Siamese Networks for Inference in Malayalam Language Texts

Sara Renjit
Department of Computer Science
CUSAT, Kochi, India
sararenjit.g@gmail.com

Sumam Mary Idicula
Department of Computer Science
CUSAT, Kochi, India
sumam@cusat.ac.in

Abstract

Natural language inference is a method of finding inferences in language texts. Understanding the meaning of a sentence and its inference is essential in many language processing applications. In this context, we consider the inference problem for a Dravidian language, Malayalam. Siamese networks train the text hypothesis pairs with word embeddings and language agnostic embeddings, and the results are evaluated against classification metrics for binary classification into entailment and contradiction classes. XLM-R embeddings based Siamese architecture using gated recurrent units and bidirectional long short term memory networks provide promising results for this classification problem.

1 Introduction

Textual entailment is a uni-directional relationship between two text pairs. It is the method of identifying the meaning from two sentences. Sentence pairs are labeled as entailment pairs if a sentence is inferred from the context of the other sentence in the pair.

It is defined in different ways as classical definition, applied definition, and mathematical definition (Ghuge and Bhattacharya, 2014). In the classical definition, a text entails a hypothesis if the hypothesis is valid in all circumstances where the text is true. In applied definition, text entails hypothesis if the hypothesis is mostly true when a human reads it. The mathematical definition (Glickman et al., 2005) is a text that entails a hypothesis if the probability of the hypothesis is true given the text is greater than the likelihood of the hypothesis being true.

The sentence from which we derive the information is called text, and the sentence which identifies its information as derived from text is called a hypothesis. Sentence pairs are named as entailed if the hypothesis has its meaning derived from the text. Contradictory pairs indicate that the information from the hypothesis is contradictory concerning the information conveyed in the text.

Nowadays, textual entailment has become part of the primary natural language processing tasks. It is also called natural language inference. The terms text and premise are used interchangeably. The performance of the latest transformer-based approaches is evaluated in NLP tasks like entailment recognition, semantic textual similarity, and paraphrase detection. Hence, textual entailment recognition has also become an evaluation criterion for many NLP tasks. It is also a necessary sub-task in applications like multi-document summarization, information retrieval, information extraction, and question answering systems.

Many text entailment or natural language inference related works in English use different sized datasets but not in the Malayalam language. Malayalam is a Dravidian language used in the southern part of India. It is a language that has various dialects and has many inflections. Malayalam language computing is developing, with few resources, namely stemmer, POS tagger, sandhi splitter, and few datasets for paraphrasing, text classification, and sentiment analysis. It has agglutinated language structure, and new words are created through word compounding and inflection, and hence there can be many inferential compound words for a word.

The main challenge of textual entailment recognition in the Malayalam language is the absence of a dataset. Dataset creation is a tedious task that involves high costs and time. We used an in-house dataset created by machines and humans in the loop translation of the Stanford Natural Language Inference dataset. It is a very cost-effective method of dataset creation that can be adapted to any low-resource language. This work attempts

Proceedings of Recent Advances in Natural Language Processing, pages 1167–1173
Sep 1–3, 2021.
https://doi.org/10.26615/978-954-452-072-4_131

to use Siamese networks with bidirectional long short term memory and gated recurrent units to understand the similarities and differences of text-hypothesis pairs for classification.

The remaining sections are organized as follows. Section 2 briefs the works related to textual entailment recognition. Section 3 details the dataset. Section 4 discusses the design of the system for classification. The experimental settings, results, and discusses sentence similarity are in Section 5. Section 6 concludes the work.

2 Related Works

Textual entailment recognition started with a Recognizing Textual Entailment (RTE)challenge in 2005 (Dagan et al., 2005). This challenge continued for years and used different-sized datasets. As the dataset size increases, feature-based methods such as word overlap, n-gram match, set-based similarities, and syntactic similarities were replaced by machine learning and deep learning methods using different word and sentence representations.

There are numerous entailment recognition systems in English and other languages like French, German, Italian, Spanish, and Arabic. Various textual entailment frameworks have also been developed in these languages, namely EXCITEMENT Open Platform (EOP) (Magnini et al., 2014). EOP uses edit distance and classification as its algorithms. Lexical level features, syntactic and surface-level features, graph-based approaches were used to recognize entailments. The increase in datasize has helped to use machine learning and deep learning strategies for this classification. Deep learning methods are widely used for textual entailment in English using different datasets.

