

# Lexical Complexity Prediction

Anshul Choudhary (17CS10005), Rithin Manoj (17CS10043), Anshul Goel (17CS30005),  
Kshitiz Sharma (17CS30021), Venktesh Lagaskar (17CS30037)

Codalab Team Name: iitkgp\_CS60075\_team13

Team ID: 13

Subtask ID: 1

## 1 Individual Contribution

- Anshul Choudhary:
  - Code: Features, Model, Testing
  - Final Report: Model
- Rithin Manoj:
  - Midsem Report
- Anshul Goel :
  - Code: Features
  - Final Report: Introduction, Model, Experiments, Results
- Kshitiz Sharma :
  - Code: Features
  - Final Report: Approach, Model, Experiments, Results
- Venktesh Lagaskar :
  - Midsem Report
  - Final Report: Approach

## 2 Introduction

Lexical complexity plays a crucial role in reading comprehension as well as in learning the language or in learning new words in the language. Predicting lexical complexity can enable various systems to better guide a user to an appropriate text, or tailor it to their needs. NLP systems have been developed to simplify texts for second language learners, native speakers with low literacy levels, and people with reading disabilities. By properly understanding and predicting the complexity of words and phrases in a text, we can measure whether it has reduced in complexity after simplification.

Previous approaches to Complex Word Identification (CWI), such as the one used in the CWI shared task (SemEval-2016 Task 11) (Paetzold and Specia, 2016), approached the task as a binary classification task in which systems predict a complexity value (complex vs. non-

complex) for a set of target words in a text. But there are several limitations of calculating complexity using binary judgements. In this project our main dataset has a complexity score for the words/phrase which is used to create a 5-point Likert scale. Likert scale (1-5) corresponding to the annotators comprehension and familiarity with the words in which 1 represents very easy and 5 represents very difficult

## 3 Approach

We first researched different metrics that could be used to include meanings/ statistical features of words. Research papers based of CWI 2016 and 2018 Shared Tasks were helpful in listing various features that proved of significance in the past. Some of them are as follows:

### 1) Token Embeddings

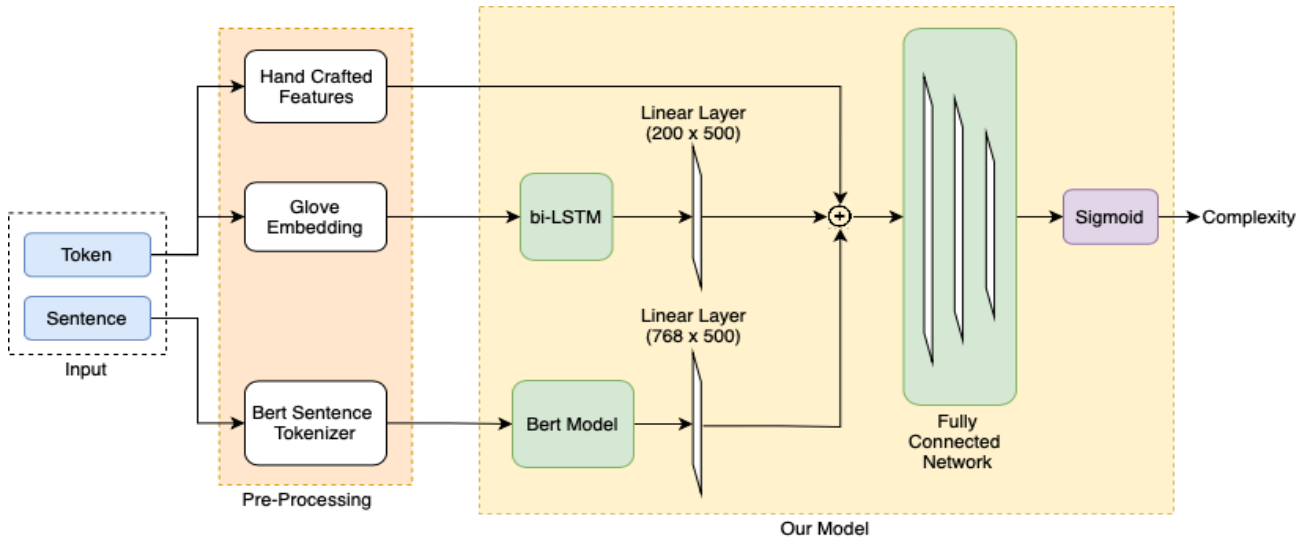
- a) Glove Embeddings: 300-dimensional  
Glove embedding is preferred over other models like word2vec because it does not just rely on local statistics but also takes into account the global statistics to obtain word vectors.

