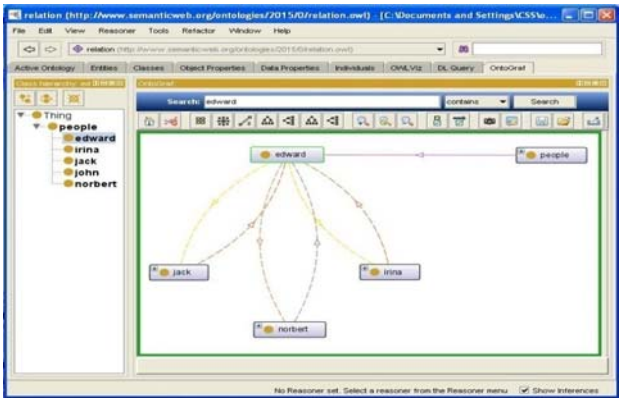

Figure 5. Relationships of Edward with all possible entities in relation.Owl.

Figure 6. Relationships of all possible nodes with Edward accessed by an agent from relation.Owl.

2) *Unknown Knowledge Discovery:* Now, time for the complex query. Let us assume that a user wants to

know about the relationships between Edward and John. From Fig. 5 it is clear that there are no direct relationship between Edward and John. In these complex situations, agent find out the closest node related to both Edward and John. From Fig. 7 we can see that the closest node is Jack. Now, agent exhibits all possible relationships between Edward-Jack and Jack-John. From these relationships user would able find out the relation between Edward and John. Ash, Edward is the son of Jack and Jack is the brother of John according to Fig. 8, then the user could obviously find out that Edward is the nephew of John and inversely John is an uncle of Edward.

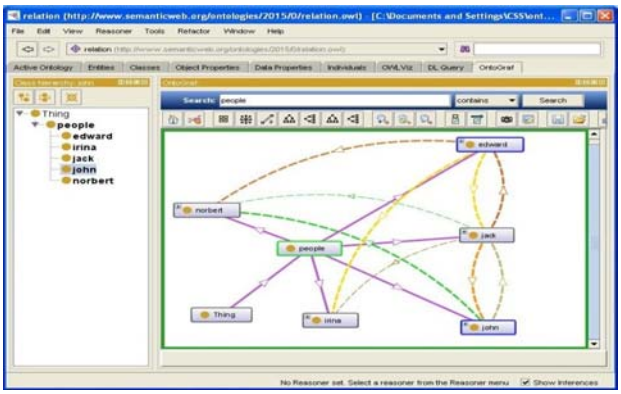

Figure 7. Relationships between Edward and John through jack in relation.Owl.

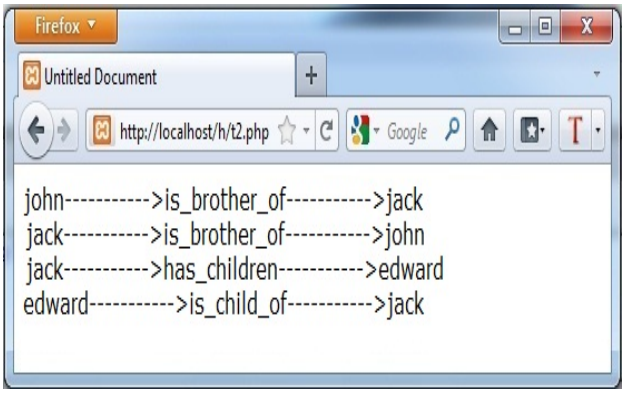

Figure 8. Agent exhibits relationships to enable user knowledge discovery.

V. Conclusion

Successfully destination path discovery by the agent through ontology as per user request provides various facilities such as automation, artificial intelligence integration, machine to machine communication ability etc. By using these facilities, we offer web users knowledge mining instead of data mining. Due to various aspects of limitations, more complex real world dealings as agent to agent communication, synchronization among multiple agents, learning from the related information in environment by agents on its own etc. which is not discussed in this paper which will be the future work.

References

[1]Martin Hepp, "*Semantic Web & Semantic Web Services*".
[2]Stuart J. Russell and Peter Norvig, *"Artificial Intelligence: A Modern Approach"*.
[3]Ajay Chakravarthy, "*Mining the Semantic Web*", 2004.
[4]W3C Semantic Web, http://www.w3.org/2001/sw/
[5]Berners-Lee, Tim; Fischetti, Mark (1999). *Weaving the Web*. HarperSanFrancisco. Chapter 12. ISBN 978-0-06-251587-2.
[6]Zhang Hui, ed, "*Ontology-based Semantic Web Mining Technology*", computer development and applications, 2009, 2.
[7]Matthew Horridge, *"A Practical Guide To Building OWL Ontologies Using Protege 4 and CO-ODE Tools"*, Edition 1.3.
[8]WANG Yong-gui, JIA Zhen, "*Research on Semantic Web Mining*",2010.
[9]William Stalling, "*Wireless Communications and Networks*".
[10]www.wikipedia.com
[11]www.semanticfocus.com
[12]www.unicse.org
[13]www.tim-wellhausen.de
[14]Zhen Zhang, Bin He, Kevin Chen Chuan Chang, "*Understanding Web Query Interfaces: Best Effort Parsing with Hidden Syntax*".

Md. Samsuzzaman was born in Jhenaidah, Bangladesh in 1982. He received the B.Sc. and M. Sc. Degree in Computer Science and Engineering from Islamic University Kushtia, Bangladesh in 2005 and 2007, respectively. Currently, he is pursuing his Ph.D. degree in telecommunication engineering in the Universiti Kebangsaan Malaysia (UKM). From February, 2008 to February, 2011, he worked as a lecturer at Patuakhali Science and Technology University (PSTU), Bangladesh. From February 2011 to till now; he is working as an Assistant Professor at the same university. He has authored or co-authored approximately 14 referred journals and conference papers. He is currently a Graduate Research Assistant at the Institute of Space Science (ANGKASA), UKM, and Malaysia. His research interests include the RF, electromagnetic field and propagation, Antenna Technology, Satellite communication, WSN and Semantic Web.

Md. Mamunur Rahman was born in Khulna, Bangladesh.He received B.Sc. in Computer Science and Engineering from Patuakhali Science & Technology University, Dhumki, Patuakhali, Bangladesh in 2012. His research interests include the Semantic web technology, Algorithm.

Mohammad Tariqul Islam was born in Dhaka, Bangladesh in 1975. He received B.Sc. and M. Sc. Degrees in Applied Physics and Electronics from the University of Dhaka, Dhaka, Bangladesh in 1998 and 2000, respectively and a Ph.D degree in telecommunication engineering from the Universiti Kebangsaan Malaysia (UKM) in 2006. In August 2000, he became an Adjunct Research Fellow at Bose Research Centre, University of Dhaka, Dhaka. He has been very promising as a researcher, with the achievement of several International Gold Medal awards, a Best Invention in Telecommunication award and a Special Award from Vietnam for his research and innovation. He has filled 6 patent applications. He has authored or co-authored 102

international journal papers and 90 international and local conference papers and 3 books. Thus far, his publications have been cited 560 times, and the H-index is 15 (Source: Scopus). He has been awarded "Best Researcher Award" in 2010 and 2011 at UKM. He served as a faculty member at the Multimedia University (MMU), Malaysia from May 2007 until May 2008. He is currently a Professor at the Institute of Space Science (ANGKASA), UKM, Malaysia. His research interests concern enabling technology for RF, antenna technology, electromagnetic absorption and radio astronomy instruments.

Tanjim Rahman was born in Barisal, Bangladesh.He received B.Sc. in Computer Science and Engineering from Patuakhali Science & Technology University, Dhumki, Patuakhali, Bangladesh in 2012. His research interests include the Semantic web technology, Algorithm, Data Mining.

