

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/352082757>

# Application of Supervised Machine Learning to Extract Brain Connectivity Information from Neuroscience Research Articles

Article in Interdisciplinary Sciences Computational Life Sciences · June 2021

DOI: 10.1007/s12539-021-00443-6

CITATION

1

READS

101

5 authors, including:



Ashika Sharma  
Indian Institute of Technology Madras

3 PUBLICATIONS 4 CITATIONS

[SEE PROFILE](#)



Jaikishan Jayakumar  
Indian Institute of Technology Madras

35 PUBLICATIONS 321 CITATIONS

[SEE PROFILE](#)



Partha Mitra  
The University of Sydney

207 PUBLICATIONS 15,547 CITATIONS

[SEE PROFILE](#)



Sutanu Chakraborti  
Indian Institute of Technology Madras

37 PUBLICATIONS 272 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



The Active Atlas: Combining 3D Anatomical Models with Texture Detectors [View project](#)



# Application of Supervised Machine Learning to Extract Brain Connectivity Information from Neuroscience Research Articles

Ashika Sharma<sup>1,2</sup> · Jaikishan Jayakumar<sup>3</sup> · Partha P. Mitra<sup>4</sup> · Sutanu Chakraborti<sup>2</sup> · P. Sreenivasa Kumar<sup>2</sup>

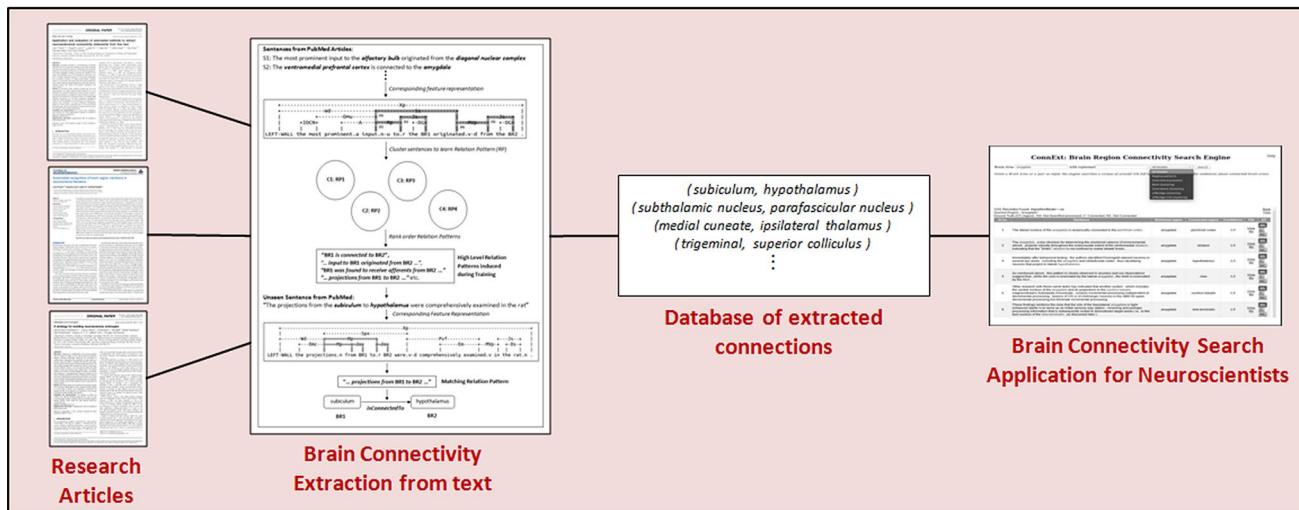
Received: 14 September 2020 / Revised: 15 May 2021 / Accepted: 18 May 2021 / Published online: 2 June 2021  
© International Association of Scientists in the Interdisciplinary Areas 2021

## Abstract

Understanding the complex connectivity structure of the brain is a major challenge in neuroscience. Vast and ever-expanding literature about neuronal connectivity between brain regions already exists in published research articles and databases. However, with the ever-expanding increase in published articles and repositories, it becomes difficult for a neuroscientist to engage with the breadth and depth of any given field within neuroscience. Natural Language Processing (NLP) techniques can be used to mine '*Brain Region Connectivity*' information from published articles to build a centralized connectivity resource helping neuroscience researchers to gain quick access to research findings. Manually curating and continuously updating such a resource involves significant time and effort. This paper presents an application of supervised machine learning algorithms that perform shallow and deep linguistic analysis of text to automatically extract connectivity between brain region mentions. Our proposed algorithms are evaluated using benchmark datasets collated from PubMed and our own dataset of full text articles annotated by a domain expert. We also present a comparison with state-of-the-art methods including BioBERT. Proposed methods achieve best recall and  $F_2$  scores negating the need for any domain-specific predefined linguistic patterns. Our paper presents a novel effort towards automatically generating interpretable patterns of connectivity for extracting *connected* brain region mentions from text and can be expanded to include any other domain-specific information.

## Graphic Abstract

### Brain Region Connectivity Extraction and Search from Neuroscience Research Articles



Extended author information available on the last page of the article

**Keywords** Brain region connectivity extraction · Natural language processing · Neuroscience · Text mining · Machine learning

## 1 Introduction

Active research results in experimental findings and data usually published in the form of articles and publications by scientific community. With the access to modern tools and analysis techniques, there is an exponential increase in published articles, particularly in the neuroscience. A simple search in the widely used Pubmed [1] using the keyword “Brain” results in close to 2 million articles. One of the key challenges in neuroscience research is the analysis of brain connectivity and structure ([2]) and currently a lot of information is already published for this key domain. Brain connectivity experiments include electrophysiological, tract-tracing, and many other types of experiments to find connections between the different regions of the brain. Outcomes of the experiments are reported in scientific literature as publications that exist in the form of large text collections. One way of efficiently collating this information is to develop automated approaches that can extract relevant articles and sentences from scientific literature accessible from search engines such as PubMed, Google Scholar etc. Existing literature about brain connectivity is often conflicted and currently there exists no centralized, synchronized repository to describe the experimental results. Rather, the experiments and related findings are scattered in plain text across individual articles, as noted by Richardet et al. [3]. Text mining approaches, in general, not only provide focused and extracted knowledge but can also aid in identification of potentially undiscovered associations and connections [4] [3] [5] between brain region mentions in reported literature. This paper proposes machine learning-based methods for automated extraction of connectivity between brain region mentions from large text collections like PubMed.

Let us consider the following example sentence *S0* from a PubMed article by Künzle et al. [6].

*“The most prominent input to the olfactory bulb originated from the diagonal nuclear complex”*

The words ‘olfactory bulb’ and ‘diagonal nuclear complex’ can be considered as mentions of brain regions. One can infer by reading sentence *S0* that the two brain regions are *connected*. However, there exists wide discrepancy in the natural language patterns used to describe the sentences. For example, in literature, information about brain region connectivity can be conveyed differently as,

1. “*Connectivity between the olfactory bulb and the diagonal nuclear complex...*”
2. “*Afferents from the diagonal nuclear complex terminate in the olfactory bulb*”

3. “*...projections from diagonal nuclear complex to olfactory bulb... ” etc.*”

Each of the above patterns of connectivity represents a variation of the relation ‘*Brain Region Connectivity*’ and can be called as a *relation pattern* or a *domain pattern* as described by J. R. Hobbs [7]. Challenge for any automated system is not only to extract *connected* pairs of brain region mentions but also to automatically recognize different patterns of connectivity along with variations of a specific pattern. First, this involves identifying sentences that potentially describe the connectivity between brain region mentions. Next step is to decide whether sentence is indicative of a connection between brain region mentions. In Natural Language Processing (NLP), this problem can best be described as *relation extraction* of “*Brain Region Connectivity*” from text.

The NLP based text extraction method, described in this paper should not be looked upon just as a classification technique to identify sentences with *connected* brain regions, but also as methods that can generate human interpretable *relation patterns* from a given corpus. There is a requirement for such an application to be recall-centric. i.e. a neuroscientist using this application to look for *connected* brain regions would not mind going through some weak evidences denoting connectivity, but will not want to miss true connections hidden in large repositories of text.

Relation Extraction from biomedical literature has been applied to other biological fields such as protein-protein interactions and drug-drug interactions [8] [9]. The growing interests in understanding brain connectivity and the subsequent literature research into the field has triggered active research in mining the “*Brain Connectivity*” relations between region mentions in the brain. An initial effort was made by French et al. [10] by compiling a dataset named “*WhiteText*”, containing 1,377 abstracts of articles from the Journal of Comparative Neurology (JCN) on PubMed. This version of the dataset contains 18,242 manually annotated brain regions. The primary focus of French et al. [10] was the automatic extraction of brain region mentions using Conditional Random Fields (CRF) with a rich set of features. French et al. [4] subsequently extended *WhiteText* dataset to include 4276 connection annotations between brain regions. Automatic extraction of *Brain Connectivity* relations from 1377 abstracts was attempted using co-occurrence and kernel-based machine learning methods adapted from protein interaction domain [8]. Shallow Linguistic Kernel [11] achieved 50% precision at 70% recall for sentence-level connectivity statements. French et al. [5] further updated *WhiteText* dataset with 2111 connectivity statements from

additional 1828 abstracts of the JCN. Earlier results were replicated with 50% precision and 67% recall by running Shallow Linguistic Kernel in a cross-validation framework on the updated corpus.

Subsequently, Richardet et al. [3] proposed nine regular expression based rules capturing sentence structure of popularly occurring connectivity patterns in a given corpus. The rules were manually designed using Apache UIMA RUTA workbench [12] and this method increased the precision to 72% on *WhiteText* corpus [4] but at the cost of recall at 12%. Rules are specific in nature and have extracted the *Brain Connectivity* relation from sentences occurring according to predefined rule patterns, with high precision. However, the approach fails to generalize and hence misses out on capturing variations of rule patterns and also other sentences not covered by any of the predefined rules.

Gokdeniz et al. [13] proposed a linguistically motivated approach for connectivity extraction based on predefined patterns over constituency and dependency parse trees of sentences. These patterns were manually designed and were specific to the neuroscience domain. Additionally, as a pre-processing step, sentences in corpus with a higher potential of having *connected* brain regions were selected using a bunch of handcrafted regular expression based patterns similar to Richardet et al. [3]. In the next step, sentences matching any of these regular expressions were checked for the presence of manually identified patterns over parse tree analyses of these sentences. Direction of connections were identified using regular expression pattern templates. This approach yielded high precision of 76.94% on the second dataset of *WhiteText* corpus [5] as compared to 51% for Shallow Linguistic Kernel. Here again, since preprocessing steps considered only sentences occurring according to predefined regular expression patterns, many sentences were discarded, causing recall to dip to a low of 14.59%.

In this paper, three connectivity extraction algorithms have been explored using shallow and deep linguistic analysis of text, categorized based on the type of feature representation used. Proposed techniques have been evaluated on the available benchmark *WhiteText* dataset and compared with methods used in the literature. Additionally, the performance of proposed methods on a newly compiled dataset of full-text articles selected from PubMed is presented. Proposed connectivity extraction techniques do not rely on domain specific, predefined regular expression patterns or patterns defined over parse tree structures of sentences. Instead, focus is on automatically inducing patterns of connectivity in the corpus for extracting *connected* brain regions. This paper provides an automated mechanism for processing large amount of neuroscience literature to identify connections between brain region mentions specified in research articles. We also present a web-based application for brain region connectivity extraction with a choice of algorithms proposed

in the paper. An effort is made towards automatically collating a centralized repository of brain region connectivity information from literature, allowing neuroscientists to validate their experiments with what has been reported previously by the neuroscience community. For e.g., our algorithm may report that two areas are functionally connected which may prompt tract tracing experiments from these results.