SNLI (Stanford Natural Language Inference) dataset is a collection of 570k sentence pairs mainly collected through Amazon Mechanical Turk, referencing the Flickr30k corpus. This dataset helped in using deep learning techniques to text entailment recognition. Sentence models with the sum of words, recurrent neural networks, and long short-term memory networks were discussed (Bowman et al., 2015).

MNLI (Multi-Genre Natural Language Inference) dataset is a collection of 433k sentence pairs from different written and spoken English genres. Some of the genres are face-to-face, government, telephone, letters, fiction, and travel. It is an improvement from SNLI with a more diverse collection of sentence pairs, and hence its baseline performance is low compared to SNLI (Williams et al., 2018).

XNLI (Conneau et al., 2018) (Cross-Lingual Natural Language Inference) corpus is a collection of data from MNLI, derived for 15 languages, including some low resource languages like Urdu and Swahili. It uses a translation-based approach with multilingual sentence encoders and then aligning sentence embeddings for inference identification. Except for English, entailment recognition systems exist for French, Spanish, German, Greek, Bulgarian, Russian, Turkish, Arabic, Vietnamese, Thai, Chinese, Hindi, Swahili, and Urdu. In languages like Swahili, Thai and Urdu, transfer learning based approaches are used, which is helpful for small-sized datasets.

For the Malayalam language, there are works related to paraphrasing, sentiment analysis, summarization, whereas text entailment recognition is seemingly a new area for Malayalam language processing. The performance of different embedding-based approaches is one work in this language for entailment recognition (Renjit and Idicula, 2021). In this work, LASER-based embedding representation showed improved performance results for entailment recognition for the Malayalam language compared with BERT and other models.

Bidirectional LSTM based dependent reading represents text and hypothesis in encoding and inference stage(Ghaeini et al., 2018). Siamese network-based architecture with sentence embeddings from BERT experiments in the English language (Reimers et al., 2019). A neural model based on LSTM using the word by word attention is another deep learning-based method for recognizing entailments (Rocktäschel et al., 2015). Child-Sum-Tree-based inference of texts generalizes well for SNLI and other entailment datasets (John et al., 2016). Text alignment based approaches along with machine learning are used for entailment recognition in Arabic language (Boudaa et al., 2019). Another method used asymmetric word embeddings to produce similarity based word-word interactions for textual entailment (Ma et al., 2018).

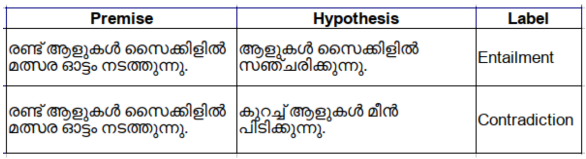
Entailment recognition has been part of Competition on Legal Information Extraction / Entailment, where sentence encoding and decomposable attention models perform entailment recognition in the context of legal texts (Son et al., 2017). Automatic translation-based approaches are used in the Italian dataset, where the dataset is translated into the English language for entailment recognition (Pakray et al., 2012).

3 Dataset

The subset of the Malayalam Language Inference (MaNLI) dataset is used for binary entailment recognition. It consists of 7989 text hypothesis pairs which are labeled as Entailment and Contradiction. This dataset is created from Stanford Natural Language Inference (SNLI) dataset. The sentence pairs from the SNLI dataset are translated to the Malayalam language with linguistic corrections from the Department of Linguistics, Malayalam University, Kerala.

There are 4026 entailment pairs and 3963 contradiction pairs in this dataset. The reason for the creation of this dataset is the unavailability of entailment datasets in Malayalam. Languages like Malayalam have many inherent linguistic properties like inflections, agglutinative nature, dialect-related differences, and no specific word order.

A sample from the dataset is shown in Figure 1.



Premise	Hypothesis	Label
രണ്ട് ആളുകൾ സൈക്കിളിൽ മത്സര ഓട്ടം നടത്തുന്നു.	ആളുകൾ സൈക്കിളിൽ സഞ്ചരിക്കുന്നു.	Entailment
രണ്ട് ആളുകൾ സൈക്കിളിൽ മത്സര ഓട്ടം നടത്തുന്നു.	കുറച്ച് ആളുകൾ മീൻ പിടിക്കുന്നു.	Contradiction

Figure 1: Sample dataset

The English translation of the sample dataset is provided in Table 1.