Sumaiya Kabir was born in Barisal, Bangladesh.She received B.Sc. in Computer Science and Engineering from Patuakhali Science & Technology University, Dhumki, Patuakhali, Bangladesh in 2012. Her research interests include the Semantic web technology.

Mohammad Rashed Iqbal Faruque was born in Chittagong, Bangladesh in 1974. He received the B.Sc. and M. Sc. Degree in Physics from University of Chittagong, Chittagong, Bangladesh in 1998 and 1999, respectively, and a Ph.D. degree in telecommunication engineering from the Universiti Kebangsaan Malaysia (UKM) in 2012. From July 2000 to until 2007, he worked as a lecturer at Chittagong University of Engineering and Technology (CUET), Chittagong, From June 2007 to November 2008; he was an Assistant Professor at University of Information Technology and Sciences (UITS), Chittagong. He has authored or co-authored approximately 60 referred journals and conference papers. He is currently a Senior Lecturer at the Institute of Space Science (ANGKASA), UKM, Malaysia. His research interests include the RF, electromagnetic field and propagation, FDTD analysis, electromagnetic radiation, metamaterial applications and electromagnetic compatibility.

Automatic generation of human-like route descriptions: a corpus-driven approach

Rafael Teles[a], Bruno Barroso[a], Adolfo Guimaraes[b], Hendrik Macedo[a]
[a] Computing Department, Federal University of Sergipe, Sergipe, Brazil
Email: rafast.telles@gmail.com, brunokreutz@hotmail.com, hendrik@ufs.br
[b] Computer Science Department, Federal University of Minas Gerais, Minas Gerais, Brazil
Email: adolfopg@dcc.ufmg.br

***Abstract*— Most of Web applications combines differents services, features and contents in order to enable the creation of new features and services. Such systems are called mashups. One of the most popular kind of mashups are the location ones that use geographic data to provide functionalites to users. The RotaCerta is a location system that uses the Google Maps and perform Natural Language Generation to provide textual descriptions of routes between two different locations. The great advantage of RotaCerta is the use of points of interest (POI) to describe routes. POIs help the user to understand and assimilate the route. However, RotaCerta suffers from a several limitation: the need for manualy updating of a POIs dataset. Such work is exhausting, costly and greatly limits their use. Another point to highlight is the poor linguistic variability of texts it provides. In this work, we propose a mechanism to enable automatic feeding of POIs and a corpus-driven approach to enhance the linguistic variability of location mashups such as RotaCerta. We adopt both manual and automatic generation of new textual templates. In order to assess the quality of the routes descriptions, we use TF-IDF and cosine distance to calculate the similarity between descriptions of routes created by human volunteers and descriptions generated by the proposed approach. Route generation examples have been performed for three different brazilian cities. We also show that the text generated from the new template base is more similar to the texts used by people when describing routes if compared to Google Maps.**

***Index Terms*— Mashups, Location Systems, Natural Language Generation, Points of Interests**

I. Introduction

THERE is a wide variety of mobile devices as mobile phones, smartphones, tablets and PDAs supporting alternative communication facilities such as WiFi, 3G and GPS. Many applications make use of these technologies in order to provide new services and features. One of these services are location systems, which are used to determine a user's position on a map and route between two geographic points. Google Maps[1] is one of the most popular location system. However, the routes descriptions are far from the way people use to communicate. By providing information about location of establishments, people often make use of landmarks/reference (Points of Interest - POI) to facilitate the assimilation of the route by the concerned person. The sentence "Go on The Street Y and after passing by the Pharmacy X, turn on the right.", for instance, seems to be more intuitive than something like "Go on The Street northbound for 500 meters and then turn right". Coordinates and cardinal notion of distance are far more difficult for human understanding.

Some approaches in the concerned literature aim to propose the inclusion of POIs on route suggestions in order to improve the description for the user. One of these systems is the GuiaMais[2]. Although it is quite similar to Google Maps in defining routes, it allows for the searching of POIs. The user provides the type of property he/she is looking for, such as pizzerias, bars, restaurants or shopping malls at a specific neighborhood, city or state. The system returns a map of the area with several POIs as requested. A great limitation of GuiaMais is that the POI information is not incorporated into the description of routes.

The system Already Talking Points [1] describes an urban orientation system based on reference points. The authors advocate the idea that a walking route may be better presented with the aid of POIs. In addition to include reference points in the descriptions of routes, the system provides an vocal-enabled interface. This allows its use primarily by the visually impaired. However, the feeding of POIs should be done manually, which requires enourmous effort and limits its usage.

Another system that uses POIs on its route suggestions are proposed by [2]. The author emphasizes the advantages of using POIs in the description of the routes, both to provide confirmation that the user is going in the right direction and also to facilitate the memorization of routes. The system implements an algorithm that weights references according to their characteristics in order to select the points that can be more easily identified by people. However, the basis of POIs only contains data regarding Australia.

RotaCerta [3] incorporates information of POIs on route suggestions. The system has its own built-in POIs and uses techniques from Natural Language Generation (NLG) along with the Google Maps API to describe routes in a way similar to that used by people on a day-to-day. Although its interesting results, the RotaCerta

[1] http://maps.google.com

[2] http://www.guiamais.com.br/

doi:10.4304/jetwi.5.4.413-423

suffers from problems of maintainability and updating the database of POIs. The feeding of POIs database is manually done, which considerably limits system scalability. Another limitation to highlight is the low linguistic variability of the text generated to describe the routes.

This paper extends the work of RotaCerta [3] along three different axis: (i) proposal of an automated mechanism to update the POI database, (ii) improve the NLG module in order to augment the linguistic variability of the set of pre-defined templates that constitute the description of the route and, finally, (iii) perform an assessment of the quality of the generated text. The goal is to ensure that the system can be used to generate more human-like route descriptions to any place of interest. Unlike the original work, our approach considers the integration with the Google Places[3] system, which consists of a collaborative system for adding POIs of several countries, and a new approach to associate POI to generated routes. The approach used to enhance linguistic variability is corpus-driven. A web page was created to allow the collection of route descriptions provided by volunteers. The extraction of new templates has been performed in two different ways: (i) manually and (ii) automatically, by the adaptation of the algorithm to generating paraphrases described in [4] .

The remainder of this paper is organized as follows. In Section II, we describe the background technologies used to the system development: Mashups, Natural Language Generation and Generation of Paraphrase-based data. Section III presents implementation details and highlight the differences to previous system. Performed experiments, results and discussions are shown in section IV. Finally, in Section V we conclude the work and discuss some possible extensions and future investigations.

II. Technological background

A. Mashups

The term Web 2.0 is commonly associated with Web applications that facilitate the sharing of information interactively such as forums, Web sites, social networks, blogs and mashups [5]. Although the concept of Web 2.0 suggest a new version of the World Wide Web, this does not refer to any change concerning technical specifications, but rather in the way that developers and end-users use the Web [6].

Originally, the term mashup was used to describe mixing or blending of two or more music tracks, commonly used by DJ's. In the context of Web, mashups are simply a new way of developing applications by combining services, features and content already available. Such elements can be formatted as RSS feeds, XML and its derivatives, HTML, Flash or any other type of graphics.

The mapping mashups are one of the most popular forms of mashups. Its main feature is the use of maps, usually used to determine routes between two points. Based on an initial geographical reference (longitude and latitude), these systems are able to describe a route to a target location. The result of this application is typically a graphical representation of the path and textual description. Currently, 81.82% of mapping mashups make use of the Google Maps API [7]. The availability of such an API [8] can be seen as a major factor for such indices [3].

Among the services provided by Google Maps API, can be highlighted: (1) the creation of paths based on source and destination addresses using the Google Directions, (2) calculate the distance between two points, (3) information of latitude and longitude, (4) coordinates of a given address through the Reverse Geocode, and (5) detailed information of a specific point such as street name, number, city, neighborhood, zip code and country.

