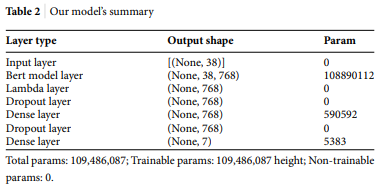
# INTENT RECOGNITION USING BERT

## BERT-IR: [Paper 0](https://link.springer.com/article/10.2991/hcis.k.211109.001)

Used dense layers with dropouts after BERT for intent recognition task.

🡪*Text preprocessing*Remove special characters, tokenize data, convert tokens to numbers, padding, remove text from other languages, messages containing < 3 words are removed, each message is taken only once to avoid repetition.



*🡪Hyperparameters*Adam with lr 0.00005  
Validation split of 0.1  
epochs: 50, batch size:16  
Dropout in dense layer with a rate of 0.1  
Early stopping with 2 epochs as patience

*🡪Metrics*Accuracy, Precision, Recall, F1 score

## PROJECT PLAN

### DATA AUGMENTATION

* **Synonym Replacement**: Replacing words in a sentence with their synonyms can create new samples that retain the same meaning but use different words.
* **Back Translation**: Translating a sentence from one language to another and back to the original language can create new samples with different phrasing and word choices.
* **Random Deletion**: Randomly deleting words from a sentence can create new samples with different sentence structures and help the model learn to deal with missing information.
* **Random Swap**: Randomly swapping two words in a sentence can create new samples with different sentence structures and word orders.
* **Random Insertion:** Randomly inserting words into a sentence can create new samples with additional information and help the model learn to handle noisy input.
* **Changing Tenses**: Changing the tense of verbs in a sentence can create new samples with different temporal contexts.
* **Masking**: Replacing words in a sentence with a unique token can create new samples with missing information, which can help the model learn to fill in gaps in the input.
* **Noise Injection**: Add noise or errors to your dataset, such as typos, grammatical mistakes, or missing words. This can simulate real-world variations.

🡪nlpaug, textblob, augly

### DATA PREPROCESSING

* Lowercasing
* Remove non-word, non-white space characters and digits.
* Tokenization
* Stop words removal.
* Stemming
* Lemmatization
* Text normalization
* Parts of speech tagging
* Padding

### FEATURE ENGINEERING, SELECTION AND DIMENSIONALITY REDUCTION

*WORD EMBEDDING*  
wORD2VEC, GLOVE, FASTEXT

* **TF-IDF (Term Frequency-Inverse Document Frequency)**: TF-IDF assigns weights to words based on their frequency in a document relative to their frequency across all documents. It helps identify important words specific to a document.
* **N-grams**
* [**Principal Component Analysis (PCA)**](https://spotintelligence.com/2023/08/25/principal-component-analysis/): PCA reduces the dimensionality of the data while preserving the most essential information. It can be applied to word embeddings to create more compact representations.
* [**Feature Engineering**](https://spotintelligence.com/2023/03/25/nlp-feature-engineering/)**with Domain Knowledge**: In some cases, domain-specific knowledge can guide feature engineering. You can include specific keywords or entities that are highly relevant to your NLP intent classification task.

### NETWORK ARCHITECTURE

### TRAINING MODEL

### HYPERPARAM TUNING

Number of epochs, learning rate, batch size, data split ratio, Gradient descent technique  
Regularization:  
Dropout, early stopping, weight decay (L2)

### EVALUATION METRICS

Accuracy, precision, recall, F1 score