## 2 Methods

We have approached the problem of finding connectivity between brain region mentions in a sentence as a two-class classification problem. Proposed classifiers were designed as a pattern matching system that labeled a sentence as positive if the *relation pattern* formed with the brain region pair in the sentence indicated a connection. In case of no evidence of a connection in the sentence, it is considered as negative. For instance, sentence *S0*,

*“The most prominent input to the olfactory bulb originated from the diagonal nuclear complex.”* [6]

was classified as ‘Positive’, since the underlying pattern of connectivity, “... *input to the BR1 originated from the BR2*”, indicated existence of a connection between BR1 and BR2. Here BR1 and BR2 denote Brain Region 1 and Brain Region 2 respectively.

Now consider sentence *S1*,

*“In addition, there are prominent cerebello-subthalamic projections to the zona incerta and the ventral geniculate nucleus.”* [14].

This sentence was classified as ‘Negative’, as the pattern “...*projections to the BR1 and the BR2*”, was not indicative of a connection between BR1 and BR2.

To achieve this classification, algorithm learned the *relation patterns* or patterns of connectivity during training phase and generalized them to form the basis for classifying unseen sentences. Three different classifiers are proposed in this paper categorized based on the feature representation scheme used to automatically generate *relation patterns*.

Proposed classifiers are broadly categorized into Surface Level Feature (SuLF) Classifiers and Syntax Level Feature (SyLF) Classifier based on the feature representation scheme used. The SuLF Classifier uses shallow word-level features and is further divided into Bag-of-Word (SuLF-BoW) and Connectivity Word (SuLF-CW) categories. In SyLF category, constituents of parse structure of the input sentence were used as features for Link Parse based Bridge (SyLF-LPBridge) classifier. Following is an explanation of the three classifiers in greater detail.

1. *Surface level feature (SuLF) Classifier:* Considers the lexical elements/ words present in input as features.

The most prominent input to the BR1 originated from the BR2		
---	--	--

CONTEXT	CORRESPONDING VECTOR
LEFT	{prominent:0.19, input:0.17, to:0.41, the:0.97}
MIDDLE	{originated:0.15, from:0.43, the:0.91}
RIGHT	{}

**Fig. 1** Left, middle and right BoW vectors for sentence “*The most prominent input to the BR1 originated from the BR2*”, considering  $n=4$  where  $n$  represents the window size for left and right contexts

- (a) Bag-of-Word (SuLF-BoW): Given sentence was represented as a vector of words containing the left, middle and right contexts as described by Agichtein et al. [15]. Words in the sentence before ‘*BR1*’ make up the *left* context. Words lying between ‘*BR1*’ and ‘*BR2*’ belong to the *middle* context and words after ‘*BR2*’ form the *right* context. All words in the sentence were stemmed and a vector space model was built for each of the contexts. Each word was represented by its normalized term frequency in the corresponding context. An  $n$ -word window was considered for left and right contexts. An example for BoW vector representation corresponding to sentence  $S_0$  in Sect. 1, considering  $n=4$ , is illustrated in Fig. 1.
- (b) Connectivity Word (SuLF-CW): Connectivity words are important words that emphasize the existence of connectivity between brain region mentions and were identified by domain experts. Complete list of connectivity words used as features is included in supplementary material (Online Resource 1). A binary-valued feature vector was used in this scheme to denote presence or absence of a connectivity word in the middle context of an input sentence. Some of the Connectivity words used in this scheme along with an example feature vector corresponding to sentence  $S_0$  in Sect. 1 are illustrated in Fig. 2.
- 3. *Syntax level feature (SyLF) Classifier*: This scheme performed a deep syntactic analysis of input sentence considering the syntactic parse structure generated by an English language grammar parser. Our parsing technique is based on Link grammar ([16]). Each sentence was parsed using link parser and parse output was used to

afferent	innervate	project
axon	input	originate
connect	interconnect	receive
efferent	pathway	terminate ...

[ afferent:0, axon:0, connect: 0, efferent:0, innervate:0, **input:1**, interconnect:0, pathway:0, project:0, **originate:1**, receive:0, terminate:0 ... ]

**Fig. 2** Sample of connectivity words used in the feature representation and feature vector for the sentence “*The most prominent input to the BR1 originated from the BR2*”

form the feature vector to represent the sentence and to further generate *relation patterns*. Figure 3 shows link parse output for sentence  $S_0$  in Sect. 1. In this approach, syntactic relation between words in sentence, specifically between the brain region mentions was used as features to represent underlying *relation pattern*. This classifier has been termed as the Link Parse-based Bridge (SyLF-LPBridge).

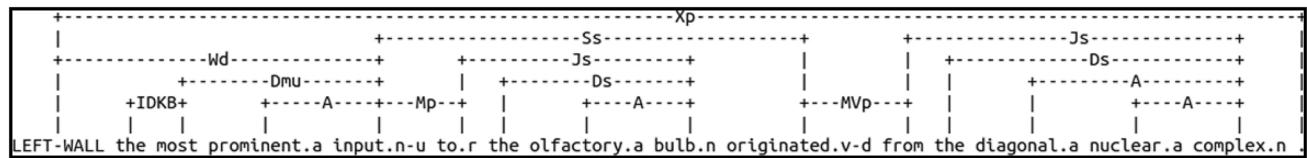
### 3 Implementation

#### 3.1 Handling Complex Sentences

The sentences and patterns described in previous sections contain only two brain region mentions. In reality, sentences with more than two brain region mentions exist in published scientific articles. We have termed such sentences as *complex sentences*. Some articles in PubMed were observed to have over 20 brain regions per sentence. Distribution of number of brain region mentions in the datasets used in the paper are detailed in the supplementary material (Online Resource 2).

Consider the following sentence from Groenewegen et al. [18], “The ***lateral hypothalamus*** and a few cells in the most ***medial part of the subthalamic nucleus*** project to the ***ventral tegmental area***, whereas progressively more ***lateral parts of the ventral mesencephalon***, in particular the ***substantia nigra***, receive input from successively more ***lateral and caudal parts of the subthalamic nucleus***.”

This sentence is complex and contains six brain region mentions highlighted in bold. A sentence with more than a pair of regions was processed for connectivity extraction by considering one pair of brain regions at a time. Extracting connectivity from the above sentence involved inspecting  ${}^6C_2$  pairs of unique brain regions for potential connections. This is more complex than the previous case with just two brain regions. Thus, all combinations were generated for each sentence having more than two unique brain regions. Unique pair of regions in the selected combination was



**Fig. 3** Link parse output for the sentence “*The most prominent input to the olfactory bulb originated from the diagonal nuclear complex.*” Dotted lines represent links between words in the sentence and *.n*, *.v* and *.a* represent the part-of-speech, which in this case stand for noun,

annotated as *BR1* and *BR2*, whereas the other brain regions in the sentence were not marked as entities, but considered as plain text part of the sentence.

Sentences with multiple brain regions in literature are often lengthy and are not handled well by the parsers. One solution to this problem has been proposed by Richardet et al. [3] who impose a limit on the number of brain regions in a given sentence. In addition, there are very lengthy sentences even when the number of brain regions in the sentence was relatively less. This sentence length is a significant limitation to the Link parser. Hence our algorithm was restricted to handle only sentences with a maximum length of 800 characters.

Applying transformations explained above, each sentence in the corpus was converted into a canonical form with exactly two unique brain regions marked up as *BR1* and *BR2*. After pre-processing, our example sentence *S0* in Sect. 1 was transformed into “*The most prominent input to the BR1 originated from the BR2.*”

### 3.2 Algorithm

The proposed classifiers are instance-based supervised learning techniques that extract underlying *relation patterns* in the corpus and utilized these extracted patterns for classification. To find the *relation patterns*, during training phase, each classifier grouped similar sentences and generated a representative pattern for every group/cluster. Each group/cluster captured the underlying *relation pattern*. Sections 3.2.1 and 3.2.2 explain the feature representation, proposed clustering technique, and associated similarity measure used by classifiers. *Relation patterns* learned during training were assigned a confidence value based on certain performance metrics.

During the testing phase, a test sentence was compared with the popular *relation patterns* generated during training. Based on similarity with the representative *relation patterns*, the test sentence was classified as ‘*Connected*’ or ‘*Not Connected*’. Section 3.2.3 explains confidence scoring for generated *relation patterns* and Sect. 3.2.4 is about classification of a test sentence based on similarity with popular *relation patterns* from the training phase. Positive

verb, and adjective respectively. Link *A* connects pre-noun adjectives to nouns, *MVp* connects prepositions to verbs, *Mp* connects nouns to prepositional phrases modifying the nouns, and so on. Detailed documentation of links can be found in [17]

or negative classification for a sentence was done based on textual description of the sentence. Steps in training and testing phases are shown in Figs. 4 and 5, respectively. Figures depict steps in the approach using SyLF-LPBridge classifier.

#### 3.2.1 SuLF Feature Representation and Matching

Surface level classifiers generate a vector of words for each sentence using surface-level lexical features as explained in Sect. 2. Pattern identification was achieved by clustering vector representation of sentences.

SuLF-BoW representation used the following similarity measure for forming clusters among sentences.

$$\text{Match}(v_i, v_j) = w_l(l(v_i).l(v_j)) + w_m(m(v_i).m(v_j)) + w_r(r(v_i).r(v_j)) \quad (1)$$

where  $v_i, v_j$  represent BoW vector representation of sentences  $i$  and  $j$  respectively and  $l, m, r$  represents the left, middle and right context vectors of the corresponding sentence.  $w_l, w_m, w_r$  represent weights assigned to left, middle and right vectors respectively. Equation 1 calculates similarity between input vectors as a sum of weighted dot products of corresponding context vectors.

The SuLF-CW representation uses the following similarity measure for clustering,

$$\text{Match}(v_i, v_j) = v_i.v_j \quad (2)$$

where  $v_i, v_j$  represents connectivity word vector representation of sentences  $i$  and  $j$  respectively (as explained in Sect. 2). Equation 2 calculates similarity between input vectors as a dot product between binary-valued feature vectors of sentences  $i$  and  $j$ .

Single-pass clustering ([19]) with a similarity threshold  $\tau_{sim}$  was used for clustering the vectors. Value for  $\tau_{sim}$  was set by cross-validation. Details of parameter values are given in supplementary material (Online Resource 3). In both SuLF-BoW and SuLF-CW methods, once clusters were formed, centroid of each cluster was found by taking mean of all vectors in the cluster. Each centroid was considered a potential relation pattern or connectivity pattern.