B. Natural Language Generation

Natural Language Generation (NLG) is a field of Artificial Intelligence that addresses computational systems able to produce understandable texts in a particular human language, starting from non-linguistic data. NLG systems use knowledge about language and the application domain to automatically produce documents, reports, and other [9].

Figure 1 gives an example of a system for generating natural language descriptions of weather events from daily meteorological records [10]. Note that from daily records of temperature and rainfall it is possible to produce textual summarization of climatic events occurred in the month.

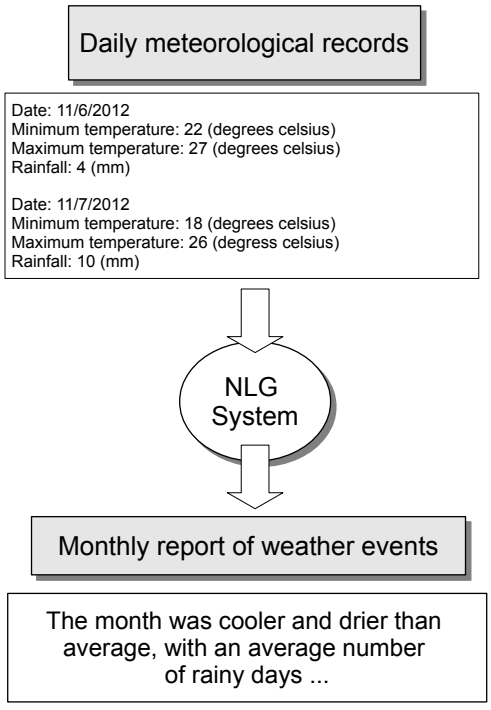

Figure 1. NLG system example

The final text of an NLG pipeline should meet the following features: (1) be linguistically correct, (2) clearly convey the information input, (3) respond to the proposal

[3]http://code.google.com/apis/maps/documentation/places/

for which it is being generated, and (4) appear fluent enough to avoid a mechanical communication [11]. Another aspect that can be taken into account is the generated text format. This can be raw text or include more sophisticated formatting elements to meet specific interface requirements, HTML, VXML, LaTeX, and others.

NLG systems is usually classified into 03 (three) categories (in decreasing order of complexity): (1) use of preprogrammed responses (canned text), (2) use of standardized sentences (template) and (3) phrase-based text generation. Each of these differ by their complexity and flexibility of results [11].

The basic idea behind canned text is to associate to each possible systems response at a given context, an answer in natural language. This method is satisfactory when it provides a small number of answers. As the number of system settings increases, this technique tends to become impracticable due to the huge amount of combinatory results [3].

NLG systems based on templates map the input data to a linguistic structure [12]. This linguistic structure may contain gaps or slots. The output result is obtained when all slots are filled or replaced by the language structures that do not have slots [13]. Suppose we want a system based on NLG templates be able to generate sentences to a train station. The description of such system can be seen in [12] and can start from a semantic representation informing that the train 306 leaves Aberdeen at 10:00 AM:

$$\text{Departure}(train_{306}, location_{abdn}, time_{1000}),$$

which is directly associated with a template such as:

[train] *is leaving* [town] *now*

The slots represented by [train] and [town] are filled by relevant information obtained from a table. This template is used only when the time said (10:00 AM) is close to the announcement message. Other templates should be used to create ads for the past or future.

Phrase-based text generation use an indirect mapping between input data and linguistic form. Such systems take as input a semantic representation of data. This is then subject of successive transformations until a linguistic structure is produced. Various system components can operate on this NLG defining, for instance, that 10:00 AM is the estimated time for the ad to be displayed and thereby transforming the input into an intermediate representation such as follows:

$$Leave_{present}(train_{demonstrative}, Aberdeen, now)$$

It is possible to note that the lexical items were determined when the linguistic morphology was still missing. This intermediate representation, in turn, can be transformed into an appropriate sentence such as:

This train is leaving Aberdeen now

The details of the sentence may vary. Such systems may contain several intermediate representations.

C. Paraphrase generation based on data

The paraphrase is usually considered a way for preserving meaning. If we are dealing with a essentially linguistic textual form, then we can say that the text A and B are paraphrases of each other, if both texts A and B have the same meaning [14]. In other words, the paraphrase is an alternative way of rewriting a text in the same language, without losing the semantic content of the original text.

Paraphrases may exist at the level of words, also known as lexical paraphrase, in which the most common are synonyms, for example, (car, automobile) and (dog, puppy). The hypernym is another example of lexical paraphrase. In this case, a word is more general or specific than the other, as in the examples (shoes, boots) and (fruit, orange).

Paraphrasing is also present at the level of sentences, as in the examples (I finished my work, I finished my task). At this level, it is possible to generate paraphrases by simply replacing one or more words in the original phrase by others with the same meaning.

The concept of distributional similarity, extremely popular technique used in the generation of paraphrase, states that words or phrases that share the same distribution - the same set of words in the same context in a set of texts written in a language that serves as a database for linguistic research (corpus) - tend to have similar meanings [15]. A commonly used model for calculating the similarity is the N-gram model.

The N-gram model aims to compute the probability of finding a word W given a history H, or P (W | H). Such assumption that the probability of a word depends only on the previous word is called Markov assumption [16]. Markov models are a class of probabilistic models that assume that we can calculate the probability of any future unity using only the near past. We can generalize the Bigram (which takes into account only one word history) for Trigram (which takes into account two word history) and thus for the N-gram (which takes into account N - 1 words of the history).

As an example of an N-gram model, suppose that a history H is the sentence "Please turn off your mobile" and we want to compute the probability that the next word is *phone*:

P*(phone | Please turn off your mobile).*

The number of parameters needed to calculate this probability increases exponentially with the number of words in the history H. For the above example we have the following N-gram models:

- *Unigram*: P*(phone)*
- *Bigram*: P*(phone | mobile)*
- *Trigram*: P*(phone | your mobile)*

The algorithm of Pasca and Dienes [17] uses as input corpus a vast collection of Web documents retrieved from Google search engine. First, all n-grams of a specific type for each sentence are computed. Next, a set of anchors (distribution) of each sentence is created and the tuple (Anchors, phrase) is then stored in a list. A count of how many anchors are shared for each pair of

sentences is also done. As in the case of distributional similarity, the greater the number of anchor phrases that are common to two candidates, the greater the likelihood that they paraphrases each other. Finally, the resulting list of paraphrases undergoes a filter in order to eliminate those that are less likely to be similar.

Lin and Pantel [4] discuss how to measure the distributional similarity through paths in dependency trees to induce generalized paraphrases templates as:

$$X \textit{ found an answer to } Y \Leftrightarrow X \textit{ solves } Y$$

A dependency relationship is an asymmetric binary relation between a word called "head" and another word called "modifier". The structure of a sentence can be represented by a set of dependency relationships in a tree structure. A word in this phrase may have various modifiers but, often, each word is the modifier of just another word. The root of the dependency tree does not modify any other word. Figure 2 shows the dependency tree generated for the sentence "Mary found an answer to the problem". Arrows represent a dependency relationship between the head and the modifier whereas the name above them indicates the type of relationship found [4].

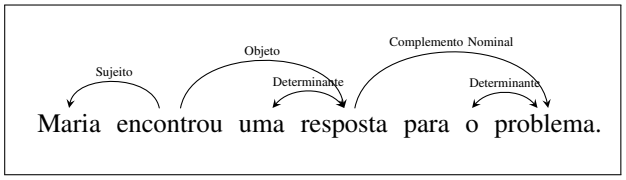

Figure 2. Example of a Dependency Tree

In a dependency tree there are two types of semantic relationship: (i) the direct and (ii) indirect. A direct relationship is represented by the arrows in the figure above (links) and the indirect relationship is represented by a path that connects two words. This path is defined by concatenating dependency relationships with the words found throughout these, excluding the ends. For the sentence of figure 2, the path between words "Mary" and problem" is represented as follows:

N:sujeito:V$\leftarrow$ encontrou $\rightarrow$ V:objeto:N $\rightarrow$ resposta $\rightarrow$ N:para:N,

which means "X found an answer to Y". The words at the extremities of the phrase are called *slots* that are filled with *path context*. For example, the slots X and Y of the template extracted from the path above could be filled with the words "Mary" and "problem", respectively.