### 2) Sentence Embeddings

- a) InferSent Embeddings: 4,096 dimensional embeddings
- b) BERT: It is better because it practices to predict missing words in the text, and because it analyzes every sentence with no specific direction, it does a better job at understanding the meaning of homonyms than the other models.

### 3) Hand-Crafted Features

- a) Token Frequency:
  - i) term\_freq\_doc - Frequency of the token in the whole dataset. This represents how common a word is.



Roughly indicating how simple a word is.

- ii) term\_freq\_wiki - frequency of the token in the nltk dataset.
- b) token\_len: Length of the target word/phrase (the word/phrase whose complexity we have to find. This approximately infer that longer a word more complex it is.
- c) vowel\_count: number of vowels in the target word/phrase. This is based on the fact that phonetic complexity is proportional to vowel count, and more phonetic complex words will tend to be lexically more complex.
- d) syllable\_count: It is the number of syllables present in the target word/phrase. This is based on the fact that phonetic complexity is proportional to syllable count, and more phonetic complex words will tend to be lexically more complex.
- e) pos\_tag: The Part of Speech tag of the word/phrase. It is a good representation of both its meaning and context which is relevant to understanding hence complexity of a word.
- f) Concreteness: Concreteness score, listed in the MRC Database
- g) Imageability: Imageability score of the word, listed in the MRC Database

### 3.1 Model

Input consists of two parts: Token, Sentence. Meaning and context are important deciding factors for complexity of a token (target word). We gather this information by breaking the input into three parts:

1. We use pertained glove embedding to generate embeddings for tokens. In case of Multi-Word expressions, we split the word and the average of the glove-embeddings of two words were used. These embeddings are sent as input to bi-LSTM, and further into a Linear layer.
2. Handcrafted features generated from a token (target word) are used to give more information about tokens. These are:
  - a. Token length
  - b. Syllable Count
  - c. Vowel Count
  - d. Token Type (Single word/ Multi word expression)
3. Sentence Embeddings: we first convert a sentence into a vector of fixed dimension by using pre trained BertTokenizer and using appropriate padding. Attention mask was also generated to retain information regarding the padded part. This was then passed to the Bert Model and further to a linear layer for training.

The output of above three are concatenated and given as input to a fully connected network consisting of three layers of dimensions 1004x400, 400x100, 100x1. A sigmoid neuron was used to generate the result.

Adam optimizer with learning rate of  $2e-5$  and Adam's epsilon =  $1e-8$  were used. We varied epochs in the range of 1-7 to find the epoch for the most efficient model.

## 4 Experiments

We experimented our code in the following ways to improvise our results:

1. Hand Crafted Features:
  - a) Tried different permutations of the features mentioned in Section 3.
2. Single, multiple train together/ separately:
  - a) Problem statement consisted of two parts, namely complexity for single word tokens, multi word expressions. Both of these could be combined/ used separately to train different parts/types of the models.
3. Hyper parameters: batch sizes, learning rate:
  - a) We have tried different batch sizes and learning rate to train our model and finally train our model with the batch size of 32 and the learning rate of  $2e-5$ .
4. Layers: We experimented by changing the number of hidden layers, dimensions and the dropout in the LSTM, linear layers and the fully connected networks.
5. Sentence embedding:
  - a) We came across different kinds of sentence embeddings, but it was found that bert, infersent were the most preferred ones. We finally used a mixture of pertained and trained bert as it gave the better results.
6. Feature handling in Multi-word expression subtask
  - a) In case of Multi-Word expressions we experimented by taking into account the features in two ways:
    - i) Concatenating them and passing into model
    - ii) Taking the average of numerical values

## 5 Results

Table 1 shows the pearson's score generated by the model described above. Single word expressions gives the maximum score at the 4<sup>th</sup> epoch, while multi word expression gives the maximum score at the 2<sup>nd</sup> epoch.