## Training Phase

Sentences from Training Corpus:

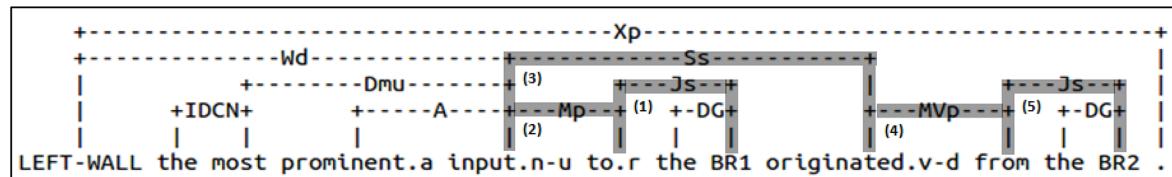
S1: The most prominent input to the **olfactory bulb** originated from the **diagonal nuclear complex**

S2: The **ventromedial prefrontal cortex** is connected to the **amygdala**

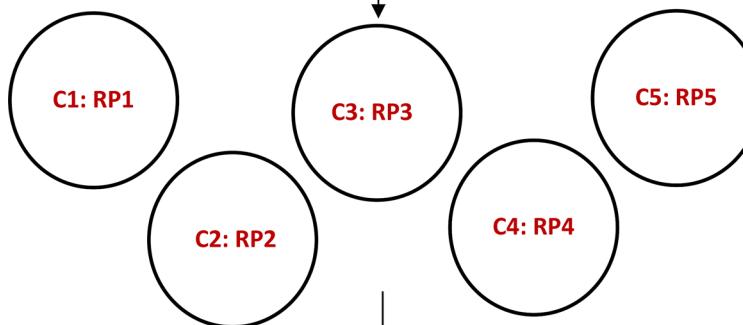
S3: In addition, there were prominent projections to the **zona incerta** and the **ventral geniculate nucleus**

S4: The **parabrachial nucleus** was found to receive afferents from the **lateral prefrontal area**

Corresponding feature representation



Cluster sentences to form Relation Pattern (RP)



Assign Relation Pattern Confidence

C1: *input to BR1 originated from BR2 (0.76)*  
 C2: *projections to BR1 and BR2 (0.21)*  
 C3: *BR1 is connected to BR2 (0.81)*  
 C4: *BR1 was found to receive afferents from BR2 (0.63)*

Rank order Relation Patterns

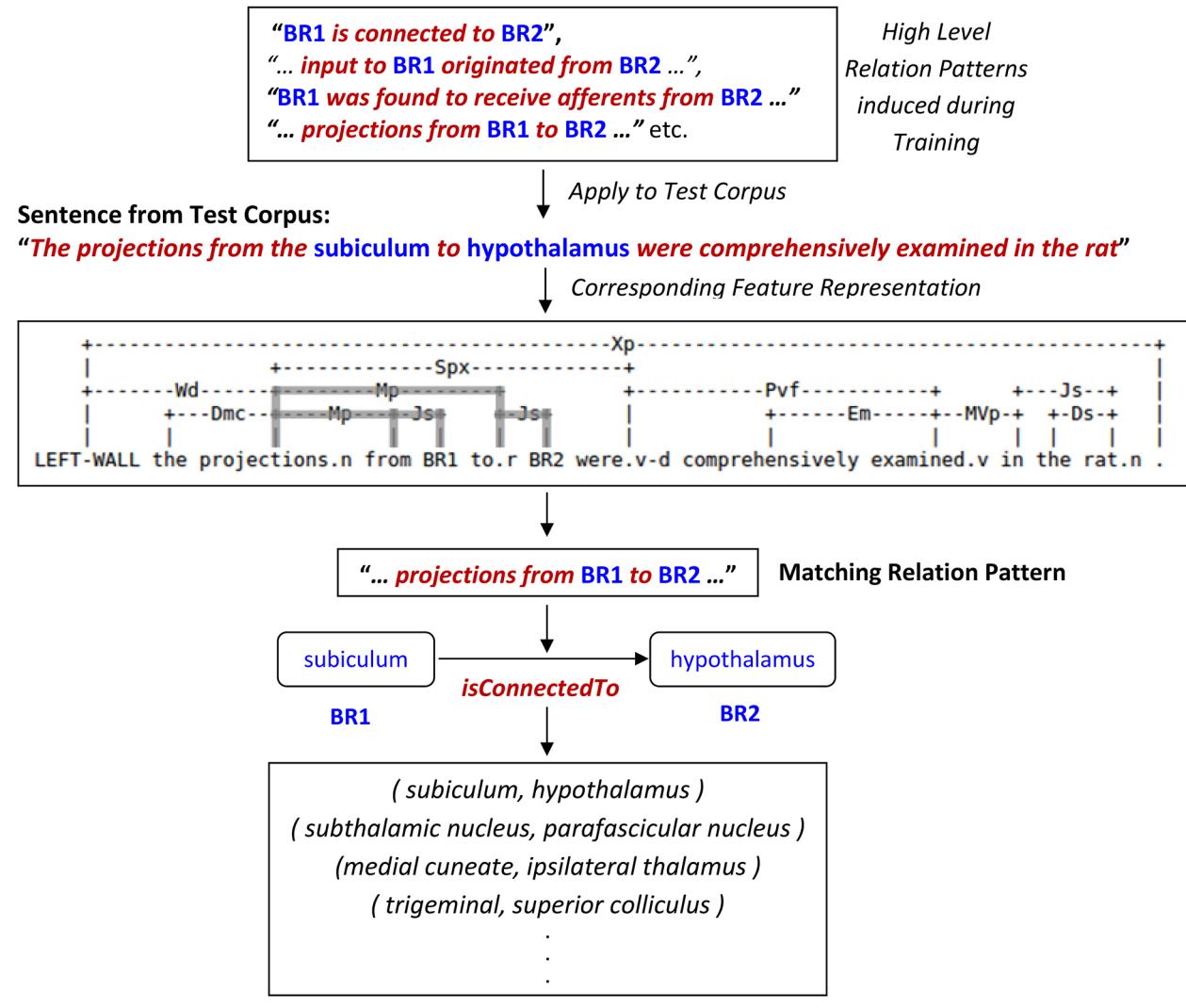
"**BR1 is connected to BR2**",  
 "... **input to BR1 originated from BR2 ...**",  
 "... **BR1 was found to receive afferents from BR2 ...**",  
 "... **projections from BR1 to BR2 ...**" etc.

High Level Relation Patterns induced during Training

**Fig. 4** Steps in training phase using the SyLF-LPBridge classifier. Sentences in the training dataset are parsed to generate the feature representation which in this case is the Bridge highlighted in bold.

Similar feature vectors from different input sentences are clustered together to generate popular high-level relation patterns in training dataset

## Testing Phase



**Fig. 5** Steps in testing phase using the SyLF-LPBridge classifier. Popular relation patterns induced during training are applied to the sentences in test dataset represented in the corresponding feature rep-

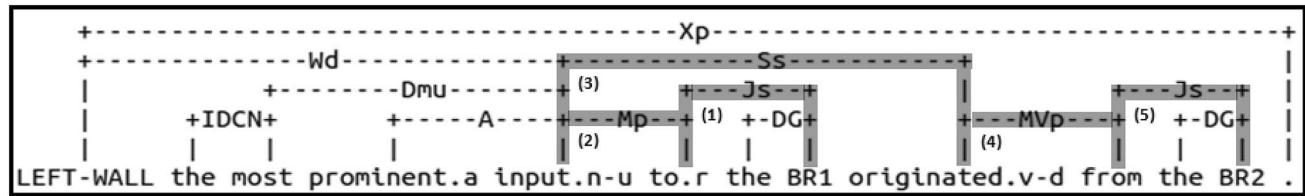
resentation. Test sentences matching any popular relation pattern are used to generate a list of potentially *connected* brain region pairs

### 3.2.2 SyLF Feature Representation and Matching

Syntax level classifier derives features from deep syntactic analysis of sentences. Two brain region mentions in the sentence were replaced by placeholders 'BR1' and 'BR2'. To check for a connection between two regions, shortest path between *BR1* and *BR2* in the link parse structure of the sentence was used. Suchanek et al. [20] called this shortest path between the two placeholders as *Bridge*. Figure 6 shows the Bridge marked in bold for example sentence *S0*. Hypothesis is the shortest path captures the underlying relation pattern in the sentence. Pattern captured by shortest path in *S0* can

be read from left to right as, "*input – to – BR1, originated – from – BR2*".

Feature vector representation of Bridge was formulated by modifying the original Bridge representation of Suchanek et al. [20]. Each edge on the shortest path forming the bridge was represented by a quadruple, which is of the form *[Link, Left-word, Right-word, Context]* and an associated *weight*. *Link* is the label on the edge representing the link type, *Left-word* and *Right-word* are stemmed versions of the two words connected to the edge. *Context* represents portion of the sentence where the edge occurs and takes as value one of left, middle and right enumerated as {1,2,3} respectively. *Context* was found to be important for relation extraction [15]. For



**Fig. 6** Bridge representation for sentence “The most prominent input to the BR1 originated from the BR2”. Sequence of links in shortest path ‘(BR1 – to – input – originated – from – BR2)’ is numbered below the links. Link *Js* connects preposition ‘to’ to object ‘BR1’. Link *Mp* is used for preposition ‘to’ that modifies noun ‘input’. In

instance, an edge is considered to occur in the middle context if associated Left-word appears before *BR2* and Right-word appears after *BR1*. Lastly, *weight* depicts importance of quadruple in the shortest path. Details about weight computation are explained in the supplementary material (Online Resource 4). Shortest path has thus been represented as a set of weighted quadruples between two brain region entities.

Bridge representation for sentence *S0* is  $\{[\{‘Js’, ‘to’, ‘BR1’, 1\}, 0.65], [\{‘Mp’, ‘input’, ‘to’, 1\}, 0.51], [\{‘Ss’, ‘input’, ‘origin’, 2\}, 0.43], [\{‘MVp’, ‘origin’, ‘from’, 2\}, 0.56], [\{‘Js’, ‘from’, ‘BR2’, 2\}, 0.24]\}$ , where each quadruple represents one edge on the shortest path forming the bridge.

Bridge representation was used to formulate features for sentences in the corpus. To induce commonly occurring patterns from training dataset, sentences having similar feature vectors, in this case, the bridge representation, were clustered together. Clustering was performed using a similarity measure computed as a weighted edit distance between two bridges. Edit distance as originally described by Wagner et al. [21] measures similarity between two character strings based on a minimum number of insert/delete/substitute operations required to make both strings equal. This notion of edit distance was extended for calculating similarity between bridge representations. Quadruples contained in the bridges were compared in a manner similar to comparing characters of a string in the original edit distance algorithm. Number of insert/delete/substitute operations (with associated penalty values) needed to transform quadruples in one bridge to another indicates the measure of similarity between two bridges. Similarity between quadruples of two bridges was estimated by comparing the corresponding contexts, links, and the two words in the quadruples. To enable matching of similar bridges, representation of words and links was generalized. In case of words, all *nouns* were considered to be equivalent and similarly, all *adjectives* were considered equivalent. For links, singular and plural forms of a link were considered equivalent. Furthermore, to make the algorithm more specific to neuroscience domain, all occurrences of connectivity words in the bridge were also considered equivalent.

turn, word ‘*input*’ connects verb ‘*originated*’ by link *Ss* which is used to connect singular words to singular verb forms. Verb ‘*originated*’ connects preposition ‘*from*’ and finally, link *Js* connects preposition ‘*from*’ to object ‘*BR2*’”

Algorithm for computing weighted edit distance between two bridges is described in Algorithm 1. Here  $B_1$  and  $B_2$  are two bridges with lengths  $l_1$  and  $l_2$  respectively. Length of a bridge is the number of quadruples contained in the bridge. Each edit operation has an associated penalty value. For insertion and deletion operations, weight of the quadruple was assumed as the penalty value. This ensured penalty applied was proportional to the importance of quadruple as defined by its weight. In case of substitute operation, penalty values used in different conditions are detailed in supplementary material (Online Resource 3). Weighted edit distance was further normalized using Eq. 3 to make the value lie between 0 and 1. Denominator in Eq. 3 represents maximum possible edit distance between two bridges  $B1$  and  $B2$  with lengths  $l_1$  and  $l_2$  respectively.

$$\text{NormEditDist}(B_1, B_2) = \frac{\text{WtdBrdgEditDist}(B_1, B_2, l_1, l_2)}{2 * \min(l_1, l_2) + |l_1 - l_2|} \quad (3)$$

#### Algorithm 1 Weighted Bridge Edit Distance

```

1: procedure WtdBrdgEditDist( $B_1, B_2, l_1, l_2$ )
2:   for  $i = 0$  to  $l_1$  do
3:     for  $j = 0$  to  $l_2$  do
4:       if ( $i == 0$  and  $j == 0$ ) then
5:         matrix( $i, j$ ) = 0
6:       else if ( $i == 0$ ) then
7:         matrix( $i, j$ ) = weight( $\text{quadruple}_{j-1}$ )
8:       else if ( $j == 0$ ) then
9:         matrix( $i, j$ ) = weight( $\text{quadruple}_{i-1}$ )
10:      else
11:        Cost = Substitute( $\text{quadruple}_{i-1}$ ,
12:                            $\text{quadruple}_{j-1}$ )
13:        matrix( $i, j$ ) = min(matrix( $i, j-1$ ) +
14:                           weight( $\text{quadruple}_{j-1}$ ), matrix( $i-1, j$ ) +
15:                           weight( $\text{quadruple}_{i-1}$ ), matrix( $i-1, j-1$ ) + Cost)
16:   return matrix( $i, j$ )

```

Similarity between two bridge representations has been computed using normalized edit distance as shown in Eq. 4. Bridges were clustered by defining a minimum similarity value that two bridges must have to be grouped into the

same cluster. Idea is that each cluster with its bridge represents a relation pattern. Thus, by representing sentences by their bridge and clustering similar bridges, underlying relation patterns in the data were automatically induced by the algorithm.

$$\text{Similarity}(B_1, B_2) = 1 - \text{NormEditDist}(B_1, B_2) \quad (4)$$

Experimental procedure and parameter choices assumed in both SULF and SyLF classifiers are detailed in the supplementary material (Online Resource 3).

