Exploring the Effects of Different Embedding Algorithms and Neural Architectures on Early Detection of Alzheimer's Disease

Minni Jain, Rishabh Doshi, Vibhu Sehra and Divyashikha Sethia

Department of Computer Engineering, Delhi Technological University, Delhi, India

Abstract

Alzheimer's Disease (AD) is an irrecoverable, progressive neurodegenerative disorder that deteriorates the cognitive and linguistic abilities of a person over time. Ample research has been done on the early detection of AD; it remains a challenging task. Doctors use the patient's history, laboratory tests, and change in behaviour to diagnose the disease. Natural Language Processing(NLP) techniques can help automate the detection of AD, as Language impairments accompany this disease. This work aims to analyze the effect of different Embedding models on the DementiaBank dataset in order to detect the disease. The work uses both Generic and domain-specific Word Embeddings on the three deep learning models - CNN, Bidirectional LSTM(BLSTM), and CNN+BLSTM. Results indicate that for a specific picture description task like cookie theft description, domain-specific Word Embeddings tend to work better. Lastly, it is discussed how results are affected by the use of different Embedding models (Fasttext, Word2Vec, GloVe).

Keywords

Alzheimer's Disease, Natural Language Processing, Word Embeddings, Deep Learning, Cookie theft Description task

1. Introduction

Alzheimer's Disease(AD) is a brain disorder that slowly damages the nerve connections in the Brain. It is the most common type of dementia and symptoms of AD include communication difficulties, memory loss, poor judgment, and changing mood and personality[1]. More than 50 million people are diagnosed with Alzheimer's Disease every year [2]. This challenge has grown substantially over the years with the ageing of the population and the agerelated nature of many dementia-producing neurodegenerative diseases [1]. This number of cases for Alzheimer's Disease will continue to grow in the coming years. There is no proven health care method to cure AD. Hence, it is necessary to develop a new method to detect AD in a patient. Around 50 to 90% of dementia cases are left undiagnosed by standard clinical examinations [1]. Early detection of Alzheimer's Disease is still a massive issue in the current scenario. Alzheimer's Disease progresses over the years, and sometimes patients can have the disease for 20 years before showing symptoms. At this point, medical treatment is not very useful after the diagnosis of the disease. Hence the early detection of Alzheimer's is still a challenge in medical science. There have been many attempts to diagnose the disease with the help of neuroimaging techniques, but non-imaging techniques are essential to personalize the treatment for a patient and monitor disease progression. Machine learning can detect the language deficits that often accompany dementia and therefore can be used for ealry detection of Alzheimer's Disease. Previously, many Natural Language Processing (NLP) techniques were proposed to help in early detection of Alzheimer's Disease. These techniques treat the problem as a supervised learning problem. Previous research works like [2, 3, 4] made use of transcripts obtained from interviews with patients to detect Alzheimer's disease by using various machine learning and deep learning algorithms. Further, other studies like [5, 6, 7] used acoustic features obtained from the audio recordings of the interviews for the classification task. Our study aims to explore the effect of various Word Embeddings and neural architectures on transcripts obtained from the cookie theft description task of DementiaBank.

This paper makes use of both generic and domain-specific Word Embeddings that are trained on the transcripts. Out of all the presented models, the CNN + Bidirectional LSTM models that make use of Fasttext domain-specific Word Embeddings provides the best results. Sentences obtained from the transcripts are input to the models, and the output is the predicted label (Healthy or Alzheimer's), no feature engineering

ISIC'2021: International Semantic Intelligence Conference, Feb 25–27, 2021, New Delhi, Delhi , India

minnijain@dtu.ac.in (M. Jain); doshirishabh26@gmail.com (R. Doshi); vibhusehra@gmail.com (V. Sehra); divyashikha@dtu.ac.in (D. Sethia)

[1]https://www.alz.org/alzheimers-dementia/10_signs

[2]https://www.alz.org/alzheimers-dementia/facts-figures

was involved in the process. Hence, this paper investigates how the task of detecting Alzheimer's Disease is affected by the use of various domain-specific and generic Embeddings on different neural architectures.