Hypothetically, if two paths connecting the same set of words they tend to be similar. Lin and Pantel [4] extend the distributional hypothesis: if two paths tend to occur in similar contexts, thus their meanings tend to be similar.

In order to compute the similarity of the extended distributional hypothesis, it is necessary to compute the frequencies of all paths within the *corpus* and the words that fill their slots. For each instance of a path P which connects two words $W1$ and $W2$, the frequency counter is incremented for both triples $(P, SlotX, W1)$ and $(P, SlotY, W2)$, where $(SlotX, W1)$ and $(SlotY, W2)$ are called *characteristics* of P. The more characteristics two paths divide, more similar they are. It uses a hash table to accumulate the frequency of all features for the whole set of paths extracted from the *corpus*. Essentially, two paths are similar if there is a large number of in common features. [4].

The relationship of mutual information between a path slot and a word that fills it can be computed by:

$$mi(p, Slot, w) = \log\left(\frac{|p, Slot, w| \times |*, Slot, *|}{|p, Slot, *| \times |*, Slot, w|}\right) \quad (1)$$

The similarity between a pair of slots (s_1, s_2) is defined as:

$$sim(s_1, s_2) = \frac{\sum_{w \in T(p_1,s) \cap_T (p_2,s)} X + Y}{\sum_{w \in T(p_1,s)} X + \sum_{w \in T(p_2,s)} Y} \quad (2)$$

where $X = mi(p_1, s, w)$ and $Y = mi(p_2, s, w)$. The similarity between a pair of paths is defined as the geometric mean of the similarities of the slots, such as:

$$S(p_1, p_2) = \sqrt{sim(SlotX_1, SlotX_2) \times sim(SlotY_1, SlotY_2)} \quad (3)$$

The performance of such algorithm is strongly tied to the type of the word that is associated with the root of the extracted path. For example, whereas the verbs tend to have various modifiers, nouns usually do not have more than one. However, if a word has less than two modifiers, it cannot be root for a path. As a consequence, this algorithm tends to perform better for paths whose regards verbs.

III. The RotaFacil system

RotaFacil is an extension of a location mashup system called *RotaCerta* [3]. The main goal is to automate the updating mechanism of POIs database and improve linguistic variability of templates. It is proposed the use of a collaborative basis of POIs and a corpus-driven approach for template generation to achieve such goals. These tasks are described in the following sections along with a detailed overview of system's architecture.

A. System's architecture

The system architecture consists of three modules: (1) Route Generator (GRotas), (2) Reference Generator (GRef) and (3) Text Generator (GText).

The GRotas aims to find a path that connects the origin and destination points and select points along the path so that a pre established minimum distance *md* is preserved. This module uses Google Directions API[4].

The GRef has the following functions: (1) obtain reference points for each route point found by GRotas, (2) find addresses for these points, (3) associate the points to

[4]http://code.google.com/apis/maps/documentation/directions/

the route points, (4) eliminate the useless references and (5) generate an XML file containing all this information.

The GText is responsible for building linguistic textual description for the suggested route. This module consists of two components, namely: (1) the *natural text generator* and (2) the *templates dataset*.

Communication among these modules is done by means of XML files that store information of the route and reference points.

Next, we describe the extension of the modules GRef and GText. In the first, RotaCerta's original POI base has been replaced by the collaborative base *Google Places*. In the second, an approach based on the paraphrases generation have been applied in order to improve the linguistic variability of the generated routes.

B. Automatic feeding of POI database

The GRef is responsible for reviewing the generated route by GRotas and associate to each route point a set of POIs extracted from Google Places.

However, using a collaborative approach leads us to an important issue: how to select the appropriate POI? We have adopted a set of constraints in order to determine if such a POI is useful or not to the route description. The first constraint consists of selecting the references according to a distance X to the point in question. This procedure eliminates remote routes that do not provide an information gain for the textual description. In our experiments, we have empirically defined $X = 50(meters)$. Another important constraint is that only commercial or residencial establishments are valid POIs; neighborhood streets, avenues and parks are not considered. Finally, references that are not at the same address of both points to which they have been attached are also desregarded.

C. Enhancing linguistic variability of templates

A Web application has been developed so to provide a pre-defined set of six different visual routes of Aracaju/SE city, Brazil. Volunteers were then asked to provide a textual description for each route, as if they were guiding someone else. These descriptions constitute the *corpus* of routes used in the creation of the templates. The set of POIs provided by the GRef module was available, so the volunteer could use it to enrich the textual description.

61 route descriptions have been obtained. 36 descriptions have been selected to compose the generation set. The remaining 25 descriptions have been used to evaluate the quality of route descriptions generated by the system. We have defined two different approaches to create the system's set of templates: (1) manual definition and (2) automated generation of templates.

It is worth point out that RotaFacil works with templates and routes in Brazilian Portuguese language. Thus, examples of route descriptions and templates presented in this paper is in Portuguese language. In some sections we provide the English version of the example in order to favor reading and understanding facilities.

Manual extraction of templates

The first method used to identify templates for route descriptions is to manually identify patterns of textual descriptions in the corpus. A pre-processing stage is performed in order to solve eventual grammatical issues. Next, we perform the marking process of the texts, highlighting core data such as street names, avenues and reference points. The purpose of marking is to facilitate recognition of description excerpts that could be used as a template. Figure 3 illustrates such manual process of extracting templates from a route description in Portuguese language. The name of avenues and streets, the points of interests and templates are highlighted.

O senhor vai seguir até o fim dessa rua, que é a *R. Frei Paulo*, chegando na *R. Rafael de Aguiar*, você irá pegar a esquerda e seguir até a **D&E Locação de Veículos**, entrando à esquerda na *Av. Dr. Edésio Vieira de Melo*. Você passará pela **AC Assessoria Contábil**, depois dela são mais 8 ruas até chegar à *Av. Hermes Fontes*. Cruza a avenida, passando pela **Eletrônica Xique**, e irá entrar na primeira rua à direta, que é a *R. Zaqueu Brandão*. Em seguida, entra na primeira rua à esquerda, segue pela *Av. Augusto Maynard*, e quando você passar pelo **O Blazer**, entra na próxima rua à direita. Entra na *Av. Gonçalo Rolemberg Leite*, passa pelo **Centro de Ortodontia Dr. Paulo Soares** e quando chegar no **Bobs Aracaju**, vira à esquerda na *Av. Anízio Azevedo*, passa pela **Fisioline Serv Fisioterapêuticos** e vai entrar na rua da **Feeling**, que é a *R. Homero Oliveira*. A casa dela vai estar na próxima rua após a **Seccaf**, que é a *R. Cristóvão de Barros*.

Label:

Street and Avenues Names
Points of Interest
Templates

Figure 3. Manual extraction of templates uses highlight marks.

It is possible to note that names of streets, avenues and reference points have been replaced by the two specific slots **[STREET]** and **[POI]**. These slots are essential to NLG component of the system. Table I shows the transformation of templates identified from the corpus and those used by the RotaFacil system.

For clarity, Table II shows the English version of the templates for the considered example.

Automatic extraction of templates

For automatic extraction of templates, we have proposed an adaptation of the algorithm of Lin and Pantel [4], for generation of paraphrases. The adapted version of the algorithm is described below.

The algorithm takes as input the dependency trees generated from the corpus of route descriptions.