### Iterative Least Cost Parsing (ILCP)

Analysis of bridges generated from corpus revealed that link parser was not able to generate bridge representation for some sentences. This resulted in the failure of the SyLF-LPBridge algorithm to process such sentences. Link parser failed to generate a bridge for a sentence under the three cases listed below:

1. No linkage was generated for the sentence
2.  $BR1$  or  $BR2$  was not covered in the parsed output of the sentence
3. No path was found between  $BR1$  and  $BR2$

First case occurred when link parser was not able to handle the input sentence and does not generate even a single parse. This could possibly occur because of an internal link parse error. Second case occurred when sentence was parsed, but parse structure did not pass through either  $BR1$  or  $BR2$  or both, due to which, a bridge could not be generated. In the third case, parse structure was generated and was passing through both  $BR1$  and  $BR2$ , but due to parsing errors, a path was not discovered between  $BR1$  and  $BR2$ . Out of total link parse failures on *WhiteText* dataset, 90% were found to be belonging to case 1, around 9% to case 2, and the remaining 1% to case 3. Case 1 occurred when no linkage or parse structure was generated for a sentence as it might be ill-formed or the link parser was not able to handle the complexity in the sentence, leading to link parser failure. As no parse output was generated, a bridge could not be generated for such a sentence and, therefore, this case was ignored. Instead, 10% failures in cases 2 and 3 were targeted to come up with the Iterative Least Cost Parse (ILCP) approach to mitigate these failures.

On parsing a sentence using link parser, multiple parse outputs are generated and each of them has an associated cost. Total cost of a particular parse is sum of costs of individual connectors. Cheapest parse was assumed to be the most likely parse ([22]). To find a bridge, the algorithm always picked the parse with the least cost. ILCP approach handles the second and third cases by iteratively looking for the next, least cost, successful parse not falling into any

of the failure cases until it finds one. If none of the parse alternatives succeed, algorithm gives up on the sentence and generates an empty bridge. In case of *WhiteText* dataset, by implementing Iterative Least Cost Parsing approach, a 33% increase has been achieved in the number of bridges generated as compared to when ILCP was not used. Detailed statistics on number of sentences falling into the above cases 1, 2, and 3, for each dataset used in the paper, is included in Supplementary material (Online Resource 5)

### 3.2.3 Relation Pattern Confidence Scoring (Training)

In the training phase, each generated relation pattern was assigned a confidence score based on its selectivity and coverage. Selectivity measures the extent of true connection mentions matching the relation pattern and was computed using Eq. 5.  $P.\text{positive}$  and  $P.\text{negative}$  are the number of positive and negative instances matching the relation pattern  $P$  respectively in the training dataset.

$$\text{Selectivity}(P) = \frac{P.\text{positive}}{(P.\text{positive} + P.\text{negative})} \quad (5)$$

Coverage shows how many positive instances are covered by relation pattern  $P$  among all possible positive instances matching any pattern  $P_i$ , where  $i$  ranges over set of all  $n$  popular relation patterns generated during training. Coverage brings out the relative importance of pattern  $P$  and was computed as in Eq. 6.

$$\text{Coverage}(P) = \frac{P.\text{positive}}{\max_{1 \leq i \leq n}(P_i.\text{positive})} \quad (6)$$

Final confidence score of pattern  $P$  is a function of its selectivity and coverage as shown in Eq. 7.

$$\text{PatternConf}(P) = \text{Selectivity}(P) * \text{Coverage}(P) \quad (7)$$

Relation patterns are rank-ordered based on their confidence scores and most confident patterns are selected as important connectivity indicators in the training dataset. Given an unseen test sentence, this bank of highly confident relation patterns was used to extract a *connected* brain region pair from sentence in case one exists.

### 3.2.4 Classification of Unseen Input (Testing)

In testing phase, as a first step, each test sentence was converted into its corresponding feature representation  $T$ . A test sentence was classified based on its confidence value, which is a function of its similarity with popular relation patterns generated during training, as explained in 3.2.3. Normalized confidence value for a test sentence  $T$  was computed as shown in Eq. 8. Equation is the sum of similarity between  $T$  and the set of relation patterns  $P_i$  matching  $T$ , weighted by

confidence of the relation pattern  $P_i$  as computed in Eq. 7. Here,  $k$  patterns are the set of top  $k$  highly confident relation patterns that were most similar to test sentence  $T$ .  $P_1$  to  $P_k$  are popular relation patterns derived during training phase.

$$\text{Confidence}(T) = \frac{\sum_{i=1}^k \text{PatternConf}(P_i) * \text{Similarity}(P_i, T)}{k} \quad (8)$$

More a test sentence was similar to one or more high confidence patterns, higher was the confidence of the test sentence  $T$ . If confidence of  $T$  was greater than a threshold confidence  $\tau_c$ , test sentence was classified as positive, else as negative. Values for  $\tau_c$  and  $k$  were derived by cross-validation. Details of parameter values are given in supplementary material (Online Resource 3).

### 3.3 Datasets

Evaluation of proposed approaches was performed on a benchmark dataset *WhiteText* [4] [5]. This dataset comprises of abstracts from articles listed in PubMed [1]. More specifically, these abstracts are from articles published in the Journal of Comparative Neurology. Brain regions and connections in the dataset were manually identified and annotated by a set of human annotators [4] [5]. To the best of our knowledge, *WhiteText* is the only benchmark dataset for depicting brain connectivity. Brain region annotations provided with the dataset were used for NER of the brain regions. Dataset includes simple sentences with two brain region mentions per sentence, making up 52% of *WhiteText* dataset and more complex sentences with more than two brain region mentions, accounting for remaining 48%. Dataset was preprocessed for handling sentences with multiple brain regions as described in Sect. 3.1. After preprocessing, 20470 sentences were derived from first version of *WhiteText* dataset [4] and were used to form the training dataset. Similarly, second version of *WhiteText* [5] was used to derive 10252 sentences to form the testing dataset.

Additionally, two other datasets have been used to evaluate the performance of the proposed methods. First is “*10NeuroPubMed*”, a dataset compiled by an experienced neuroscientist (domain expert) on abstracts and full-text articles from PubMed (all sentences from 10 randomly chosen mouse brain articles). Brain regions within these articles were identified using Allen’s Brain Atlas (ABA) [23] for the mouse brain. Ground truth annotation for connection mentions in each sentence of the “*10NeuroPubMed*” dataset was manually marked up by the domain expert and these annotations were treated as the gold standard for evaluating the performance of proposed approaches on *10NeuroPubMed*. Preprocessing (Sect. 3.1) of the dataset was performed for handling sentences with multiple brain regions. This resulted in 2435 sentences which were used to elicit brain connectivity information. These 2435

sentences were exclusively used in test mode to evaluate the performance of proposed methods on this dataset. Learning model used for this experiment is the one trained over 20470 sentences from the first version of *WhiteText* dataset [4]. ABA dictionary used to markup brain regions in this dataset is made available in the supplementary material (Online Resource 6). *10NeuroPubMed* dataset including the ten annotated articles after preprocessing and merging is included as supplementary material (Online Resource 7).

Next dataset is “*2BRWhiteText*”, which is a subset of *WhiteText* dataset with sentences containing only two brain region mentions. Brain region annotations provided in *WhiteText* dataset [4] [5] were used for NER of brain regions. Distribution of number of brain region mentions per sentence in *WhiteText* dataset revealed that around 52% of the sentences have only two brain region mentions in them [24]. This subset corpus was used as a dataset to target relatively simple sentences with exactly two brain region mentions per sentence. Considering sentences with only two unique brain regions, the training dataset derived from first version of *WhiteText* [4], had 2133 sentences and the test dataset from second version [5] had 1071 sentences. For ease of reference, this dataset is included as supplementary material (Online Resource 8 and Online Resource 9).

### 3.4 Evaluation Measures

Precision, Recall, and F-score ([25]) have been used as performance metrics for evaluation. Precision indicates percentage of sentences with true connection mentions out of the total sentences classified as having a connection by the proposed approach. Recall indicates percentage of sentences classified as having a connection mention by the proposed approach, out of the total number of sentences with true connection mentions in the dataset.  $F_\beta$  score is the weighted harmonic mean of precision and recall.  $F_1$  is a balanced measure with equal importance to precision and recall, whereas  $F_2$  gives more importance to recall over precision.  $F_2$  score, a variation of F-score, assigns more weight to recall as compared to precision. For reasons previously mentioned, the current application domain requires high recall over high precision, and thus relies more on  $F_2$  score as the basis for comparison.

## 4 Results

### 4.1 Main Experiment

Approaches proposed in this paper, surface-level feature based *SuLF-BoW*, *SuLF-CW*, and syntax level feature-based *SyLF-LPBridge* classifiers were evaluated on the benchmark *WhiteText* dataset. Additionally, *SyLF-LPBridge* and

SuLF-CW based classifiers were cascaded. This kind of cascade of results of a high recall method, followed by a high precision method aided in pruning of false positives generated by the high recall method and thereby boosted precision. Thus, the output of SyLF-LPBridge was fed to SuLF-CW based classifier, leading to SyLF-LPBridge + SuLF-CW cascade approach. Comparison with methods used in literature for discovering connectivity between brain regions is presented. First method is *Shallow Linguistic Kernel (SLK)* used by French et al. [5], adapted from Tikk et al. [8] for extracting protein-protein interaction. Second method is *UIMA RUTA rules*, the rule-based method proposed by Richardet et al. [3], and third method is *Linguistically Motivated Approach (LMA)* by Gokdeniz et al. [13].

Comparison of proposed methods was also made with the deep learning based transfer learning approach, BioBERT [26]. BioBERT was built over BERT (Bidirectional Encoder Representations from Transformers) [27], a state-of-the-art pre-trained language representation model using a large neural network. BERT was trained on entire English Wikipedia containing 2,500M words and BooksCorpus [28] containing 800M words for performing downstream tasks like Natural Language Inferencing, Question Answering, etc.

BioBERT is an extension of BERT customized for biomedical domain. BioBERT is based on pre-training BERT model on a large repository of [PubMed](#) and [PMC articles](#). With this pre-training, BioBERT had shown to perform well on fine-tuned tasks such as Named Entity Recognition, Relation Extraction, etc., in the biomedical domain. Downstream task for *Relation Extraction* provided with BioBERT was used to compare with proposed brain connectivity extraction approaches.

First version of *WhiteText* dataset [4] with 4276 connection annotations was used for training and second version of *WhiteText* [5] with 2111 connections was used for testing. Both French et al. [5] and Gokdeniz et al. [13] used *WhiteText* to evaluate their algorithms for brain connectivity extraction task. training-test configuration adopted for *WhiteText* dataset by this experiment is exactly the same as that used in [5] and [13].