Premise	Hypothesis	Label
Two men on bicycles.	People are riding bikes.	Entailment
Two men on bicycles.	A few people are catching fish.	Contradiction

Table 1: Sample Dataset translation in English

4 Proposed Architecture

The design of this system consists of neural networks of identical architecture consisting of an embedding layer and bidirectional long short-term memory/gated recurrent unit networks. The Siamese network for sentence representation takes each sentence from text-hypothesis pair to an embedding layer. This layer uses different types of embeddings, namely Word2Vec and LASER (Language Agnostic Sentence Representations), and XLM-R. The different layers in the system are:

4.1 Embedding

The first layer is the embedding layer, where each sentence in text or hypothesis gets an efficient representation that captures its meaning in high dimensional vector space. In this layer, we used approaches, namely, Word2Vec, language-agnostic sentence representation (LASER), and XLM-R.

4.1.1 Word2Vec

Word2vec (Mikolov et al., 2013) is a word embedding neural model that produces distributed representation of words in vector space. The neural model trains in two ways, namely skip-gram and continuous bag of words, using hierarchical softmax or negative sampling (Rong, 2014). Words having semantic similarity represented through vectors are closer in high dimensional space. The embeddings from this model are input to the Keras embedding layer in the form of an embedding matrix to obtain text representation and hypothesis.

4.1.2 LASER

Language Agnostic Sentence Representations (Artetxe and Schwenk, 2019) is a toolkit modeled for more than 90 languages, including the Malayalam language. LASER embeddings are representations of sentences so that a sentence representation in two or more languages will be close to each other in their high dimensional vector space. It also uses an encoder-decoder architecture based on neural machine translation.

4.1.3 XLM-R

XLM-R (Conneau et al., 2019) is a self-supervised model that is trained on cross-lingual representations. This transformer-based masked language model is trained for 100 languages, including the Malayalam language. The cross-lingual sentences are taken from Common Crawl data.

4.1.4 Dimensionality Reduction using PCA

Principal component analysis (PCA) reduces the sentence embedding obtained from the LASER model. It compromises with accuracy, and hence selecting an adequate number of features is critical to the model. Depending on the system's configuration, dimensions of 100, 500, and 1000 are tested for 1024 dimensional LASER embeddings. Dimension reduction of 100 leads to 0.87% loss of features, dimension 500 causes 0.488% feature loss, and dimension 1000 led to 0.023% feature loss.

4.2 BLSTM

The bidirectional LSTM layer consists of two LSTM layers, one in forward and the other in the backward direction for sequence processing. It is used when a sequence of data is to be processed. The text and hypothesis are passed through BLSTM layers to obtain a sequence representation that embeds the complete information.

4.3 GRU

This layer has recurrent neural networks with a gating mechanism. It is similar to long short-term memory networks, and it does not have output gates. As such, it has less number of parameters with good performance on small-sized datasets.

4.4 CONCATENATION

The sequence representation of text and hypothesis are then concatenated in this layer, and batch normalization is done. Dropout configurations are then applied and passed to a Dense layer.

4.5 DENSE LAYER

The dense layer has its input from the concatenation layer and has rectified leaky unit activation function. It is then batch normalized, and dropout is applied and flattened to feed to the following dense layer.

4.6 SIGMOID CLASSIFICATION

For binary classification, the sigmoid activation function is used in the final dense layer. The sigmoid function is given by

$$S(x) = 1/(1 + e^{-x}) \quad (1)$$

The system design is shown in Figure 2. The representations from the embedding layer is then passed to a bidirectional LSTM / GRU layer where each sentence gets a context representation. It is performed for both text and hypothesis in Siamese network architecture, followed by a concatenation of the outputs from BiLSTM / GRU. The concatenated text hypothesis representation is then fed to a Dense layer with ’RELU’ activation followed by classification. Sigmoid function with binary cross-entropy loss function performs the model training and classification.