TABLE I.
THE RESULT FOR THE MANUAL GENERATION OF TEMPLATES
(EXAMPLES IN BRAZILIAN PORTUGUESE LANGUAGE)

Potential Template	New Template
Pegar esquerda e seguir até a **D&E Locação de Veículos**	Pegue à esquerda e siga até o **[POI]**
Quando você passar pelo **O Blazer**, entra na próxima rua à direita	Quando você passar pelo **[POI]**, entre na próxima **[STREET]** à direita
Quando chegar no **Bobs Aracaju**, vira à esquerda na **Av. Anízio Azevedo**	Quando chegar no **[POI]**, vire à esquerda na **[STREET]**
A casa dela vai estar na próxima rua após a **Seccaf**, que é a **R. Cristóvão de Barros**	O seu destino estará na próxima rua após **[POI]**, que é a **[STREET]**
Entrar na rua da **Feeling**, que é a **R. Homero Oliveira**	Entre na rua da **[POI]**, que é a **[STREET]**

TABLE II.
THE RESULT FOR THE MANUAL GENERATION OF TEMPLATES
(EXAMPLES IN ENGLISH)

Potential Template	New Template
Pick up left and follows the **D & E Car Rental**	Pick up left and follows the **[POI]**
When you go through **O Blazer**, enter the next street on the right	When you go through **[POI]**, enter the next **[STREET]** on the right
When you arrive at **Bob's Aracaju**, turn left on **Anízio Azevedo Avenue**	When you arrive at **[POI]**, turn left on **[STREET]**
Her house will be on the next street after **Seccaf**, which is **Cristóvão de Barros Street**	Her house will be on the next street after **[POI]**, which is **[STREET]**
Enter the street where it is located **Feeling Shop**, which is **Romero Oliveira Street**	Enter the street where it is located **[POI]**, which is **[STREET]**

Algorithm 1: Lin and Pantel Algorithm Adapted

Data: List of dependency trees for the corpus (ArvDep)
Result: List of similarities between the paths found in each dependency tree
foreach $tree \in ArvDep$ **do**
  $Paths \leftarrow$ FindPath($tree$)
  Remove of $Paths$ all path with less than 3 words
end foreach
foreach $p_i \in Paths$ **do**
  $Sim \leftarrow$ SimilarityCalculus(p_i, p_{i+1})
  if $Sim > 70\%$ **then**
    $Similarity_{i,i+1} \leftarrow Sim$
    Adiciona $Similarity_{i,i+1}$ na $SimilarityList$
  end if
end foreach
Return $SimilarityList$

Given the dependency trees as input, all the paths (indirect semantic relationships) with less than three words are removed. Next, we calculate the similarity between all remaining paths by means of Lin and Pantel methodology. Finally, for best results, just the paths with similarity values greater than 70% are considered. For results with more than three words *SlotX* has been removed.

In order to illustrate the application of the algorithm, let's consider the description of the route presented in Figure 4. The first step is to extract the dependency tree of this description. For this, we use the tool proposed in [18].

Vire a primeira à esquerda, siga até a Avenida Dr. Edésio Vieira de Melo. Lá, você deve virar para a esquerda novamente. Siga reto e pegue a primeira à direita depois da Av. Hermes Fontes. Logo em seguida, pegue a primeira à esquerda. Quando chegar na Av. Gonçalo Rolemberg Leite, vire à direita e siga reto até a Av. Anízio de Azevedo. Nessa, vire à esquerda e siga até a Rua Homero de Oliveira, que é primeira depois da Av. Acrisio Cruz. Ao final do segundo quarteirão você chegará ao seu destino.

Figure 4. Route description from *corpus*

The dependency tree is thus provided as input to the algorithm described above. Finally, a set of templates is automatically generated as shown in Figure 5. Again, we present the templates in both Brazilian Portuguese and English for clarity.

Siga até o → Siga até o [POI]
Fallow up → Fallow up [POI]

Vire depois de → Vire depois do [POI]
Turn after → Turn after [POI]

Siga até passando → Siga até [STREET] passando [POI]
Follow up through → Follow up [STREET] through [POI]

Figure 5. Automatically extracted set of templates, using the adapted algorithm.

IV. EXPERIMENTS AND RESULTS

Experiments have been divided into two parts. The first part shows how the automatic POI base updating mechanism of RotaFacil solves the huge limitation of RotaCerta's, thus enabling the utilization of POIs for whatever intended Brazilian city in the generation of route descriptions. Different paths for three different cities in Brazil have been generated: Aracaju/SE, Belo Horizonte/MG and Porto Alegre/RS. The second part of experiments aims to evaluate the quality of the text generated by RotaFacil. At this stage we use the similarity of texts based on TF/IDF to compare the route descriptions generated by RotaFacil to the set of route descriptions provided by volunteers and stored as test set. The results of these experiments are presented below.

A. The POI collaborative dataset

City 1: Aracaju/SE

Figure 6 illustrates a route generated in the city of Aracaju/SE. The route includes the ordered streets **Lagarto** and **Lorival Chagas**. The landmarks between the points A and B represent the POIs selected for this route.

Figure 6. Screenshot of RotaFacil system for the route in Aracaju/SE

Figures 7 and 8 show the text generated by Google Maps and the RotaFacil, respectively, for the route of Figure 6. Names in italics represent the descriptions of streets and avenues, and the names in bold represent the POIs selected by the system for the route in question.

Google Maps route description
(Aracaju/SE)

1. Siga na direção sul na *R. Lagarto* em direção à *Av. Barão de Maruim*
2. Vire à direita na *Av. Pedro Paes Azevedo*
3. Continue para *Av. Mariquinha Seixas Dorea*
4. Faça um retorno
5. Pegue a primeira à direita em *R. Lourival Chagas*

Figure 7. Google Maps route description for Aracaju/SE

Note that the textual description provided by RotaFacil (Figure 8) provides a significant amount of references present in the path that allow the user to evaluate the correctness of his/her displacement at every moment.

For example, if we analyze the description number 13 of RotaFacil: "Arrive at destination that is close to **Marlange Hairdressers** in *Lourival Chagas Street*" (translated into English), it is more informative than the corresponding description in google maps (description number 5: Take the first right in *Lorival Chagas Street*").

RotaFacil route description
(Aracaju/SE)

1. Siga na direção sul na *R. Lagarto* em direção à *Av. Barão de Maruim*
2. Passe por **Cardio Imagem** na *R. Lagarto*
3. Passe por **Odonto Consultórios Odontológicos** na *R. Campos*
4. Passe por **INTERDATA SOLUES EM AUTOMAO** na *Av. Augusta Maynard*
5. Vire na *R. Const. João Alves* próximo a **AS Cosméticos**
6. Vire à direita na *Av. Pedro Paes Azevedo*
7. Continue pela *Av. Pedro Paes Azevedo* passando por **UPSIDE COMUNICAÇÃO E GRÁFICA LTDA**
8. Continue pela *Av. Pedro Paes Azevedo* passando por **Pronto Socorro Espiritual Bezerra de Menezes**
9. Siga pela *Av. Mariquinha Seixas Dorea*
10. Continue pela *Av. Mariquinha Seixas Dorea* passando por **EV Projetos e Consultoria LTDA**
11. Continue pela *Av. Mariquinha Seixas Dorea* passando por **SELMA ATELIÊ DE COSTURA**
12. Passe por**Marlange Cabeleireiros** na *R. Lourival Chagas*
13. Chegue no destino que fica próximo à **Marlange Cabeleireiros** na *R. Lourival Chagas*

Figure 8. RotaFacil route description for Aracaju/SE

City 2: Belo Horizonte

Figure 9 illustrates a route in the city of Belo Horizonte/MG. The chosen path is located between Falcatas Street and Federal University of Minas Gerais, at Antonio Carlos Avenue.

Figure 9. Screenshot of RotaFacil system for the route in Belo Horizonte/MG

Again, we present both textual descriptions. In Figure 10, the text generated by Google Maps and Figure 11,

the text generated by RotaFacil with POI information highlighted in bold. Both textual descriptions are for the route shown in Figure 9.