The rest of the paper comprises section 2, which consists of the Related works followed by our proposed work and experimental setup in sections 3 and 4, respectively. Then we present our results and discussion in sections 5 and 6, respectively, which is followed by the conclusion and future work in section 7.

2. Related Work

This section discusses the previous research, done in the field of Alzheimer's detection using the various machine learning and deep learning techniques.

2.1. Machine Learning Techniques

Existing research found on early detection of Alzheimer's Disease using Natural language processing made use of various machine learning techniques. [8] used three different machine learning algorithms - namely Decision trees, Support Vector Machine, and K-Nearest neighbours on a sample of 80 conversations to achieve the best accuracy of 79.5% using their Decision tree model. [9]proposed a model using Support Vector machine making use of 14 lexical features, nine syntactic features, and n-grams extracted from the Pitt Corpus in Dementia Bank Dataset by using 99 dementia transcripts and 99 control transcripts from the dataset. They used Area Under Curve (AUC) metric to test the performance of the algorithm achieving a maximum AUC score of 0.93 by using the top 1000 features obtained using a Leave Pair Out Cross-Validation (LPOCV) crossvalidation technique.

Further, [7] used the DementiaBank dataset to extract the acoustic measures and semantic measures to predict the clinical scores of the patients by making use of the bivariate dynamic Bayes network. [5] extracted acoustic features from the DementiaBank dataset and created a regression model to predict clinical scores (MMSE) used for dementia prediction. [6] made use of acoustic features on various Machine Learning models like Logistic Regression, KNN, Naive Bayes, Dummy classifier, Random Forests, and achieved the best accuracy of 78% with Logistic regression classifier.

2.2. Deep Learning Techniques

[10] had made use of Deep-Deep neural networks and had achieved an accuracy of 87.5% using the sparse vector representations of 4, 5 n-grams. The dataset was equally divided by making use of 99 dementia transcripts and 99 control transcripts from the dataset. Recently, [2] proposed the use of 3 different deep learning algorithms- 2D-CNN, LSTM, and 2D CNN - RNN models by making use of the complete Dementia bank dataset which consists of 1017 Alzheimer's transcripts and 243 control transcripts. They used each utterance as a separate data sample, therefore obtaining 14362 utterance samples. They achieve the best accuracy of 91.1% using the CNN-RNN model by using Word Embeddings along with POS tagged data to the classifier. [3] used a Hierarchical attention network (HAN) on the transcripts obtained from DementiaBank Dataset. They made use of Word Embeddings along with demographic features for the prediction task obtaining an accuracy of 86.9%. [11] proposed a model that combined bidirectional hierarchical recurrent neural network with an attention mechanism for dementia detection. [12] showed that fine tuned BERT model outperformed the models that used hand crafted feature engineering. Table.4 summarizes the approach used by previous research works.

3. Proposed Work

3.1. Preprocessing

This work uses the transcripts in the Dementia Bank dataset [13], which are available in the form of CHAT transcription [14]. The transcripts are passed through a series of steps as given below and illustrated in Fig. 1. PyLangAcq library [15], which is a powerful library that can handle CHAT data, reads the transcripts. We then convert all obtained utterances to lower text and remove all punctuations. We use 99 transcripts from each set (Dementia and Control) from the Cookie Theft task as suggested by [9, 10] where they made use of an equal number of dementia and control patients.

3.2. Word Embeddings used for early detection of Alzheimer's Disease

This work uses three types of Word Embeddings- Word2Vec [16], Glove [17] and, Fasttext [18]. These embeddings are chosen because they are widely used and have different architechtures which may tell us the best way to proceed with the problem in hand. All the Word Embeddings have a 300-dimensional vector representation for each Word. For each of the types mentioned above, two-Word Embeddings are used, Domain-specific and generic Word Embeddings. All the transcripts from DementiaBank are used to create the domain specific Word Embeddings stated above. The maximum size of a transcript was 498 words. Hence, we keep the size of the Word Embedding as (500,300).

3.2.1. Domain-Specific Word Embeddings

Domain-Specific Word Embeddings are Embeddings that are trained on a specific corpus that contains data from the interested domain. They are highly effective for a specific domain but require extra training time. Gensim library [19] is used to create Word2vec [16] and Fasttext [18] Word Embeddings from the corpus. Glove[3] library is used to create the GloVe Embeddings [17].