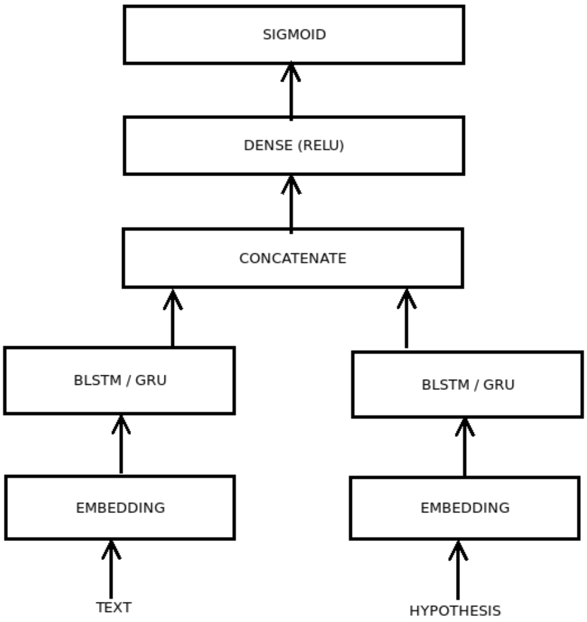


Figure 2: System Design

5 Experimental Results

5.1 Experimental Setup

Implementations used Google Colab platform and Spyder IDE using Python, Tensorflow, and Keras library for machine learning and Scikit-Learn for evaluations.

The parameter configurations for training the system are 100 LSTM nodes, 100 dense units, RELU activation function for dense layer, drop-out rate of 0.17,0.25 for LSTM and 0.25 dropout for dense layer.

5.2 Results

The results are evaluated in terms of classification metrics, namely Precision(P), Accuracy, Recall(R), F1-score(F1), and Support(S). Experimental results of the Siamese network architecture detailed above are shown in Table 2.

Class	P	R	F1	S
Contradiction	0.71	0.60	0.65	500
Entailment	0.65	0.76	0.70	500
Accuracy			0.68	1000
Macro average	0.68	0.68	0.68	1000
Weighted average	0.68	0.68	0.68	1000

Table 2: Results based on LASER embedding with dimensions reduced to 1000.

From Table 3, we observe that reducing the embedding dimension to 500 through the principal component analysis technique results in performance drops, as there is information loss.

Class	P	R	F1	S
Contradiction	0.91	0.19	0.31	500
Entailment	0.55	0.98	0.70	500
Accuracy			0.58	1000
Macro average	0.73	0.58	0.51	1000
Weighted average	0.73	0.58	0.51	1000

Table 3: Results based on LASER embeddings with dimensions reduced to 500.

When the embedding dimension is reduced to 100, we obtain the results in Table 4. Hence dimensions of 100 and 1000 yield good results compared with dimension 500. It resulted due to a mismatch in network configuration with embedding size.

Class	P	R	F1	S
Contradiction	0.68	0.62	0.65	500
Entailment	0.65	0.71	0.68	500
Accuracy			0.66	1000
Macro average	0.66	0.66	0.66	1000
Weighted average	0.66	0.66	0.66	1000

Table 4: Results based on LASER embeddings with dimension reduced to 100.

Comparison with word2vec: The system design is compared with word2vec based embedding. The dataset is trained using the Word2vec model with dimension 100, minimum count of words 1. The difference in configurations showed better results, as shown in the tables below.

Configuration 1: With negative sampling and using a continuous bag of words approach for Word2Vec model produced the results as in Table 5.

Class	P	R	F1	S
Contradiction	0.60	0.43	0.50	500
Entailment	0.56	0.71	0.63	500
Accuracy			0.57	1000
Macro average	0.58	0.57	0.57	1000
Weighted average	0.58	0.57	0.57	1000

Table 5: Results based on Word2Vec with Configuration 1.

Configuration 2: With hierarchical softmax and skip-gram based approach resulted in Table 6. We infer that Word2Vec of 100 dimensions with hierarchical softmax and LASER embedding reduced to 1000 dimension shows good performance. Hence Word2vec is better for this Siamese network based architecture based on its performance with lesser dimensional embeddings.

Class	P	R	F1	S
Contradiction	0.76	0.49	0.60	500
Entailment	0.63	0.85	0.72	500
Accuracy			0.67	1000
Macro average	0.69	0.67	0.66	1000
Weighted average	0.69	0.67	0.66	1000

Table 6: Results based on Word2Vec with Configuration 2.

Comparison with GRU For the same architecture, when BiLSTM is replaced with GRU, Word2Vec based system showed the same performance as below.

Class	P	R	F1	S
Contradiction	0.76	0.51	0.61	500
Entailment	0.63	0.84	0.72	500
Accuracy			0.67	1000
Macro average	0.69	0.67	0.66	1000
Weighted average	0.69	0.67	0.66	1000

Table 7: Results based on Word2Vec with GRU layer instead of BiLSTM

LASER based GRU system shows the below results for classification as in Table8.