Google Maps Route Description
(Belo Horizonte/MG)

1. Siga na direção leste na *Alameda das Falcatas* em direção à *Alameda dos Coqueiros*
2. Vire à direita na *Alameda das Latânias*
3. Pegue a primeira à esquerda para pegar a *Avenida Coronel José Dias Bicalho*
4. Pegue a primeira à direita em *Avenida Presidente Antônio Carlos*

Figure 10. Google Maps route description for Belo Horizonte/MG

RotaFacil Route Description
(Belo Horizonte/MG)

1. Siga em direção do **New Commerce TI Ltda** na *Alameda das Falcatas*
2. Vire à direita na *Alameda das Latânias*
3. Passe por **Pierrot Fantasies** na *Avenida Coronel José Dias Bicalho*
4. Continue pela *Avenida Coronel José Dias Bicalho* passando por **Comercial Pandoro Ltda**
5. Passe por **Colégio Brasileiro de Medicina Estática** na *Rua Leopoldino dos Passos*
6. Chegue no destino que fica próximo a **CPEJr - Consultoria e Projetos Elétricos Júnior** na *Avenida Presidente Antônio Carlos*

Figure 11. RotaFacil route description for Belo Horizonte/MG

This example clearly shows how the RotaFacil can assist in reading the description of the route. If we look at the description number 1, the text provided by Google Maps (in english: "Go east on Alameda das Falcatas Street toward the Alameda dos Conqueiros Street"), we notice the allusion to cardinal points (*east*) in order to guide the user. Unfortunately, identifying the east in an urban environment, it is not always a simple task. The RotaFacil provide, for the same description, information from reference point in order to facilitate the user's task. Translating the description RotaFacil 1 for English, we have: "Head towards the **New Commerce IT Ltd** in **Alameda das Falcatas Street**", where "New Commerce IT Ltd" is a known commercial establishment, used as POI.

City 3: Porto Alegre/RS

Figure 12 illustrates a route in the city of Porto Alegre/RS. The chosen path is located between two important known city locations: the Estadio Olimpico de Futebol and the Estadio de Futebol Beira Rio.

Figure 12. Screenshot of RotaFacil system for the route in Porto Alegre/RS

The text generated by the query to Google Maps is shown in Figure 13 and the text generated by RotaFacil is described in Figure 14:

Google Maps Route Description
(Porto Alegre/RS)

1. Siga na direção noroeste na *Avenida Coronel Gastão Haslocher Mazeron* em direção à *Rua Catão Coelho*
2. Vire à esquerda na *Rua José de Alencar*
3. Vire à direita na *Avenida Borges de Medeiros*
4. Faça um retorno na *Avenida Praia de Belas*
5. Continue para *Avenida Padre Cacique*. O destino estará à direita.

Figure 13. Google Maps route description for Porto Alegre/RS

The chosen example concerns a route with much more guiding details.

Google Maps description only considers the names of the avenues. However, the avenues have several points of reference that help the user to verify he/she following the right path. For example, the José de Alencar Street referenced by Google Maps in the description of number 2 (in english: "Turn left at *José de Alencar Street*") is described in RotaFacil with a set of reference points such as the description of number 13 (in english: "Turn on *José de Alencar Street* which is near of **SIDI medical clinic**").

B. Assessing the quality of generated text

In this section, we present the results of the evaluation of the quality of texts generated by RotaFacil using both template generation approaches: (i) automatic extraction (RotaFacil AT) and (ii) manual creation (RotaFacil MA). By hypothesis, it is assumed that the RotaFacil MA produces texts that are more similar to the corpus if compared to RotaFacil AT system which is, in turn, assumed to have greater similarity to Google Maps. In other words, the hypothesis is that the generation of routes in natural language using manually built templates is the closest to the way people usually guide other people to move between different points of a city.

RotaFacil Route Description
(Porto Alegre/RS)

1. Siga em direção do **Mini Mercado Sto. Antonio** na *Avenida Dr. Carlos Barbosa*
2. Continue pela *Avenida Dr. Carlos Barbosa* passando por **Mini Mercado Sto. Antonio**
3. Continue pela *Avenida Dr. Carlos Barbosa* passando por **Massolin de Fiori Societa Taliana**
4. Continue pela *Avenida Dr. Carlos Barbosa* passando por **FGTAS-Fundação Gaúcha do Trabalho e Ação Social**
5. Siga pela *Avenida Cascatinha - Medianeira*
6. Passe por **Alemão Lanches** na *Avenida Cascatinha*
7. Vire à esquerda na *Rua José de Alencar*
8. Siga pela *Rua José de Alencar*
9. Continue pela *Rua José de Alencar* passando por **José Ernesto Azzolin Pasquotto**
10. Siga pela *Rua Gonçalves Dias - Menino Deus*
11. Passe por **Berçário e Escola de Educação Infantil Beija-flor** na *Rua Dr. Oscar Bittencourt*
12. Passe por **Farol** na *Rua Grão Pará - Menino Deus*
13. Vire na *Rua José de Alencar* próximo a **SIDI-Serviço de Investigação Diagnóstica**
14. Vire à direita na *Avenida Borges de Medeiros*
15. Siga pela *Avenida Borges de Medeiros*
16. Siga pela *Viaduto Pedro - Praia de Belas*
17. Siga pela *Avenida Borges de Medeiros*
18. Siga pela *Avenida Padre Cacique*
19. Continue pela *Avenida Padre Cacique* passando por **Armazém do Sabor**
20. Chegue no destino que fica próximo a **Zé Pneus** na *Avenida Padre Cacique*

Figure 14. RotaFacil route description for Porto Alegre/RS

The quality measurement of the RotaFacil system has been made by comparing the generated descriptions to the texts of the test set. This is done by computing the similarity between texts. For that, we have used the statistical measure Term Frequency - Inverse Document Fequency (TF-IDF). TF-IDF is a value that represents how a word is relevant for such a document in regards to a collection (corpus). This importance increases proportionally with the number of times the word appears within the document and decreases according to the frequency of the word throughout the collection [19].

TF (term frequency) corresponds to the number of times the term occurs in the document normalized according to the document size and is calculated by:

$$tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}} \tag{4}$$

where $n_{i,j}$ is the number of occurrences of the term *i* in the document *j* and the denominator is the number of occurrences of all terms in the document *j*.

The IDF (Inverse Document Frequency) evaluates the importance of the term in the collection and is given by:

$$idf_i = \log \frac{|D|}{|\{d_j : t_i \in d_j\}|} \tag{5}$$

where $|D|$ is the total number of documents in the corpus and $|\{d_j : t_i \in d_j\}|$ is the number of documents where the word t_i appears. The TF-IDF is calculated by multiplying the equations 4 and 5, such that:

$$tfidf_{i,j} = tf_{i,j} * idf_i \tag{6}$$

Each term within a document is associated to a weight given by the value *TF-IDF*, computed for the whole corpus. Then, each document (d) is represented by a vector ($\overrightarrow{v}$) of real numbers concerning the weights *TF-IDF* of its terms. Given two documents d_1 and d_2, the similarity between them is given by:

$$simCos(d_1, d_2) = \frac{\overrightarrow{v}(d_1).\overrightarrow{v}(d_2)}{|\overrightarrow{v}(d_1)||\overrightarrow{v}(d_2)|} \tag{7}$$

where $\overrightarrow{v}(d_1)$ e $\overrightarrow{v}(d_2)$ are the term vectors of documents d_1 and d_2, respectively.

We have provided 6 route descriptions using the points of origin and destination described in Table III. These routes were the same used for the generation of the corpus. Since each route has more than one description within the corpus, the texts of RotaFacil have been compared with each of these descriptions.