3.2.2. Generic Word Embeddings

Generic Word Embeddings are Embeddings that are trained on vast generic corpora. Hence these Embeddings reduce training time and often give outstanding results. The work trains the pretrained Glove [17] Embeddings on 6 billion words. It trains Word2vec Embedding, which includes word vectors for a vocabulary of 3 million words and phrases on roughly 100 billion words from a Google News dataset. It also trains Fasttext [18] Embedding, which contains vectors for 1 million words, on Wikipedia 2017, UMBC web base corpus, and statmt.org news dataset having a total of 16 billion tokens.

3.3. Deep Learning Models Used

This section explains the deep learning models that are used for the classification of control and dementia patients. Keras functional API [20] is used to create all the deep learning models explained below. To address the concern of overfitting, we use L2 regularizer [21] as the kernel initializer. Due to the small size of the dataset, the research makes use of 10-fold cross-validation on each model. The model atempts to capture the language impairments that are often seen in the ealry phases of dementia. The Annexure provides the details of the model architecture.

3.3.1. CNN Model

In this work, the CNN model consists of a combination of 1DConvolution layers with an increasing number of kernels followed by MaxPool layers. A Dense network follows this. We use the Tanh activation for the 1D Convolution layer, ReLU [22] as the activation function for the Dense layers, and Softmax for classification.

3.3.2. Bi-Directional LSTM Model

The model has a series of the Bidirectional LSTM layer and Dropout [23] layer; further layers consist of a Dense network for classification. The Dropout layers are added to prevent overfitting in the model and dropout rate is kept at 30%. All the layers use default Tanh activation except the last one, which uses Softmax for classification.

3.3.3. Hybrid CNN + Bi-Directional LSTM Model

This model is a combination of the above two models. We pass the Embeddings through a series of 1D-convolutional layers followed by a MaxPooling layer, with two bidirectional LSTM layers stacked over the Maxpool layer. A dense network follows this. Fig. 2. illustrates the proposed model. The Activations used for CNN and bidirectional LSTM is Tanh, while we use ReLU [22] activation for dense layers followed by a SoftMax function for classification.

3.4. Training Details

The above-stated models are trained using the Adam Optimizer [24] for 30 epochs, each using Binary cross-entropy as the loss function. L2 regularization [21] is applied in each layer has $\lambda = 10^{-5}$

4. Experimental Details

This work uses Pitt Corpus, which is the largest English dataset available in DementiaBank [13]. DementiaBank is a part of the TalkBank project initiated by Carnegie Mellon University. The National Institute of Aging funds it. The project encourages research for human communication. It uses the Codes for the Human Analysis of Transcripts (CHAT) system [14], which provides automatic analysis and testing. The CHAT system is commonly used in many datasets to provide uniformity and easy usage. Various participants from each group (Control and dementia) visited annually for the interview. Pitt Corpus [13] is a collection of transcripts and audio files that were collected as a part of a longitudinal study conducted by Alzheimer's and Related dementia at the University of Pittsburgh School of Medicine. This dataset contains interviews

[3] https://github.com/JonathanRaiman/glove

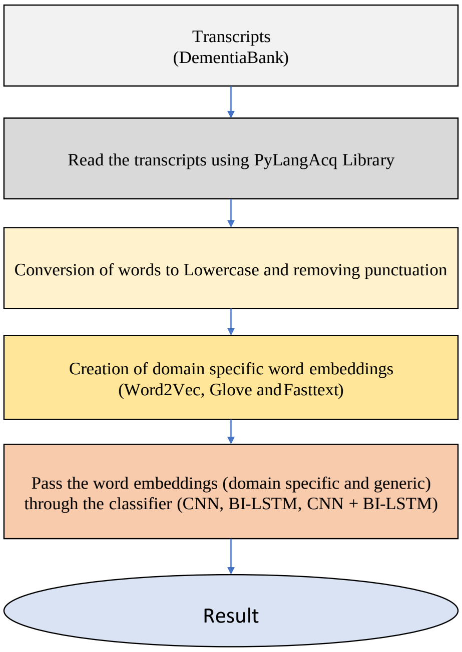

Figure 1: Proposed Approach for early detection of Alzheimer's Disease

of patients with possible Alzheimer's along with control patients, containing transcripts of 104 control patients and 208 dementia patients. The patient's ages range from 49-90 years in the dataset. It comprises of four different tests on the patients:

- Cookie Theft: Patients see an image provided by the Boston Diagnostic Aphasia Examination, and then the patients (Control and Dementia) recall the events taking place in the image (Fig. 3).
- Fluency: This task is done only for dementia patients where they respond to a word Fluency task.
- Recall: The Dementia Patients undergo a story recall test.
- Sentence: The Dementia Patients perform a Sentence construction task.