Class	P	R	F1	S
Contradiction	0.73	0.29	0.42	500
Entailment	0.56	0.89	0.69	500
Accuracy			0.59	1000
Macro average	0.64	0.59	0.55	1000
Weighted average	0.64	0.59	0.55	1000

Table 8: Results based on LASER with GRU layer instead of BiLSTM

Comparison with XLM-R embeddings XLM-R is a masked language model trained for 100 languages, including Malayalam. This transformer-based architecture produced the results shown in Table 9. The default dimension is 768, which is reduced to 100 dimensions.

XLM-R is also used with BiLSTM layer and the results are shown in Table10.

Class	P	R	F1	S
Contradiction	0.69	0.73	0.71	500
Entailment	0.71	0.67	0.69	500
Accuracy			0.70	1000
Macro average	0.70	0.70	0.70	1000
Weighted average	0.70	0.70	0.70	1000

Table 9: Results based on XLM-R with GRU layer in Siamese architecture.

Class	P	R	F1	S
Contradiction	0.75	0.65	0.70	500
Entailment	0.69	0.79	0.74	500
Accuracy			0.72	1000
Macro average	0.72	0.72	0.72	1000
Weighted average	0.72	0.72	0.72	1000

Table 10: Results based on XLM-R with BiLSTM layer in Siamese architecture.

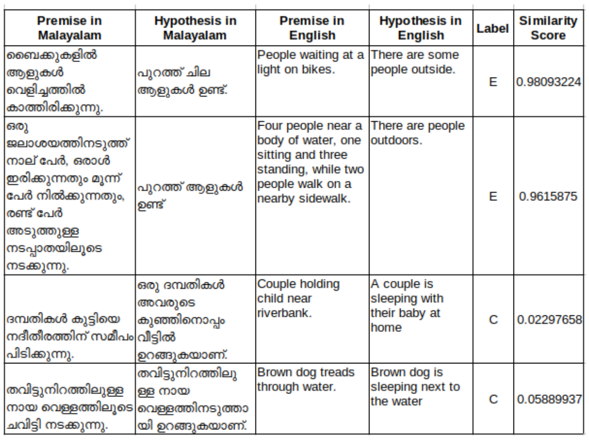
Table 11 shows the accuracy values obtained for different configurations of Siamese networks. It also includes the previous densenet based system results also.

Model	Accuracy
Siamese+BiLSTM+100D Word2Vec	0.67
Siamese+BiLSTM+100D LASER	0.68
Siamese+BiLSTM+1000D LASER	0.68
Siamese+GRU+100D LASER	0.59
Siamese+GRU+100D Word2Vec	0.67
Siamese+GRU+100D XLM-R	0.70
Siamese+BiLSTM+100D XLM-R	0.72
Densenet + 1024D LASER (Renjit and Idicula, 2021)	0.77

Table 11: Summary of different model configurations applied and their accuracies.

5.3 Sentence Similarity

As part of this classification, similarities of sentences with respect to text hypothesis pairs are measured quantitatively as entailment confidence. For example, the similarity score obtained for the classification of instances is shown in Figure 3. The entailment confidence score is helpful for sentence similarity tasks to identify the extent to which the pairs are similar. Thus it also aids in multi-document summarization tasks, in which we can avoid similar sentences in summary based on the entailment/similarity score.



Premise in Malayalam	Hypothesis in Malayalam	Premise in English	Hypothesis in English	Label	Similarity Score
ബൈക്കുകളിൽ ആളുകൾ വെളിച്ചത്തിൽ കാത്തിരിക്കുന്നു.	പുറത്ത് ചില ആളുകൾ ഉണ്ട്.	People waiting at a light on bikes.	There are some people outside.	E	0.98093224
ഒരു ജലാശയത്തിനടുത്ത് നാല് പേർ, ഒരാൾ ഇരിക്കുന്നതും മൂന്ന് പേർ നിൽക്കുന്നതും, രണ്ട് പേർ അടുത്തുള്ള നടപ്പാതയിലൂടെ നടക്കുന്നു.	പുറത്ത് ആളുകൾ ഉണ്ട്	Four people near a body of water, one sitting and three standing, while two people walk on a nearby sidewalk.	There are people outdoors.	E	0.9615875
ദമ്പതികൾ കുട്ടിയെ നദീതീരത്തിന് സമീപം പിടിക്കുന്നു.	ഒരു ദമ്പതികൾ അവരുടെ കുഞ്ഞിനൊപ്പം വീട്ടിൽ ഉറങ്ങുകയാണ്.	Couple holding child near riverbank.	A couple is sleeping with their baby at home	C	0.02297658
തവിട്ടുനിറത്തിലുള്ള നായ വെള്ളത്തിലൂടെ ചവിട്ടി നടക്കുന്നു.	തവിട്ടുനിറത്തിലുള്ള നായ വെള്ളത്തിനടുത്തായി ഉറങ്ങുകയാണ്.	Brown dog treads through water.	Brown dog is sleeping next to the water	C	0.05889937