Among the 10252 sentences in test dataset, SyLF-LPBridge was able to run only on 8177 sentences, due to link parse issues explained in Sect. 3.2.2. For a fair comparison, SLK, UIMA RUTA rules, and the BioBERT approaches were evaluated on this 8177 sentence subset of the test dataset.

Code for SLK as pointed by Tikk et al. [8], with default parameterization was used and in case of RUTA, the Apache UIMA RUTA workbench [12] was used to report results. For BioBERT, the pre-trained model, “BioBERT-Base v1.0 (+PubMed 200k + PMC 270K)” [29] was used with default parameterization of relation extraction code in [30]. Here again, first version of *WhiteText* dataset [4] was used for

**Table 1** Precision (P), Recall (R),  $F_1$  and  $F_2$  score computed on *WhiteText* dataset

Approach	P in %	R in %	$F_1$ score	$F_2$ score
SLK	52	66	0.58	0.62
UIMA RUTA Rules	<b>87</b>	10	0.18	0.12
SuLF-BoW †	20	71	0.31	0.46
SuLF-CW †	43	66	0.52	0.60
SyLF-LPBridge †	33	<b>75</b>	0.46	0.60
SyLF-LPBridge + SuLF-CW †	54	70	<b>0.61</b>	<b>0.67</b>
BioBERT	82	31	0.45	0.35
LMA *	77	15	0.25	0.17

Numbers highlighted in bold indicate best performance with respect to considered evaluation measure

† indicates methods proposed in this paper

\* Results of LMA are computed by running on entire *WhiteText* dataset (as reported by Gokdeniz et al. [13]), while results of all other methods are based on the 8177 sentence subset of *WhiteText* dataset. Details are given in Sect. 4.1

training and second version of *WhiteText* [5] was used for testing.

In the case of LMA, as code for the approach has not been made available, results reported by Gokdeniz et al. [13] on entire *WhiteText* dataset with the same training-test configuration is reported.

Table 1 shows results of proposed classifiers on *WhiteText* dataset. Numbers highlighted in bold in the experimental results indicate best performance with respect to considered evaluation measure.

Results show that RUTA rules achieved the best precision. Optimizing precision, led to a significant decrease in recall for UIMA RUTA rules, while SLK struck a good balance between precision and recall. Among methods proposed in this paper, SuLF-CW classifier achieved relatively higher precision at 43%. SyLF-LPBridge using ILCP had a highest recall among all methods, but lesser precision. By applying SyLF-LPBridge + SuLF-CW cascade, the method was able to achieve the best balance between precision and recall and thereby the best  $F_1$  and  $F_2$  scores. Cascade classifier performed better than BioBERT and previous state-of-the-art, SLK. In the case of BioBERT, precision was high, but recall was limited to 31%.  $F_1$  score was comparable to SyLF-LPBridge classifier, whereas  $F_2$  score was significantly lesser than SyLF-LPBridge and cascade classifiers.

Loss in the precision of SyLF-LPBridge can be attributed to complex nature of sentences in this dataset, with a large number of brain regions adding to noise in parse structures and hence an increase in the number of false positives. *WhiteText* dataset contains sentences with over 20 brain regions in some cases and hence was noisy. This could potentially be dealt with by breaking complex sentences into simple ones and then running SyLF-LPBridge approach.

Next issue was of lengthy sentences. In *WhiteText* dataset, around 7.5% of the sentences were found to have more than 800 characters in length. As link parser was not optimized to process lengthy sentences, the algorithm only handles sentences below 800 characters. One way to manage this length restriction could be to replace lengthy, multi-word brain region names in a sentence by abbreviations.

To process the entire dataset, lengthy sentences not handled by SyLF-LPBridge along with the 2075 sentences that SyLF-LPBridge method could not process, were routed to the BioBERT model. BioBERT resulted in a precision of 76% with 53% recall on these sentences. This way, BioBERT can be used to complement SyLF-LPBridge only at the loss of interpretability of the results, but providing 100% coverage on the dataset.

As already mentioned, LMA was evaluated on the entire test dataset of *WhiteText*. LMA improved precision by pre-processing input sentences using manually defined regular expression patterns to select only those sentences containing a connectivity pattern. This significantly increased precision, but at the cost of recall at just 15%.

Critical study of results related to this experiment is discussed in Sect. 5.

## 4.2 Other Experiments

Performance evaluation of proposed methods was done on two other datasets described in Sect. 3.3. First was the *10NeuroPubMed*. Classifiers SuLF-BoW, SuLF-CW, and SyLF-LPBridge were evaluated on *10NeuroPubMed* dataset and results were compared with UIMA RUTA rules and BioBERT. RUTA rules using the Apache UIMA RUTA workbench [12] were executed on the *10NeuroPubMed*, and in case of BioBERT, the pre-trained model, “BioBERT-Base v1.0 (+PubMed 200k + PMC 270K)” [29] was used with default parameterization of relation extraction code in [30]. In this experiment, training models built on abstracts of *WhiteText* dataset [4] were used for computing performance of BioBERT and proposed classifiers on *10NeuroPubMed* dataset. Results for LMA in [13] were reported only on *WhiteText* dataset and since code for LMA is not made available by the authors, comparison with this dataset has not been presented. Table 2 shows performance on *10NeuroPubMed* dataset.

*10NeuroPubMed* is a collection of full-text articles. Sentences in this dataset have more than two brain regions and also have brain region co-references spanning across sentences. Precision of proposed classifiers on this dataset was relatively low compared to benchmark dataset *WhiteText*. This can be attributed to the difference in the nature of the two datasets. *WhiteText* dataset used for training the models comprises of sentences only from titles and abstracts. Therefore the sentences in *WhiteText* are relatively short, concise,

**Table 2** Precision (P), Recall (R),  $F_1$  and  $F_2$  score computed on *10NeuroPubMed* dataset

Approach	P	R	$F_1$	$F_2$
UIMA RUTA Rules	<b>83</b>	7	0.13	0.09
SuLF-BoW †	12	76	0.21	0.37
SuLF-CW †	28	75	0.41	0.56
SyLF-LPBridge †	21	78	0.33	0.50
SyLF-LPBridge + SuLF-CW †	38	<b>84</b>	0.53	<b>0.68</b>
BioBERT	53	68	<b>0.59</b>	0.64

Numbers highlighted in bold indicate best performance with respect to considered evaluation measure

† indicates methods proposed in this paper

and have very few long-range dependencies as opposed to *10NeuroPubMed*.

In contrast, recall values are comparable in *10NeuroPubMed* and *WhiteText*. As shown in Table 2, cascade of SyLF-LPBridge + SuLF-CW boosted the precision, by taking it to 38%. UIMA RUTA rules achieved the best precision, but very poor recall of 7%, since this method retrieved very few connections, most of which were correct, but at the same time, missed out on a large proportion of actually true connections. Cascade classifier achieved the best recall and  $F_2$  score compared to the counterparts and once again proved to strike a good balance between precision and recall while optimizing recall. Relatively long sentences in this dataset caused the precision of BioBERT to drop by 29% as compared to its performance on *WhiteText* dataset; whereas recall seems to be better, resulting in a good  $F_1$  score. Despite this,  $F_2$  score was lesser than SyLF-LPBridge + SuLF-CW cascade classifier.

Next dataset is *2BRWhiteText*. SuLF-BoW, SuLF-CW and SyLF-LPBridge classifiers were evaluated on “*2BRWhiteText*” dataset. Results were compared with SLK, UIMA RUTA rules, and BioBERT. For the purpose of comparison, code of SLK as pointed by Tikk et al. [8], with default parameterization was evaluated on *2BRWhiteText* dataset. Similarly, RUTA rules designed by Richardet et al. [3] were executed using Apache UIMA RUTA workbench [12]. For BioBERT, the pre-trained model, “BioBERT-Base v1.0 (+PubMed 200k + PMC 270K)” [29] was used with default parameterization of relation extraction code in [30]. In case of LMA [13], results have not been reported on the two brain region subset of *WhiteText* dataset. Code for LMA is not made available by the authors and therefore comparison with this dataset has not been presented. Table 3 shows results of this experiment.

Results reveal that UIMA RUTA rules achieved the best precision, but least recall. This was because rules were very specific and did not cover all patterns of connectivity and their variations. SuLF-BoW achieved the best recall, but

**Table 3** Precision (P), Recall (R),  $F_1$  and  $F_2$  score computed on *2BRWhiteText* dataset

Approach	P	R	$F_1$	$F_2$
SLK	78	67	0.72	0.69
UIMA RUTA Rules	<b>92</b>	28	0.43	0.32
SuLF-BoW †	32	<b>97</b>	0.48	0.69
SuLF-CW †	61	56	0.58	0.57
SyLF-LPBridge †	41	90	0.57	<b>0.73</b>
SyLF-LPBridge + SuLF-CW †	64	56	0.60	0.58
BioBERT	80	68	<b>0.73</b>	0.70

Numbers highlighted in bold indicate best performance with respect to considered evaluation measure

† indicates methods proposed in this paper

with limited precision. SLK on the other hand had a higher  $F_1$  score, while the SyLF-LPBridge achieved a good recall and the best  $F_2$  score. SyLF-LPBridge + SuLF-CW cascade classifier resulted in an improvement of precision and  $F_1$  score compared to individual approaches. BioBERT had a better balance between precision and recall and thus, better  $F_1$  score on *2BRWhiteText* dataset compared to all other methods listed in Table 3.

## 5 Discussion

We present the performance of our algorithm on three datasets: benchmark dataset (*WhiteText*), a subset of *WhiteText* dataset limited to sentences with only two unique brain regions (*2BRWhiteText*) and a compiled set of full-text articles from PubMed (*10NeuroPubMed*). Among the proposed classifiers, the SuLF-BoW approach uses the frequency of occurrence of words, which can be directly considered as a statistical measure. SuLF-CW leverages domain knowledge about the set of words indicative of connectivity in the corpus. Finally, the SyLF-LPBridge uses linguistic information in the sentences by parsing and uses syntactic features to represent the sentences. The SuLF-BoW has a high recall, denoting that, this technique retrieves many sentences but with limited precision. On the other hand, SuLF-CW retrieves many correct sentences as it has higher precision, but misses many sentences with a true connection mention.

The SyLF-LPBridge classifier gives maximum  $F_2$  score, striking the best trade-off between precision and recall while optimizing recall for *2BRWhiteText* dataset with simple sentences. *WhiteText* dataset contains complex sentences with more number of brain regions. This sometimes results in noisy relation patterns for extracting connections between brain region pairs resulting in a drop

in precision of SyLF-LPBridge on this dataset; however, recall was higher. SuLF-CW achieved better precision and F-score indicating that factoring in domain knowledge for building features helps classify the dataset better. When precision is important, cascade method SyLF-LPBridge + SuLF-CW can be used to achieve better precision while not greatly compromising on recall. Thus, cascade method was able to achieve balanced performance, resulting in best  $F_1$  and  $F_2$  scores compared to all the other methods.

Precision of proposed classifiers on full-text data can be improved further by training classification models on full-text articles rather than abstracts in the benchmark corpus. Connectivity ground truth for the full-text article dataset *10NeuroPubMed* was manually annotated by a domain expert, whereas the SyLF-LPBridge approach is purely based on supervised learning with syntactic knowledge from link parse structures. This made the evaluation of the approach relatively tough.

### 5.1 Using Connectivity words in SYLF-LPBridge feature representation

Connectivity words are important indicators of domain knowledge and can be leveraged to improve feature representation using syntax analysis of input. Potentially, words in the sentence could be weighed differently based on factors like their importance, whether it is a connectivity word, etc. Towards this, experiments were conducted to leverage the presence of connectivity words in SYLF-LPBridge feature representation. Consider sentence S2, “These results suggest that many GABAergic neurons may represent interneurons in the BR1 and potentially exert an inhibitory influence on adjacent BR2 projecting cholinergic neurons”.