The work uses the Cookie theft part of the corpus as it contains the maximum number of participants, and previous researchers have used it.

5. Results

All the three neural models - 1D CNN, Bidirectional LSTM(BLSTM), and 1D CNN + Bidirectional LSTM (CNN + BLSTM) use the generic and domain-specific Word Embeddings of each Embedding model. For domain-specific Word Embeddings, we achieved maximum accuracies of 89.9%, 85%, and 90.6% with Fasttext Embedding for CNN, BLSTM, and CNN + BLSTM models, respectively. While for pre-trained Word Embeddings, maximum accuracies obtained were 85.2% with Glove for both CNN and BLSTM, and 85.5% with Fasttext for CNN + BLSTM. The baseline model used is constant label classifier which gives the same result for any input which achieved an accuracy of 50% since we have two classes. Tables 1, 2, and 3 summarize the results obtained by using the three Embedding models (Glove, Word2Vec, Fasttext) for the three given deep learning models. Fig. 4. compares the F1 scores achieved by these models which makes clear that Domain Specific Fasttext embeddings outperform all the other embeddings.

Accuracy, precision, recall, and F1-score are used as the evaluation metrics. Previous works using deep learning techniques such as [2] used accuracy, [10] used AUC (Area Under Curve), and [3] used precision, recall beddings was better than that of Generic Word-Embeddings. The and F1 score as the evaluation metrics. The results obtained show probable causes are discussed further in the next section. that generally, the performance of the domain-specific Word Embeddings was better than that of Generic Word Embeddings. The probable causes are discussed further in the next section.

6. Discussions

The paper aims to explore how the different Word Embedding models and types of Embeddings perform on different neural models. It uses both the domain specific and the generic Word Embeddings to classify the transcripts. However, since the domain-specific Word Embeddings have been trained on the same corpus being used, it generally provides better results. As the cookie theft data comprises of explaining a particular image, the vocabulary found in the transcripts is limited, and as a result, it is easier to understand the relationship between words. Using Domain-specific, Fasttext, and Word2vec provides better results than their Generic counterparts. Results indicate that Glove Embeddings provide similar results on both types of Word Embeddings.

If we had a combination of different tasks (not only cookie theft) having a larger corpus and vocabulary, Generic Embedding might perform better.

Results indicate that Word2vec has the lowest ac-

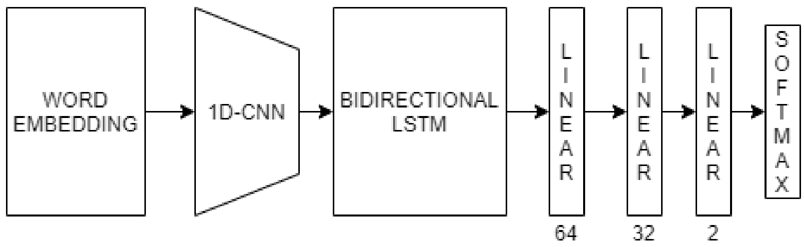

Figure 2: Pictorial Representation of the CNN+BLSTM used

Table 1
Results obtained for the CNN model

Word Embedding	Accuracy	Precision	Recall	F1-score
Fasttext				
Generic	0.85	0.86	0.85	0.85
Domain-specific	0.90	0.92	0.90	0.91
GloVe				
Generic	0.85	0.85	0.85	0.85
Domain-specific	0.83	0.83	0.81	0.82
Word2Vec				
Generic	0.77	0.78	0.77	0.77
Domain-specific	0.80	0.80	0.80	0.80

curacy amongst the three Embedding models. This is possible because domain-specific Word2vec requires a larger corpus to develop the semantic relation as it only captures local word relations. The domain specific Fasttext Embedding gives the best result since it does not require a large corpus as it breaks each word into character n-grams, thereby increasing the vocabulary size.