Figure 3: Similarity scores example where C label denotes contradiction and E label denotes entailment class.

6 Conclusion

In this work, we focused on the application of Siamese network architecture to recognize entailment in Malayalam language. The results shows adequate performance. The use of newer embedding models leads to better accuracy but the embedding dimension is a limiting factor with the network configuration. As the embedding dimension increases, the time and space complexity increases in Siamese model architecture, where text and hypothesis are processed as sequences parallely.

Through this work, we aim to depict the performance of Siamese network based entailment recognition with respect to Malayalam language, which is a low resource Dravidian language.

References

Mikel Artetxe and Holger Schwenk. 2019. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610.

Tarik Boudaa, Mohamed El Marouani, and Nourddine Enneya. 2019. Alignment based approach for arabic textual entailment. *Procedia computer science*, 148:246–255.

Samuel R Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. 2015. A large annotated corpus for learning natural language inference. *arXiv preprint arXiv:1508.05326*.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel R. Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. Xnli: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2005. The pascal recognising textual entailment challenge. In *Machine Learning Challenges Workshop*, pages 177–190. Springer.

Reza Ghaeini, Sadid A Hasan, Vivek Datla, Joey Liu, Kathy Lee, Ashequl Qadir, Yuan Ling, Aaditya Prakash, Xiaoli Fern, and Oladimeji Farri. 2018. Drbilstm: Dependent reading bidirectional lstm for natural language inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1460–1469.

Swapnil Ghuge and Arindam Bhattacharya. 2014. Survey in textual entailment. *Center for Indian Language Technology, retrieved on April*.

Oren Glickman, Ido Dagan, and Moshe Koppel. 2005. A probabilistic lexical approach to textual entailment. In *IJCAI*, volume 5, pages 1682–1683.

Adebayo Kolawole John, Luigi Di Caro, Livio Robaldo, and Guido Boella. 2016. Textual inference with tree-structured lstm. In *Benelux Conference on Artificial Intelligence*, pages 17–31. Springer.

Tengfei Ma, Chiamin Wu, Cao Xiao, and Jimeng Sun. 2018. Awe: asymmetric word embedding for textual entailment. *arXiv preprint arXiv:1809.04047*.

Bernardo Magnini, Roberto Zanoli, Ido Dagan, Kathrin Eichler, Günter Neumann, Tae-Gil Noh, Sebastian Pado, Asher Stern, and Omer Levy. 2014. The excitement open platform for textual inferences. In *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, pages 43–48.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Partha Pakray, Snehasis Neogi, Sivaji Bandyopadhyay, and Alexander Gelbukh. 2012. Recognizing textual entailment in non-english text via automatic translation into english. In *Mexican International Conference on Artificial Intelligence*, pages 26–35. Springer.

Nils Reimers, Iryna Gurevych, Nils Reimers, Iryna Gurevych, Nandan Thakur, Nils Reimers, Johannes Daxenberger, and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Sara Renjit and Sumam Idicula. 2021. Natural language inference for malayalam language using language agnostic sentence representation. *PeerJ Computer Science*, 7:e508.

Tim Rocktäschel, Edward Grefenstette, Karl Moritz Hermann, Tomáš Kočiskỳ, and Phil Blunsom. 2015. Reasoning about entailment with neural attention. *arXiv preprint arXiv:1509.06664*.

Xin Rong. 2014. word2vec parameter learning explained. *arXiv preprint arXiv:1411.2738*.

Nguyen Truong Son, Viet-Anh Phan, and Le Minh Nguyen. 2017. Recognizing entailments in legal texts using sentence encoding-based and decomposable attention models. In *COLIEE@ ICAIL*, pages 31–42.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122.