S2 was falsely classified as *Negative* by SYLF-LPBridge method. Reason being it was hypothesized that *middle* context was most important to detect connectivity, followed by the *left* and finally the *right* context. But S2 contains an important connectivity indicator in the form of the word ‘*projecting*’ occurring in *right* context, and hence was assigned least weightage. In an attempt to leverage this, algorithm was tweaked to dynamically assign a higher weight to context where a connectivity word is present. This assigned more weight to *right* context compared to the other two contexts in bridge representation of sentence S2. This change tuned the algorithm to correctly classify the sentence as *Positive*. This experiment showed that differently weighting the contexts based on their importance can help improve performance of the algorithm.

Sl no.	Sentence	Retrieved region	Connected region	Confidence	File	GT
1	The lateral nucleus of the <i>amygdala</i> is reciprocally connected to the <i>perirhinal cortex</i> .	amygdala	<i>perirhinal cortex</i>	1.0	<a href="#">View file</a>	<span style="background-color: blue; color: white;">NS</span>
2	The <i>amygdala</i> , a key structure for determining the emotional valence of environmental stimuli , projects robustly throughout the rostral-caudal extent of the ventromedial <i>striatum</i> , indicating that the "limbic" <i>striatum</i> is not confined to rostral striatal levels .	amygdala	<i>striatum</i>	1.0	<a href="#">View file</a>	<span style="background-color: blue; color: white;">NS</span>
3	Immediately after behavioral testing , the authors identified Fluorogold-stained neurons in several key areas , including the <i>amygdala</i> and orbitofrontal cortex , thus identifying neurons that project to lateral <i>hypothalamus</i> .	amygdala	<i>hypothalamus</i>	1.0	<a href="#">View file</a>	<span style="background-color: blue; color: white;">NS</span>
4	As mentioned above , this pattern is clearly observed in anurans and our observations suggest that ; while the core is innervated by the lateral <i>amygdala</i> , the shell is innervated by the <i>MeA</i> .	amygdala	<i>mea</i>	1.0	<a href="#">View file</a>	<span style="background-color: blue; color: white;">NS</span>
5	Other research with these same tasks has indicated that another system , which includes the central nucleus of the <i>amygdala</i> and its projections to the <i>nucleus basalis magnocellularis</i> /substantia innominata , controls incremental processing independent of decremental processing : lesions of CN or of cholinergic neurons in the nBM /SI spare decremental processing but eliminate incremental processing .	amygdala	<i>nucleus basalis</i>	1.0	<a href="#">View file</a>	<span style="background-color: blue; color: white;">NS</span>
6	These findings reinforce the view that the role of the basolateral <i>amygdala</i> in light-enhanced startle is to serve as an initial sensory way station , receiving and perhaps processing information that is subsequently routed to downstream target areas (ie , to the bed nucleus of the <i>stria terminalis</i> , as discussed later ).	amygdala	<i>stria terminalis</i>	1.0	<a href="#">View file</a>	<span style="background-color: blue; color: white;">NS</span>

**Fig. 7** Web application for connectivity extraction depicting the brain region search query, choice of algorithms, and search results for the query “Amygdala”, using the algorithm “Conn Word Clustering”. Last column in the search result represents Ground Truth (GT) to be marked by a human. By default, GT value has been set to Not Seen

(NS) indicating that human annotator has not yet processed this sentence to validate the result. On inspecting the connection between brain region pair in the sentence, GT can be set as ‘Connected’ (C) or ‘Not Connected’ (NC)

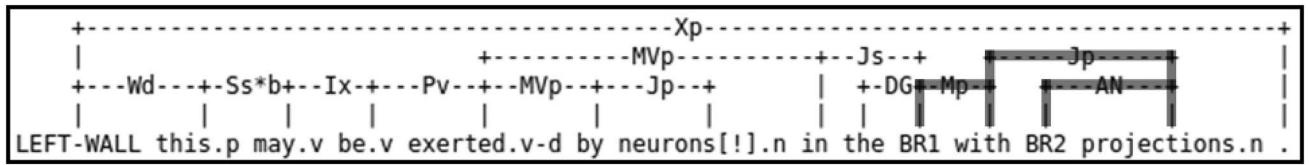
## 5.2 Search Application

We have developed a web-based application<sup>1</sup> using the methods proposed in the paper to serve as a simple search engine for finding and extracting brain region connectivity from large text repository of neuroscience research articles. Clustering-based connectivity extractor algorithms proposed in the paper and additional methods were packaged as part of this tool. Clustering based connectivity extraction algorithms that use supervised learning with different feature representation schemes and additional tools including methods like regular expression pattern match and checking

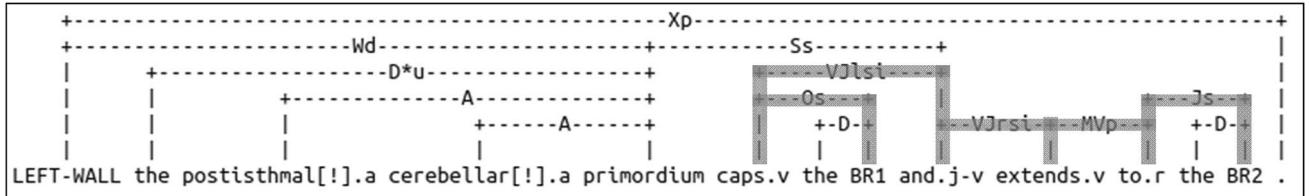
presence of connectivity words have been packaged as part of the application.

The web-based application provides brain region-based search facility by finding and highlighting sentences in the repository that match the queried region and also fetches *connected* brain regions as found by each of the algorithms. This application can play an important role in “explaining” the classifier results by identifying and displaying the relation pattern(s) that triggered extraction of connectivity between brain regions in the search result. A screenshot of the tool is depicted in Fig. 7. Tool also provides a facility to annotate correctness of each query result. This enables the domain expert using the tool to create ground truth for sentences in the repository, which can further be used to optimize the learning model for connectivity extraction.

<sup>1</sup> <http://brainarchitecture.org/text-mining>



**Fig. 8** Sentence from training dataset with SyLF-LPBridge generated bridge pattern marked in bold edges



**Fig. 9** Shortest path in Bridge representation of SyLF-LPBridge algorithm for sentence  $S_3$  from test dataset. Bridge “ $BR1 - caps - and - extends - to - BR2$ ”

Detailed analysis of experiments in Sect. 4.1 has been presented in the following two sub-sections.

### 5.3 Interpretability of SyLF-LPBridge Classifier

In all our techniques, sentences in corpus have been abstracted into a set of patterns. Advantage of proposed approach is it relies on these abstract patterns, which are more interpretable, unlike SVM, Kernel-based methods, or neural network models. Though syntactic, the approach has certain elements of semantics still exposed and facilitates explanation of results as to why a pair of brain regions is *connected* and why a certain pattern of connection emerged. For instance, consider the bridge pattern generated from sentence shown in Fig. 8,  $['Mp', 'br1', 'with', 2], ['Jp', 'with', 'n', 2], ['AN', 'br2', 'n', 3]$ , representing the connectivity pattern “ $BR1 - with - projections(noun) - BR2$ ”. This is a pattern generated during training with confidence score of 0.98. Manual inspection of this relation pattern shows it is quite relevant in terms of extracting connection between the brain region mentions.

Patterns induced by *SyLF-LPBridge* method can be inspected and filtered by domain experts depending on the relevance of the pattern to identify a connection. Effectiveness of domain expert feedback is demonstrated using an example. Consider sentence  $S_3$  from test dataset of *WhiteText*:

“*The postisthmal cerebellar primordium caps the BR1 and extends to the BR2.*”

$S_3$  represents a ‘*Negative*’ instance of brain connectivity, *SyLF-LPBridge* method wrongly identified this sentence as ‘*Positive*’. Inspection revealed that the bridge of sentence  $S_3$  was generated from the shortest path as depicted in Fig. 9.

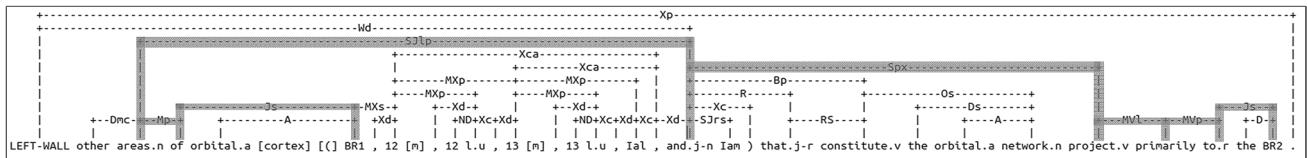
Bridge pattern “ $BR1 - caps - and - extends - to - BR2$ ”, generated for this sentence was not indicative of connectivity between the brain regions. Furthermore, the popular relation pattern from training dataset having maximum similarity with sentence  $S_3$  was inspected. A sentence in the training dataset, say  $S_4$ , generated this popular relation pattern. “*Other areas of orbital cortex (BR1, 12m, 12l, 13m, 13l, Ial, and Iam) that constitute the orbital network project primarily to the BR2.*”

Bridge for the above sentence  $S_4$  is depicted in Fig. 10. Status of  $S_4$  is ‘*Connected*’.  $S_4$  has a similarity of 0.56 with  $S_3$ . Reason for misclassification of  $S_3$  is due to its similarity with the noisy relation pattern generated during training phase by  $S_4$ .

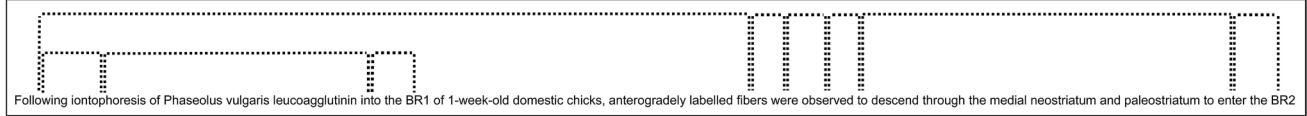
Inspection by a domain expert was carried out to analyze the effect of manual filtering of such relation patterns. Expert confirmed the relation pattern generated by  $S_4$  to be noisy. Therefore pattern of  $S_4$  was deleted from the set of relation patterns generated during training phase. Subsequently, evaluating the test algorithm without the noisy pattern caused the confidence  $S_3$  to drop. This caused the *SyLF-LPBridge* algorithm to classify  $S_3$  correctly as ‘*Negative*’.

This is the advantage of having the learning model generate interpretable patterns. A domain expert could verify and filter the noisy patterns generated during the training phase as a one-time activity, thereby improving performance of the algorithm. Over time, good patterns would be promoted over noisy ones as part of this manual quality check over generated patterns. However, results presented in Tables 1, 2, and 3 are based on algorithm-generated training bridge patterns without any expert supervision.

Proposed models generalize sentences into generic, interpretable pattern representations that can be applied not only



**Fig. 10** Shortest path in Bridge representation of SyLF-LPBridge algorithm for sentence *S4* from training dataset. Bridge “*BR1 – of – areas – and – project – primarily – to – BR2*”



**Fig. 11** Shortest path in Bridge representation of SyLF-LPBridge algorithm for test sentence *S5* from test dataset. Pattern “*Following – iontophoresis – into – BR1 , fibres – were – observed – to – enter – BR2*”, represented by bridge is highlighted in bold

in exactly same contexts but also similar contexts. To classify a new sentence, models obviate the need for making an assessment against every sentence in corpus. Rather an assessment has to be made against a pattern, which is a generalized expression of a cluster of sentences. Once similar patterns are identified, a new sentence is classified based on its extent of similarity to the popular patterns.