Results also indicate that the hybrid CNN + BLSTM model achieves the highest accuracy of 90.6%. The CNN + BLSTM model works better than any single use of either of the model, because:

- CNN model captures the short-term dependencies in text.
- LSTM model captures long term dependencies in the text. Bidirectional LSTM is better than the LSTM as it trains on two LSTM cells instead of one cell in a single input sequence.

Compared to similar previous works like [2] and [3] use a Word Embeddings layer that is trained along with the neural architecture, this study uses three Word Embedding models and from each Embedding model, a domainspecific and pre-trained Embedding is created to identify how different Embedding models and the type of data on which the Embeddings are trained

Table 2
Results obtained for the BLSTM model

Word Embedding	Accuracy	Precision	Recall	F1-score
Fasttext				
Generic	0.80	0.85	0.80	0.82
Domain-specific	0.85	0.86	0.85	0.85
GloVe				
Generic	0.85	0.88	0.85	0.86
Domain-specific	0.84	0.85	0.84	0.84
Word2Vec				
Generic	0.74	0.75	0.74	0.74
Domain-specific	0.80	0.80	0.80	0.80

Table 3
Results obtained for the CNN+BLSTM model

Word Embedding	Accuracy	Precision	Recall	F1-score
Fasttext				
Generic	0.86	0.86	0.85	0.85
Domain-specific	0.91	0.91	0.91	0.91
GloVe				
Generic	0.84	0.85	0.83	0.84
Domain-specific	0.87	0.88	0.87	0.87
Word2Vec				
Generic	0.77	0.79	0.78	0.78
Domain-specific	0.80	0.80	0.80	0.80

Table 4
Comparision of proposed work with results and techniques of existing work

Author	Accuracy	Model	Technique
Orimaye et al. (2018) [10]	87.5%	Neural Network	4-5 n-grams
Karlekar et al. (2018) [2]	82.8%	2D-CNN	Word Embeddings
Karlekar et al. (2018) [2]	83.7%	RNN	Word Embeddings
Karlekar et al. (2018) [2]	91.1%	2D-CNN + RNN	Word Embeddings along with POS tagged data
Kong et al. (2019) [3]	86.9%	Hierarchical Attention Network	Word Embeddings
Proposed work	90.6%	1D-CNN + BLSTM	Doamin-Specific Fasttext Word Embedding

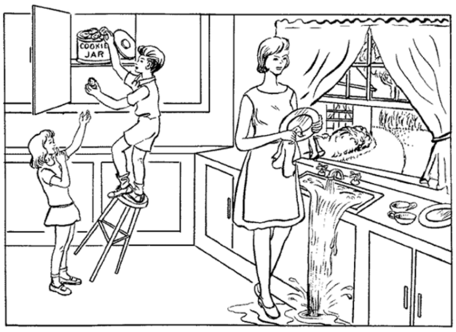

Figure 3: Boston cookie theft description task

affects the performance of detecting Alzheimer's Disease. [2] breaks down each transcript into utterances and considers them as separate data samples thereby creating 14362 samples as compared to our 198 samples which are complete transcripts of a patient.

7. Conclusion and Future Work

This study employs three Word Embedding algorithms on three different Neural Models that make use of CNN and Bidirectional LSTM for Alzheimer's Disease Classification. For each word embedding algorithm 2 different types of word embeddings were used - Domain Specific and Generic Embeddings, where it was found that Domain Specific word embeddings performed better than Generic Word Embeddings. This work was limited by the small amount of dataset available. In future, we may gather a larger dataset that may help in creation of a more generalized embedding. Further, we can also extend the dataset for people speaking different languages.