TABLE III.
TYPES OF MAP DESCRIPTIONS

Route	Source Address	Destination Address
A	Rua Geru	Rua Aquidabã
B	Rua Ananias de Azevedo	Rua Ns. Sra. das Dores
C	Rua Frei Paulo	Rua Cristóvão de Barros
D	Rua Lúcio Mota	Rua Moacir Wandeley
E	Rua Manoel Eculides de Oliveira	Rua João Vitor de Matos

The test set consists of 6 descriptions of type A, 5 descriptions of type B, 6 descriptions of type C, 6 descriptions of type D and 2 descriptions of type E, totaling 25 route descriptions. Obviously, route descriptions have been compared within its particular type (same source and destination addresses).

Tables IV, V, and VI show the results of similarity to Google Maps, RotaFacil MA and RotaFacil AT, respectively, when compared with the descriptions of test set.

TABLE IV.
SIMIALRITY BETWEEN GOOGLE MAPS AND THE TEST SET

Route	1	2	3	4	5	6	Average
A	0,01	0,008	0,03	0,02	0,02	0,02	0,02
B	0,10	0,02	0,05	0,01	0,05	-	0,04
C	0,03	0,08	0,11	0,16	0,12	0,01	0,08
D	0,07	0,16	0,09	0,10	0,03	0,12	0,09
E	0,01	0,03	-	-	-	-	0,02

Table VII shows the comparative result between RotaFacil MA, the RotaFacil AT and Google Maps. As seen earlier, each route type has different amounts of

TABLE V.
SIMIALRITY BETWEEN ROTAFACIL MA AND THE TEST SET

Route	1	2	3	4	5	6	Average
A	0,05	0,13	0,15	0,08	0,03	0,17	0,10
B	0,18	0,08	0,07	0,02	0,20	-	0,11
C	0,08	0,07	0,08	0,10	0,12	0,06	0,08
D	0,09	0,17	0,11	0,14	0,12	0,15	0,13
E	0,02	0,05	-	-	-	-	0,03

TABLE VI.
SIMIALRITY BETWEEN ROTAFACIL AT AND THE TEST SET

Route	1	2	3	4	5	6	Average
A	0,04	0,07	0,05	0,10	0,01	0,06	0,06
B	0,14	0,03	0,10	0,03	0,20	-	0,10
C	0,03	0,04	0,14	0,13	0,13	0,10	0,10
D	0,12	0,19	0,15	0,16	0,10	0,12	0,14
E	0,08	0,04	-	-	-	-	0,06

descriptions. In order to compare the systems with each other, the weighted average has been calculated according to the number of descriptions of each route.

TABLE VII.
RESULT OF COMPARISON BETWEEN GOOGLE MAPS, ROTAFACIL MA AND ROTAFACIL AT

Route	Google Maps	RotaFacil MA	RotaFacil AT
A	0.021	0.105	0.06
B	0.049	0.114	0.104
C	0.089	0.088	0.101
D	0.097	0.133	0.143
E	0.026	0.038	0.065
Weighted Average	0.062	0.104	0.099

From the comparison of RotaFacil MA and Google Maps, we see that RotaFacil MA had a weighted average of 10.42% whereas Google Maps achieved 6.2%. It means that the route descriptions generated by the RotaFacil MA is on average 68.64% more similar to the routes generated by people in comparison to Google Maps.

From the comparison of RotaFacil AT and Google Maps, we note that RotaFacil AT had similar performance to that of RotaFacil MA: 9.93% compared to 6.2% of Google Maps, which means 60.16% higher.

We have noticed no significant difference between RotaFacil MA and RotaFacil AT, which points out that the RotaFacil AT is the better choice for the generation of templates due to the lack of need to manually create the identify and extratct the templates from corpus.

V. CONCLUSION

In this paper, we presented the mashup location system RotaFacil. RotaFacil provides natural language descriptions of route, considering Points of Interest (POI). We detail its automated mechanism for automatic feeding of the POI database system and two different approaches for the creation of language templates from a corpus of route descriptions provided by volunteers. A first approach is to manually identify common patterns of text. The second approach is an adaptation of an algorithm for generating paraphrases and enables the automatic identification of such patterns, thus lowering human effort.

The POI base updating mechanism has proved to be effective for generating natural route descriptions regardless of region and independent from human intervention. We present examples for three different brazilian cities.

The corpus-driven approach to provide liguistic templates in support of generation of route descriptions has shown good results. The set of templates has been improved both quantitatively and in terms of linguistic variability. The quality of generation provided by RotaFacil was evaluated by measuring the similarity of generated route descriptions with a test set containing textual descriptions of same routes provided by volunteers. Experiments have shown that both manual and automatic approaches to creation of linguistic templates, proposed in this paper, lead to generation of route descriptions much closer to the way people actually orient others and themselves in their way if compared to Google Maps.

Once the results has shown no significant difference between both approach, we conclude that the automatic approach for creating linguistic templates should be prioritzed, since it considerably reduces human effort.

A known limitation is that although both mechanism for POI base updating and automatic generation of template is language independent, current version of RotaFacil cannot provide route descriptions in any language other than Portuguese. We are working to extend its template base to other languages.

A mobile version of RotaFacil which takes GPS data of user's current location is also being considered.

ACKNOWLEDGEMENT

The authors thank the *Programa Especial de Incluso em Iniciao Cientfica* (Piic/Proest/Posgrap/UFS) for granting a scholarship to Bruno Barroso.

REFERENCES

[1] J. Stewart, S. Bauman, M. Escobar, J. Hilden, K. Bihani, and M. W. Newman, "Accessible contextual information for urban orientation," in *Proceedings of the 10th international conference on Ubiquitous computing*, ser. UbiComp '08. New York, NY, USA: ACM, 2008, pp. 332–335. [Online]. Available: http://doi.acm.org/10.1145/1409635.1409679

[2] M. Duckhama, S. Wintera, and M. Robinsonb, "Including landmarks in routing instructions," *Journal of Location Based Services*, vol. 4, no. 1, pp. 28–52, 2010.

[3] A. Guimaraes and H. Macedo, "A mashup system to generate route descriptions based on points of interest," in *Proceedings of the 5th Euro American Conference on Telematics and information Systems*, 2010, pp. 1–8.

[4] D. Lin and P. Pantel, "Dirt @sbt@discovery of inference rules from text," in *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, ser. KDD '01. New York, NY, USA: ACM, 2001, pp. 323–328. [Online]. Available: http://doi.acm.org/10.1145/502512.502559

[5] M. Batty, A. Hudson-Smith, R. Milton, and A. Crooks, "Map mashups, web 2.0 and the gis revolution," *Annals of GIS*, vol. 16, no. 1, pp. 1–13, 2010. [Online]. Available: http://www.tandfonline.com/doi/abs/10.1080/19475681003700831

[6] Wikipedia, "Wikipedia," http://en.wikipedia.org/wiki/ (acessado em 24 de outrubro de 2012), 2012. [Online]. Available: http://en.wikipedia.org/wiki/

[7] ProgrammableWeb, "ProgrammableWeb," http://www.programmableweb.com/ (acessado em 13 de novembro de 2012), 2012. [Online]. Available: http://www.programmableweb.com/

[8] Google, "Google Maps API," http://code.google.com/apis/maps, 2012. [Online]. Available: http://code.google.com/apis/maps

[9] E. Reiter and R. Dale, *Building Natural Language Generation Systems*. Cambridge University Press, 2000.

[10] J. D. Moore and J. Oberlander, "Natural language generation: An introduction," 2012. [Online]. Available: http://www.inf.ed.ac.uk/teaching/courses/nlg/lectures/2011/NLG2011Lect1.pdf(acessadoem27deoutrubrode2012)

[11] H. T. Macedo, "A Software Architecture for Ubiquitous Web Browsing with Application to Recommendation Systems," Ph.D. dissertation, Centro de Informtica da Universidade Federal de Pernambuco, 2006.