#### 5.4 Comparison with BAMS and BioBERT

The Brain Architecture Management system (BAMS) [31] is a comprehensive and manually curated database of brain connectivity information from different species and is used extensively by the neuroscience community for extracting connectivity information. While our extracted connectivity overall agrees with BAMS, we found additional connections that are not listed under BAMS. For e.g., our algorithm extracted that connectivity exists between the primary motor area (MO) and the Nucleus accumbens (AcB).

As mentioned previously, BioBERT [26] is a deep neural network built over BERT architecture [27]. BERT has been trained using masked language modeling objective where certain masked words in a sentence are predicted based on context words in the sequence. This training objective is an attempt to learn the statistical properties of word sequences in the input. Word representation thus learned by BERT has been further fine tuned on a downstream task like Relation Extraction. BioBERT approaches relation extraction as a sentence classification task. An aggregate representation of sentence is constructed from underlying word embedding. This aggregate representation has been used as a feature vector for sentence classification task, irrespective of the relation being extracted and irrespective of the downstream task. BioBERT does not generate any readable relation pattern specific to the

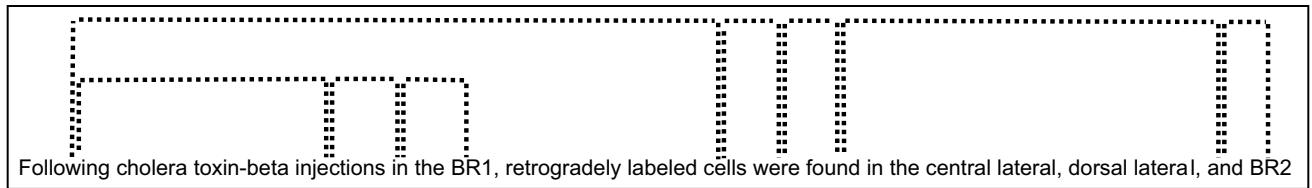
relation being extracted from given corpus. Interpretability of BioBERT model is limited in its ability to explain why a particular sentence has been classified as ‘Connected’ or ‘Not Connected’.

##### 5.4.1 BioBERT error analysis

Analysis of results of BioBERT and SyLF-LPBridge was carried out to understand how each of the methods classify a sentence. Table 1 shows though BioBERT had high precision, its recall was limited on *WhiteText* dataset. False negatives of BioBERT were inspected to find most of the errors occurred in sentences where the two brain regions were far apart within the sentence. Consider sentence *S5* from the test dataset of *WhiteText*:

“*Following iontophoresis of Phaseolus vulgaris leucoagglutinin into the BR1 of 1-week-old domestic chicks, anterogradely labelled fibers were observed to descend through the medial neostriatum and paleostriatum to enter the BR2.*”

*S5* was incorrectly classified as ‘Not Connected’ by BioBERT, but SyLF-LPBridge algorithm identified the same as ‘Connected’. In case of SyLF-LPBridge algorithm, the shortest path in the link parse output of *S5* that defined the bridge representation between ‘*BR1*’ and ‘*BR2*’ is shown in Fig. 11. The path, “*Following – iontophoresis – into – BR1 , fibres – were – observed – to – enter – BR2*”, shows that the Bridge representation clearly captured underlying pattern of connectivity and hence the relation pattern. Classification decision of SyLF-LPBridge algorithm was analyzed by inspecting the most similar training bridge to this sentence. Sentence *S5* had maximum similarity score (0.68) with the generalized training bridge/ popular relation pattern generated from the following sentence, say *S6* from training dataset:



**Fig. 12** Shortest path in Bridge representation of SyLF-LPBridge algorithm for sentence *S6* from training dataset. Pattern “*Following – injections – in – BR1 , cells – were – found – and – BR2*”, represented by bridge is highlighted in bold

“*Following cholera toxin-beta injections in the BR1, retrogradely labeled cells were found in the central lateral, dorsal lateral, and BR2.*”

Corresponding shortest path for *S6* is depicted in Fig. 12. Sentence *S6* is an example of a ‘Connected’ instance in training dataset. Comparison of Bridge representations in Figs. 11 and 12 reveals that, though the two sentences are quite different in length and words used, SyLF-LPBridge was still able to find a high degree of similarity between the two relation patterns. This shows the Bridge representation learned from a positive instance in the training set was able to match sentences with a *similar* bridge in the test dataset and thus classify them as ‘Connected’ despite the *long-range dependency* between ‘*BR1*’ and ‘*BR2*’.

#### 5.4.2 SyLF-LPBridge error analysis

Table 1 shows BioBERT achieved high precision, better than SyLF-LPBridge method, and most other connectivity extraction methods discussed in the literature. In contrast, precision of SyLF-LPBridge was impacted by false positives. Analysis of false positives generated by SyLF-LPBridge method on *2BRWhiteText* dataset revealed the false positives generated by the method fell into certain categories: (i) Sentences with ground truth ‘*Not Connected*’ in test dataset, but had a relation pattern very similar to a ‘*Connected*’ sentence in the training dataset. For instance, consider test sentence, “*Brainstems were cut in sagittal section to facilitate examination of the rostral-caudal extent of BR1 neurons projecting to the BR2*”.

This sentence had a similarity value of 0.68 with the following sentence from the training dataset,

“*The BR1 fibres, which initially project across the whole BR2, withdraw from the caudal part and their terminals become clustered into deep, discrete clumps rostrally*”, which was marked as ‘*Connected*’. But test sentence was originally marked as ‘*Not Connected*’ in the corpus. Given to a human, both these sentences are likely to be assigned same labels. Many sentences were found with such conflicting labels in the corpus. (ii) Sentences that grammatically related *BR1* and *BR2*, but indication of ‘*Connected*’/‘*Not Connected*’ status between them was not clear. For instance consider the sentence, “*The organization of BR1 input to*

the *BR2 will heavily influence the mechanisms by which acoustic information is processed*”. Connection mention in this sentence is ambiguous. (iii) Sentences that depict the existence of weak connections between the brain regions. Example: “*In all three orders, extensive brainstem-spinal pathways were present with only minor representation of BR1 projections originating in BR2*”, “*In addition, the BR1 may project lightly to the BR2*” and “*Staining in the BR1 is generally less dense than in the BR2*”, (iv) Sentences with a negation. Example: “*Choline acetyltransferase-positive cells of BR1 do not retrogradely label from BR2 and are probably cholinergic interneurons similar to those in mammals*”, “*The BR1 does not appear to be a significant source of gamma-aminobutyric acid ergic input to interneurons in the BR2*” and “*No BR1 projections were found to the BR2*”. These sentences had a negation to the mentioned connection by use of words ‘*not*’/‘*no*’.

SyLF-LPBridge had misclassified the above sentences as positive. However, closer inspection of the categories of misclassification revealed the following. In case of sentences with conflicting labels, the algorithm was correct in predicting a ‘*Positive*’ label for the test sentences, remaining in sync with the training samples, and the corresponding relation patterns. Weak connections and ambiguous sentences required extra knowledge and could not be resolved unless an expert human looks at it. Moreover, the classifier was not trained to classify a weak/ambiguous connection. Sentences with negation are the difficult ones which were not handled by SyLF-LPBridge.

Comparing with BioBERT revealed, out of the total sentences handled by SyLF-LPBridge in the *2BRWhiteText* dataset, around 28% were correctly predicted by BioBERT, but misclassified by SyLF-LPBridge. *2BRWhiteText* dataset mostly contains short and crisp sentences that were captured well by BioBERT. In addition, the high dimensional representation captures the context of connection in a better manner, thereby achieving the maximum  $F_1$  score, while SyLF-LPBridge still managed the best  $F_2$  score due to high recall on the *2BRWhiteText* dataset. However, long-range dependencies and complex negations are a challenge for BioBERT as well.

Further, BERT is a compute-heavy model and requires a Graphics Processing Unit (GPU) for fine-tuning. BERT

architecture is exposed as two models, BERT Base with 110M parameters and BERT Large with 340M parameters. Due to the large model footprint of BERT, both training and inference time is more than that of simpler models. In the recent past, lighter variants of BERT like Tinybert [32], ALBERT [33], DistilBERT [34] etc., have been proposed and can be explored further.

BERT uses transformers [35] with multi-headed attention and is known to perform better than the previous state-of-the-art methods in many NLP tasks [27]. However, questions have been raised about the interpretation of self-attention patterns in BERT by analyzing BERT's capacity to capture linguistic information in input [36]. With the fast-paced advancement in BERT and its variations, one can hope to have solutions to address these concerns in the near future.

On one hand, modeling of long-range dependencies in language models is a challenge due to memory restrictions. In contrast, SyLF-LPBridge method tries to model such dependencies in a more principled manner by resorting to grammatical parse structure of sentence. On the other hand, BioBERT is pre-trained on PubMed, which is a large resource of scientific articles in biomedical domain. BioBERT leverages this vast amount of background knowledge, which is not available to SyLF-LPBridge. However, proposed methods (SyLF-BoW and the SyLF-LPBridge) were designed to be domain agnostic and could be used to extract any binary relation in a given corpus. Performance of these methods could be improved by domain-specific tuning in the way PubMed was used for pre-training BioBERT. Though SyLF-LPBridge generates rich, interpretable syntax-based representations, it is depleted in terms of such bottom-up knowledge available to BioBERT. Therefore, BioBERT and proposed methods could be used to complement each other with high dimensional representation and the ability to explain results respectively.

## 5.5 Perspectives for Further Research

The Connections retrieved need to be compared with a standard database like the Brain Architecture Management System (BAMS) ([31]). BAMS is a manually curated database of connections predominantly for the rat brain. An attempt was made to compare the connections retrieved by our classifiers with connections in BAMS. Lack of normalization of brain region names between the mined literature and BAMS posed obstacles in comparing results of the automated text mining methods with that of the manually curated standard database.

BAMS was originally curated using the Swanson-98 [37] and Swanson-04 [38] nomenclature. Whereas, connections in literature are authored using Swanson (1998, 2004) [37] [38], Paxinos (2014) [39], and several other nomenclatures. For example, the *accumbens* (ACb) can appear as

*accumbens*, *nucleus (of) accumbens*, *accumbens nucleus*, or even *ventral striatum*. On the other hand, “*ventral striatum*” may mean more than *accumbens* in specific contexts.

Our current algorithm does not consider negation for further processing. The main reason for it is reporting of negative relationship, in this case connectivity, is rare and in literature often is cited as lack of evidence rather than comprehensively proving that such a relationship does not exist. This negative parsing requires the identification of the negation and then further parsing the sentence and is considered future work in expanding the algorithm.

Automatic identification of species and dominant nomenclature adopted by the author of the article is a necessary step before attempting connectivity extraction. Named Entity Recognition (NER) and Named Entity Disambiguation (NED) have been critical components of this effort and solutions in this space need to be explored for improving and evaluating brain connectivity extraction techniques. All approaches presented in this paper assume that entities are already marked up.

Ontologies like the BAMS Ontology ([40]) describe the structural parts organized into a *part-of* and *is-a* hierarchy. Inferring connectivity by leveraging hierarchy defined in such ontologies is a potential work to be done in this field.

Another challenge was to extract the direction of a connection. Gokdeniz et al. [13] exploit the predefined pattern template to identify agent/target and hence the direction of the connection. In this case, algorithms automatically induce the pattern template and hence further processing would be required to identify the agent and target entities.

Feature space of SyLF-LPBridge method can be evolved to improve its precision in case of handling complex sentences with more number of brain regions. Differently weighing features based on domain knowledge, part-of-speech(PoS) tags etc., in the feature representation of SyLF-LPBridge can potentially be explored.