A. Appendix

A.1. Neural Model Details

We used the following neural models. The batch size was kept at 10. In the last dense layer of each model

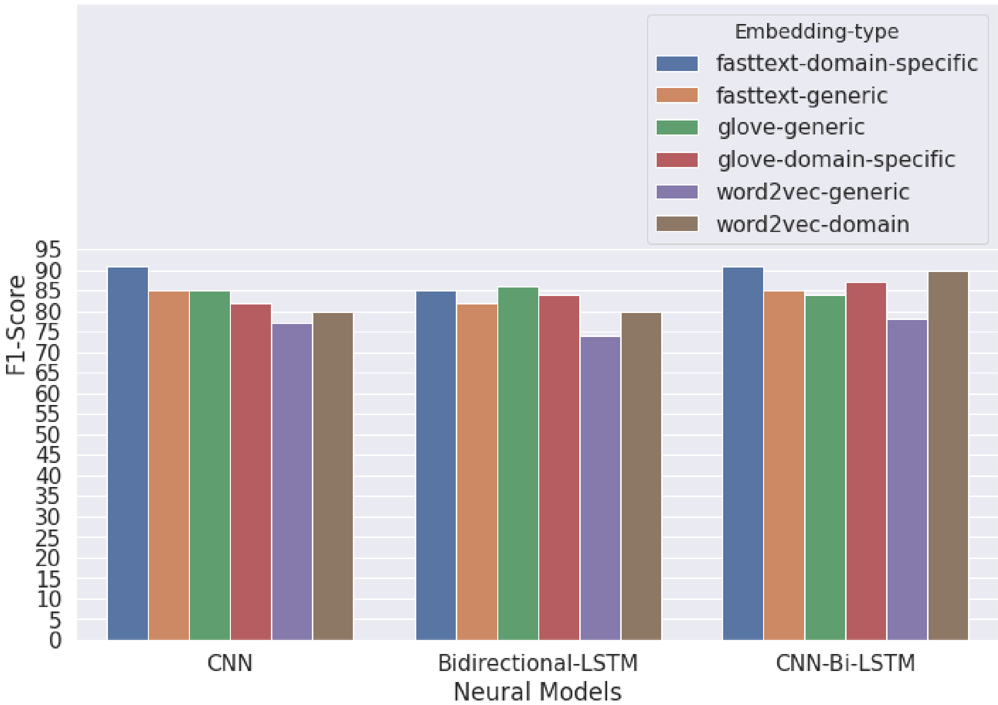

Figure 4: Comparsion of F1-scores achieved by different neural models and Word Embeddings

softmax activation function was used. Other dense layers use a rectified linear activation function.

A.1.1. CNN Model

Each CNN-1D layer in brackets represents(no-of-filters , kernel-size)
CNN-1D(8,3) ⟶ CNN-1D(10,3) ⟶ MaxPool-1D(3) ⟶ CNN-1D(12,3) ⟶ CNN-1D(14,3) ⟶ MaxPool-1D(3) ⟶ Flatten() ⟶ Dense(20,Relu) ⟶ Dense(10,Relu) ⟶ Dense(2,Softmax)

A.1.2. BLSTM

Each LSTM layer in brackets represents(no-of-lstm-cells-in-that-layer)
Bidir(LSTM(16)) ⟶ Dropout(0.3) ⟶ Bidir(LSTM(8)) ⟶ Bidir(LSTM(4)) ⟶ Bidir(LSTM(2)) ⟶ Dropout(0.2) ⟶ Dense(8) ⟶ Dense(2,Softmax)

A.1.3. CNN+BLSTM

CNN-1D(8,3) ⟶ CNN-1D(10,3) ⟶ MaxPool-1D(3) ⟶ CNN-1D(16,3) ⟶ CNN-1D(20,3) ⟶ MaxPool-1D(3) ⟶ Bidir(LSTM(8)) ⟶ BatchNorm() ⟶ Bidir(LSTM(16)) ⟶ Dense(64,Relu) ⟶ Dense(32,Relu) ⟶ Dense (2,Softmax)

References

[1] M. W. Bondi, E. C. Edmonds, D. P. Salmon, Alzheimer's disease: past, present, and future, Journal of the International Neuropsychological Society 23 (2017) 818–831.

[2] S. Karlekar, T. Niu, M. Bansal, Detecting linguistic characteristics of Alzheimer's dementia by interpreting neural models, arXiv preprint arXiv:1804.06440 (2018).

[3] W. Kong, H. Jang, G. Carenini, T. Field, A neural model for predicting dementia from language, in: Machine Learning for Healthcare Conference, 2019, pp. 270–286.