[12] E. Reiter and R. Dale, "Building applied natural language generation systems," *Nat. Lang. Eng.*, vol. 3, no. 1, pp. 57–87, Mar. 1997. [Online]. Available: http://dx.doi.org/10.1017/S1351324997001502

[13] K. Van Deemter, E. Krahmer, and M. Theune, "Real versus template-based natural language generation: A false opposition?" *Comput. Linguist.*, vol. 31, no. 1, pp. 15–24, Mar. 2005. [Online]. Available: http://dx.doi.org/10.1162/0891201053630291

[14] P. W. Culicover, "Mechanical translation and computational linguistics," in *Paraphrase generation and information retrieval from stored text*, 1968, vol. 11, pp. 78–88.

[15] N. Madnani and B. J. Dorr, "Generating phrasal and sentential paraphrases: A survey of data-driven methods," *Comput. Linguist.*, vol. 36, no. 3, pp. 341–387, Sept. 2010. [Online]. Available: http://dx.doi.org/10.1162/coli-a-00002

[16] D. Jurafsky and J. H. Martin, *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition*, 2nd ed. Prentice Hall, Feb. 2008. [Online]. Available: http://www.worldcat.org/isbn/013122798X

[17] M. Pasca and P. Dienes, "Aligning needles in a haystack: Paraphrase acquisition across the web," in *Natural Language Processing ? IJCNLP 2005*, ser. Lecture Notes in Computer Science, R. Dale, K.-F. Wong, J. Su, and O. Kwong, Eds. Springer Berlin Heidelberg, 2005, vol. 3651, pp. 119–130. [Online]. Available: http://dx.doi.org/10.1007/11562214-11

[18] E. Bick, "The parsing system "palavras": Automatic grammatical analysis of portuguese in a constraint grammar framework," Ph.D. dissertation, Aarhus University, 2000.

[19] L. C. G. Maia and R. R. Souza, "Medidas de similaridade em documentos eletrnicos," 2008.

Rafael Teles was born in Aracaju/SE, Brazil, in 1990. He graduated in Computer Science at the Federal University of Sergipe in 2012. His current scientific research focuses on natural language generation.

Bruno Barros was born in Belo Horizonte/MG, Brazil, in 1991. He is an undergraduated student of Computer Science course at the Federal University of Sergipe. His current academic interest relies on natural interfaces for mobile devices.

Adolfo Guimaraes was born in Aracaju/SE, Brazil, in 1984. He graduated in Computer Science at the Federal University of Sergipe in 2009. He obtained a Master's degree in Computer Science from the Federal University of Minas Gerais in 2013. From July 2013 until the present time, he works as a temporary professor at the Departament of Information System, Federal University of Sergipe, Brazil. His current scientific research focuses on natural language processing, recommender systems and genetic programming.

Hendrik Macedo was born in Aracaju/SE, Brazil, in 1977. He graduated in Computer Science at the Federal University of Sergipe in 1998. He obtained a Master's degree in Computer Science from the Federal University of Pernambuco in 2001 and a doctorate in Computer Science also from the Federal University of Pernambuco in 2006, having done an internship PhD at the University of Paris VI in 2002. From July 2006 until the present time, serves as an associate professor in the Department of Computer Science, Federal University of Sergipe, Brazil, where he held the position of vice-coordinator of the Graduate Program in Computer Science at this University. His current scientific research primarily focuses on speech natural interfaces.

Call for Papers and Special Issues

Aims and Scope

Journal of Emerging Technologies in Web Intelligence (JETWI, ISSN 1798-0461) is a peer reviewed and indexed international journal, aims at gathering the latest advances of various topics in web intelligence and reporting how organizations can gain competitive advantages by applying the different emergent techniques in the real-world scenarios. Papers and studies which couple the intelligence techniques and theories with specific web technology problems are mainly targeted. Survey and tutorial articles that emphasize the research and application of web intelligence in a particular domain are also welcomed. These areas include, but are not limited to, the following:

- Web 3.0
- Enterprise Mashup
- Ambient Intelligence (AmI)
- Situational Applications
- Emerging Web-based Systems
- Ambient Awareness
- Ambient and Ubiquitous Learning
- Ambient Assisted Living
- Telepresence
- Lifelong Integrated Learning
- Smart Environments
- Web 2.0 and Social intelligence
- Context Aware Ubiquitous Computing
- Intelligent Brokers and Mediators
- Web Mining and Farming
- Wisdom Web
- Web Security
- Web Information Filtering and Access Control Models
- Web Services and Semantic Web
- Human-Web Interaction
- Web Technologies and Protocols
- Web Agents and Agent-based Systems
- Agent Self-organization, Learning, and Adaptation
- Agent-based Knowledge Discovery
- Agent-mediated Markets
- Knowledge Grid and Grid intelligence
- Knowledge Management, Networks, and Communities
- Agent Infrastructure and Architecture
- Agent-mediated Markets
- Cooperative Problem Solving
- Distributed Intelligence and Emergent Behavior
- Information Ecology
- Mediators and Middlewares
- Granular Computing for the Web
- Ontology Engineering
- Personalization Techniques
- Semantic Web
- Web based Support Systems
- Web based Information Retrieval Support Systems
- Web Services, Services Discovery & Composition
- Ubiquitous Imaging and Multimedia
- Wearable, Wireless and Mobile e-interfacing
- E-Applications
- Cloud Computing
- Web-Oriented Architectrues

Special Issue Guidelines

Special issues feature specifically aimed and targeted topics of interest contributed by authors responding to a particular Call for Papers or by invitation, edited by guest editor(s). We encourage you to submit proposals for creating special issues in areas that are of interest to the Journal. Preference will be given to proposals that cover some unique aspect of the technology and ones that include subjects that are timely and useful to the readers of the Journal. A Special Issue is typically made of 10 to 15 papers, with each paper 8 to 12 pages of length.

The following information should be included as part of the proposal:

- Proposed title for the Special Issue
- Description of the topic area to be focused upon and justification
- Review process for the selection and rejection of papers.
- Name, contact, position, affiliation, and biography of the Guest Editor(s)
- List of potential reviewers
- Potential authors to the issue
- Tentative time-table for the call for papers and reviews

If a proposal is accepted, the guest editor will be responsible for:

- Preparing the “Call for Papers” to be included on the Journal’s Web site.
- Distribution of the Call for Papers broadly to various mailing lists and sites.
- Getting submissions, arranging review process, making decisions, and carrying out all correspondence with the authors. Authors should be informed the Instructions for Authors.
- Providing us the completed and approved final versions of the papers formatted in the Journal’s style, together with all authors’ contact information.
- Writing a one- or two-page introductory editorial to be published in the Special Issue.

Special Issue for a Conference/Workshop

A special issue for a Conference/Workshop is usually released in association with the committee members of the Conference/Workshop like general chairs and/or program chairs who are appointed as the Guest Editors of the Special Issue. Special Issue for a Conference/Workshop is typically made of 10 to 15 papers, with each paper 8 to 12 pages of length.

Guest Editors are involved in the following steps in guest-editing a Special Issue based on a Conference/Workshop:

- Selecting a Title for the Special Issue, e.g. “Special Issue: Selected Best Papers of XYZ Conference”.
- Sending us a formal “Letter of Intent” for the Special Issue.
- Creating a “Call for Papers” for the Special Issue, posting it on the conference web site, and publicizing it to the conference attendees. Information about the Journal and Academy Publisher can be included in the Call for Papers.
- Establishing criteria for paper selection/rejections. The papers can be nominated based on multiple criteria, e.g. rank in review process plus the evaluation from the Session Chairs and the feedback from the Conference attendees.
- Selecting and inviting submissions, arranging review process, making decisions, and carrying out all correspondence with the authors. Authors should be informed the Author Instructions. Usually, the Proceedings manuscripts should be expanded and enhanced.
- Providing us the completed and approved final versions of the papers formatted in the Journal’s style, together with all authors’ contact information.
- Writing a one- or two-page introductory editorial to be published in the Special Issue.

More information is available on the web site at http://www.academypublisher.com/jetwi/.

(Contents Continued from Back Cover)