## 6 Conclusions

In this paper, we have proposed three Instance-based learning algorithms using different feature representation schemes. In addition, we have explored surface-level statistical features and rich syntactic features using linguistic information. The performance of three proposed classifiers and a combination method have been compared and contrasted with previous state-of-the-art techniques for neuronal connectivity extraction. We have also compared our methods with the neural network-based transfer learning method, BioBERT. Although transfer learning used in BioBERT has the advantage of using knowledge gained during pre-training on large amounts of background data, it does not provide an

interpretable model in terms of relation patterns extracted from our corpus.

On the other hand, the proposed SyLF-LPBridge method performs deep syntactic analysis of input and induces human-readable patterns from text which can be inspected and potentially improved based on expert feedback and is the main focus of the paper. Our application is recall oriented and  $F_2$  metric is a good performance indicator that optimizes recall while not sacrificing greatly on precision. SyLF-LPBridge method achieved the best  $F_2$  score on all the datasets in spite of not being powered by a background resource such as PubMed. Despite the higher precision, BioBERT may not be very appropriate for the given recall-oriented task due to its low  $F_2$  score.

The goal of this work has been to aid in the collation of experimental results about brain region connectivity that has already been published in various forums, into a structured database. Among the range of proposed techniques, some methods achieve better recall, and some better precision. We have attempted to balance the two as the default setting for the application. However, users of the system can tune the parameters to control precision and recall according to their needs. Our proposed methods, along with the web application would help neuroscientists to automatically compile brain connectivity results and enable quick validation of wet lab experiments with the latest work done by the community. As mentioned earlier, the paper does not aim to replace the deep learning-based approaches, instead, BioBERT and proposed methods can be used to complement each other with features like high dimensional representation (of the neural networks) and the ability (of the syntax-based method) to explain results of *Connectivity* extraction.

**Supplementary Information** The online version contains supplementary material available at <https://doi.org/10.1007/s12539-021-00443-6>.

**Funding** Not applicable.

**Data Availability** Models generated on the benchmark corpus White-Text, Supplementary material including the datasets are made available at <https://github.com/ashika55/BRConnExt>.

## Declarations

**Conflict of interest** On behalf of all authors, the corresponding author states that there is no conflict of interest.

**Code availability** Software application, ‘ConnExt1’ can be accessed at <http://brainarchitecture.org/text-mining>.

## References

- Canese K, Weis S (2013) Pubmed: the bibliographic database. In: The NCBI Handbook [internet]. 2nd edition, National Center for Biotechnology Information (US), <https://www.ncbi.nlm.nih.gov/sites/books/NBK153385/>
- Sporns O (2011) The human connectome: a complex network. Annals of the New York Academy of Sciences 1224(1):109–125. [https://doi.org/10.1016/S0920-9964\(12\)70100-7](https://doi.org/10.1016/S0920-9964(12)70100-7)
- Richardet R, Chappelier JC, Telefont M, Hill S (2015) Large-scale extraction of brain connectivity from the neuroscientific literature. Bioinformatics 31(10):1640–1647. <https://doi.org/10.1093/bioinformatics/btv025>
- French L, Lane S, Xu L, Siu C, Kwok C, Chen Y, Krebs C, Pavlidis P (2012) Application and evaluation of automated methods to extract neuroanatomical connectivity statements from free text. Bioinformatics 28(22):2963–2970. <https://doi.org/10.1093/bioinformatics/bts542>
- French L, Liu P, Marais O, Koreman T, Tseng L, Lai A, Pavlidis P (2015) Text mining for neuroanatomy using whitetext with an updated corpus and a new web application. Front Neuroinform 9:13. <https://doi.org/10.3389/fninf.2015.00013>
- Künzle H, Radtke-Schuller S (2000) Basal telencephalic regions connected with the olfactory bulb in a madagascan hedgehog tenrec. J Comparative Neurol 423(4):706–726
- Hobbs JR (2002) Information extraction from biomedical text. J Biomed Inform 35(4):260–264. [https://doi.org/10.1016/S1532-0464\(03\)00015-7](https://doi.org/10.1016/S1532-0464(03)00015-7)
- Tikk D, Thomas P, Palaga P, Hakenberg J, Leser U (2010) A comprehensive benchmark of kernel methods to extract protein-protein interactions from literature. PLoS Comput Biol 6(7):e1000837. <https://doi.org/10.1371/journal.pcbi.1000837>
- Wu HY, Chiang CW, Li L (2014) Text mining for drug–drug interaction. In: Biomedical Literature Mining, Springer, pp 47–75, 10.1007/978-1-4939-0709-0\_4
- French L, Lane S, Xu L, Pavlidis P (2009) Automated recognition of brain region mentions in neuroscience literature. Front Neuroinform 3:29. <https://doi.org/10.3389/neuro.11.029.2009>
- Giuliano C, Lavelli A, Romano L (2006) Exploiting shallow linguistic information for relation extraction from biomedical literature. In: 11th Conference of the European Chapter of the Association for Computational Linguistics, <https://www.aclweb.org/anthology/E06-1051>
- Kluegl P, Toepper M, Beck PD, Fette G, Puppe F (2016) Uima ruta: rapid development of rule-based information extraction applications. Nat Lang Eng 22(1):1–40. <https://doi.org/10.1017/S1351324914000114>
- Gökdeniz E, Özgür A, Canbeyli R (2016) Automated neuroanatomical relation extraction: a linguistically motivated approach with a pvt connectivity graph case study. Front Neuroinform 10:39. <https://doi.org/10.3389/fninf.2016.00039>
- Künzle H (1998) Thalamic territories innervated by cerebellar nuclear afferents in the hedgehog tenrec, *echinops telfairi*. J Comparative Neurol 402(3):313–326. 10.1002/(SICI)1096-9861(19981221)402:3%3c313::AID-CNE3%3e3.0.CO;2-E
- Agichtein E, Gravano L (2000) Snowball: Extracting relations from large plain-text collections. In: Proceedings of the fifth ACM conference on Digital libraries, ACM, pp 85–94, 10.1145/336597.336644,
- Sleator DD, Temperley D (1995) Parsing english with a link grammar. arXiv preprint cmp-lg/9508004 <https://www.aclweb.org/anthology/1995.iwpt-1.22>
- Sleator DD, Temperley D (Website) Index to link grammar documentation. <https://www.abisource.com/projects/link-grammar/dict/index.html>
- Groenewegen HJ, Berendse HW (1990) Connections of the subthalamic nucleus with ventral striatopallidal parts of the basal ganglia in the rat. J Comparative Neurol 294(4):607–622. <https://doi.org/10.1002/cne.902940408>

19. Frakes WB (1992) Information retrieval: data structures & algorithms. Pearson Education India, DOI 10(1145/182119):1096164
20. Suchanek FM, Ifrim G, Weikum G (2006) Leila: Learning to extract information by linguistic analysis. In: Proceedings of the 2nd Workshop on Ontology Learning and Population: Bridging the Gap between Text and Knowledge, pp 18–25, <https://www.aclweb.org/anthology/W06-0503>
21. Wagner RA, Fischer MJ (1974) The string-to-string correction problem. J ACM (JACM) 21(1):168–173. <https://doi.org/10.1145/321796.321811>
22. Grinberg D, Lafferty J, Sleator D (1995) A robust parsing algorithm for link grammars. arXiv preprint cmp-lg/9508003 <https://www.aclweb.org/anthology/I95.iwpt-1.15>
23. Dong HW (2008) The Allen reference atlas: A digital color brain atlas of the C57Bl/6J male mouse. John Wiley & Sons Inc, 10.1111/j.1601-183x.2009.00552.x
24. Sharma A, Sharma A, Deodhare D, Chakraborti S, Kumar PS, Mitra PP (2016) Case representation and retrieval techniques for neuroanatomical connectivity extraction from pubmed. In: International Conference on Case-Based Reasoning, Springer, pp 370–386, 10.1007/978-3-319-47096-2\\_25
25. Schütze H, Manning CD, Raghavan P (2008) Introduction to information retrieval. In: Proceedings of the international communication of association for computing machinery conference, vol 4, 10.1017/CBO9780511809071
26. Lee J, Yoon W, Kim S, Kim D, Kim S, So CH, Kang J (2019) Biobert: pre-trained biomedical language representation model for biomedical text mining. arXiv preprint arXiv: 190108746. 10.1093/bioinformatics/btz682
27. Devlin J, Chang MW, Lee K, Toutanova K (2018) Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:181004805 <https://arxiv.org/abs/1810.04805>
28. Zhu Y, Kiros R, Zemel R, Salakhutdinov R, Urtasun R, Torralba A, Fidler S (2015) Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In: Proceedings of the IEEE international conference on computer vision, pp 19–27, 10.1109/ICCV.2015.11
29. Lee J, Yoon W, Kim S, Kim D, Kim S, So CH, Kang J (Websiteb) Biobert model. <https://github.com/naver/biobert-pretrained>
30. Lee J, Yoon W, Kim S, Kim D, Kim S, So CH, Kang J (Websitea) Biobert codebase. <https://github.com/dmis-lab/biobert>
31. Bota M, Dong HW, Swanson LW (2005) Brain architecture management system. Neuroinformatics 3(1):15–47. <https://doi.org/10.1385/NI:3:1:015>
32. Jiao X, Yin Y, Shang L, Jiang X, Chen X, Li L, Wang F, Liu Q (2019) Tinybert: Distilling bert for natural language understanding. arXiv preprint arXiv :190910351. 10.18653/v1/2020. findings-emnlp.372
33. Lan Z, Chen M, Goodman S, Gimpel K, Sharma P, Sororic R (2020) Albert: A lite bert for self-supervised learning of language representations. In: International conference on learning representations, <https://openreview.net/forum?id=H1eA7AEtvS>
34. Sanh V, Debut L, Chaumond J, Wolf T (2019) Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. arXiv preprint arXiv: 191001108
35. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, Polosukhin I (2017) Attention is all you need. In: Advances in neural information processing systems, pp 5998–6008, 10.5555/3295222.3295349
36. Kovaleva O, Romanov A, Rogers A, Rumshisky A (2019) Revealing the dark secrets of bert. In: Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP), pp 4356–4365, 10.18653/v1/D19-1445
37. Swanson L (1998) Structure of the rat brain: a laboratory guide with printed and electronic templates for data, models and schematics. Brain maps: Structure of the Rat Brain, 2nd Edn Amsterdam: Elsevier Science p pp 17–30, <https://searchworks.stanford.edu/view/4106941>
38. Swanson L (2004) Brain maps : structure of the rat brain : a laboratory guide with printed and electronic templates for data, models and schematics. Brain Maps: Structure of the Rat Brain, 3rd Edn Amsterdam: Elsevier <https://searchworks.stanford.edu/view/4106941>
39. Paxinos G, Watson C (2014) The rat brain in stereotaxic coordinates: hard cover edition. Elsevier, 10.1016/c2009-0-63235-9
40. Bota M, Swanson LW (2008) Bams neuroanatomical ontology: design and implementation. Front Neuroinform 2:2. <https://doi.org/10.3389/neuro.11.002.2008>

## Authors and Affiliations

**Ashika Sharma<sup>1,2</sup> · Jaikishan Jayakumar<sup>3</sup> · Partha P. Mitra<sup>4</sup> · Sutanu Chakraborti<sup>2</sup> · P. Sreenivasa Kumar<sup>2</sup>**

<sup>✉</sup> Ashika Sharma  
ashika@cse.iitm.ac.in

<sup>1</sup> Center for Artificial Intelligence and Robotics, DRDO Complex, Bangalore 560093, India

<sup>2</sup> Department of Computer Science and Engineering, Indian Institute of Technology Madras, Chennai 600036, India

<sup>3</sup> Center for Computational Brain Research, IIT Madras, Chennai 600036, India

<sup>4</sup> Cold Spring Harbour Laboratory, Cold Spring Harbour, New York 11724, USA