[4] S. O. Orimaye, J. S.-M. W. K. J. Golden, Learning Predictive Linguistic Features for Alzheimer's Disease and related Dementias using Verbal Utterances, in: Proceedings Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality, 2014, pp. 78–87.

[5] S. Al-Hameed, M. Benaissa, H. Christensen, Simple and robust audio-based detection of biomarkers for alzheimer's disease, in: 7th Workshop on Speech and Language Processing for Assistive Technologies (SLPAT), 2016, pp. 32–36.

[6] V. Masrani, Detecting dementia from written and spoken language, Ph.D. thesis, University of British Columbia, 2018.

[7] M. Yancheva, F. Rudzicz, Vector-space topic models for detecting Alzheimer's disease, in: Proceedings Annual Meeting of the Association for Computational Linguistics, 2016, pp. 2337–2346.

[8] C. Guinn, A. Habash, Language analysis of speakers with dementia of the Alzheimer's type, in: AAAI Fall Symposium Series, 2012, pp. 8–13.

[9] S. O. Orimaye, J. S.-M. Wong, K. J. Golden, C. P. Wong, I. N. Soyiri, Predicting probable Alzheimer's disease using linguistic deficits and biomarkers, BMC Bioinformatics 18 (2017).

[10] S. O. Orimaye, J. S.-M. Wong, C. P. Wong, Deep language space neural network for classifying mild cognitive impairment and alzheimer-type dementia, PloS one 13 (2018).

[11] Y. Pan, B. Mirheidari, M. Reuber, A. Venneri, D. Blackburn, H. Christensen, Automatic hierarchical attention neural network for detecting ad, 2019, pp. 4105–4109. doi:10.21437/Interspeech.2019-1799.

[12] A. Balagopalan, B. Eyre, F. Rudzicz, J. Novikova, To bert or not to bert: Comparing speech and language-based approaches for alzheimer's disease detection, 2020. arXiv:2008.01551.

[13] J. T. Becker, F. Boller, O. L. Lopez, J. Saxton, K. McGonigle, The natural history of Alzheimer's disease. Description of study cohort and accuracy of diagnosis., Archives of Neurology 51 (1994) 585–594.

[14] B. Macwhinney, The CHILDES project: tools for analyzing talk, Child Language Teaching and Therapy 8 (2000).

[15] J. L. Lee, R. Burkholder, G. B. Flinn, E. R. Coppess, Working with CHAT transcripts in Python, Technical report TR-2016-02, Technical Report, Department of Computer Science, University of Chicago, 2016.

[16] T. Mikolov, et al., Efficient Estimation of Word Representations in Vector Space, in: Proceedings International Conference on Learning Representations, 2013.

[17] Pennington, Jeffrey, R. Socher, C. D. Manning, Glove: Global vectors for word representation, in: Proceedings International Conference Empiricial Methods in Natural Language Processing, 2014.

[18] P. Bojanowski, E. Grave, A. Joulin, T. Mikolov, Enriching Word Vectors with Subword Information, arXiv:1607.04606 (2016).

[19] R. Rehurek, P. Sojka, Software framework for topic modelling with large corpora, in: Proceedings International Workshop on New Challenges for NLP Frameworks, 2010.

[20] Keras, Deep learning for humans, https://github.com/fchollet/keras, 2015. Last accessed on Nov 2019.

[21] J. Schmidhuber, Deep learning in neural networks: An overview, Neural Networks 61 (2015) 85–117.

[22] A. F. Agarap, Deep Learning using Rectified Linear Units (ReLU), arXiv:1803.08375 (2018).

[23] N. Srivastava, et al., Dropout: A Simple Way to Prevent Neural Networks from Overfitting, Journal of Machine Learning Research 15 (2014) 1929–1958.

[24] D. P. Kingma, J. Ba, A Method for Stochastic Optimization, ?arXiv:1412.6980 (2014).

[25] L. Hebert, P. A. Scherr, J. L. Bienias, D. A. Bennett, D. A. Evans, Alzheimer disease in the US population: prevalence estimates using the 2000 census, Arch Neurol 60 (2003) 1119–1122.