No. of epoch	Single word	Multi word
0	0.6948594	0.7789802

1	0.7261809	0.8135070
2	0.7440118	<b>0.8259746</b>
3	0.7496528	0.8233133
4	<b>0.7503996</b>	0.8142233
5	0.7503395	0.8224805
6	0.7482303	0.8211479

**Table. 1** Pearson's score generated by our model on various epochs

## 6 References


1. Mounica Maddela and Wei Xu Department of Computer Science and Engineering The Ohio State University; *A Word-Complexity Lexicon and A Neural Readability Ranking Model for Lexical Simplification*
2. Sian Gooding Dept of Computer Science and Technology University of Cambridge and Ekaterina Kochmar ALTA Institute University of Cambridge; *Complex Word Identification as a Sequence Labelling Task*
3. Seid Muhie Yimam†, Sanja Štajner, Martin Riedl, and Chris Biemann, Language Technology Group, Department of Informatics, Universität Hamburg, Germany Data and Web Science Group, University of Mannheim, Germany; *CWIG3G2 – Complex Word Identification Task across Three Text Genres and Two User Groups*
4. Segun Taofeek Aroyehun CIC, Instituto Politecnico Nacional Mexico City, Mexico and Jason Angel CIC, Instituto Politecnico Nacional Mexico City, Mexico and Daniel Alejandro Perez Alvarez CIC, Instituto Politecnico Nacional Mexico City, Mexico and Alexander Gelbukh CIC, Instituto Politecnico Nacional Mexico City, Mexico; *Complex Word Identification: Convolutional Neural Network vs. Feature Engineering*
5. Matthew Shardlow and Michael Cooper, Manchester Metropolitan University, UK and Marcos Zampieri, Rochester Institute of Technology, USA; *CompLex: A New Corpus for Lexical Complexity Prediction from Likert Scale Data*

6. Matthew Shardlow, Richard Evans, Marcos Zampieri; *Predicting Lexical Complexity in English Texts*.
7. Jacob Devlin Ming-Wei Chang Kenton Lee Kristina Toutanova, Google AI Language; BERT: *Pre-training of Deep Bidirectional Transformers for Language Understanding*
8. Jeffrey Pennington, Richard Socher, Christopher D. Manning, Computer Science Department, Stanford University, Stanford, CA 94305; *GloVe: Global Vectors for Word Representation*
9. MICHAEL WILSON, Rutherford Appleton Laboratory, Oxfordshire, England; *MRC Psycholinguistic Database: Machine-usable dictionary, version 2.00*

## Submission Screenshots of Simple and Multi Word Complexity Prediction:

CodaLab
My Competitions
Help
kshitizs2809

Competition



### SemEval 2021 Task 1 - Lexical Complexity Prediction (LCP)

Organized by ghpaetzold - Current server time: April 12, 2021, 2:04 p.m. UTC

Previous

Current

End

TEST: Sub-Task 1 (Single Words)

TEST: Sub-Task 2 (Multi-Word Expressions)

Competition Ends

Jan. 11, 2021, midnight UTC

Jan. 11, 2021, midnight UTC

Never

Learn the Details
Phases
Participate
Results

Get Data
Files
Submit / View Results

TRIAL: Sub-Task 1 (Single Words)

TRIAL: Sub-Task 2 (Multi-Word Expressions)

TEST: Sub-Task 1 (Single Words)

TEST: Sub-Task 2 (Multi-Word Expressions)

Phase description  
None

Max submissions per day: 3  
Max submissions total: 3


Click the Submit button to upload a new submission.  

Optionally add more information about this submission

Submit

Here are your submissions to date (✓ indicates submission on leaderboard):

#	SCORE	FILENAME	SUBMISSION DATE	STATUS	✓	+
1	0.7515011343	NLP_SemEval21_Team_13_single.zip	04/12/2021 13:18:27	Finished	✓	+



### SemEval 2021 Task 1 - Lexical Complexity Prediction (LCP)

Organized by ghpaetzold - Current server time: April 12, 2021, 2:04 p.m. UTC

Previous

Current

End

TEST: Sub-Task 1 (Single Words)

TEST: Sub-Task 2 (Multi-Word Expressions)

Competition Ends

Jan. 11, 2021, midnight UTC

Jan. 11, 2021, midnight UTC

Never

Learn the Details
Phases
Participate
Results

Get Data
Files
Submit / View Results

TRIAL: Sub-Task 1 (Single Words)

TRIAL: Sub-Task 2 (Multi-Word Expressions)

TEST: Sub-Task 1 (Single Words)

TEST: Sub-Task 2 (Multi-Word Expressions)

Phase description  
None

Max submissions per day: 3  
Max submissions total: 3

Click the Submit button to upload a new submission.  

Optionally add more information about this submission

Submit

Here are your submissions to date (✓ indicates submission on leaderboard):

#	SCORE	FILENAME	SUBMISSION DATE	STATUS	✓	+
1	0.8251218155	NLP_SemEval21_Team_13_multi.zip	04/12/2021 13:48:35	Finished	✓	+