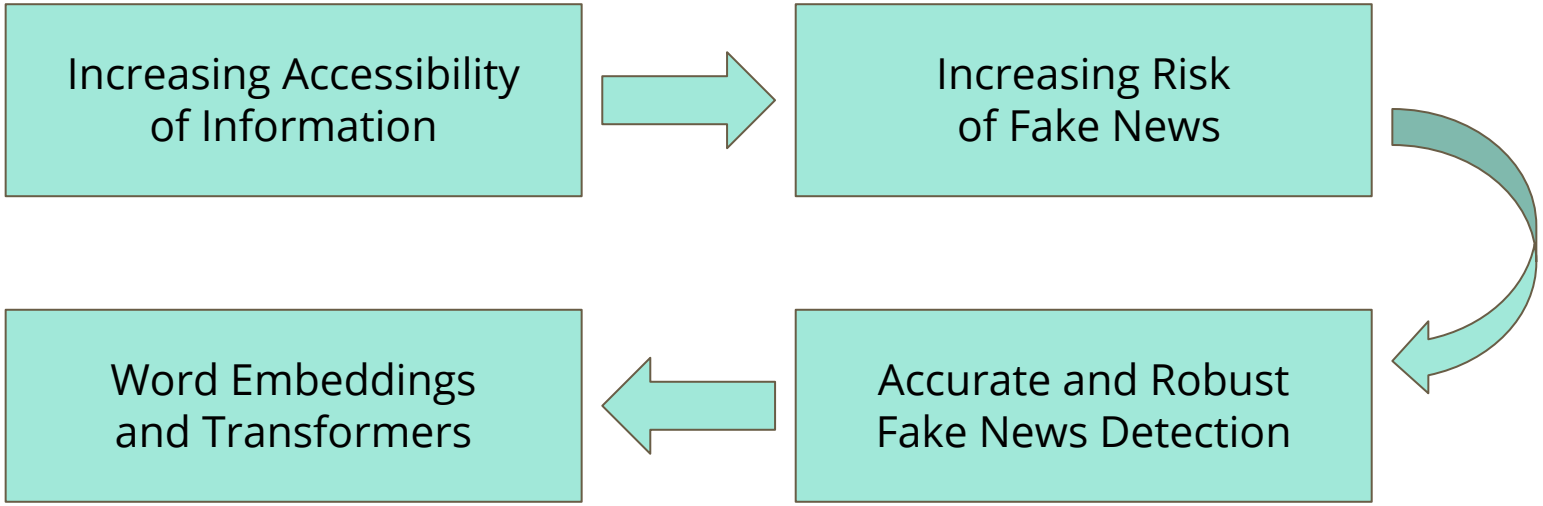
Performance Analysis of Different Word Embeddings and Transformers on Fake News Detection

PATUPAT Albert John Lalim - 20544416
CHENG I-Tsun - 20576079

Introduction



How well do different word embeddings and transformers perform on fake news detection?

Machine Learning Task

Fake News Detection, a Binary Text Classification Task

Title Text Label = 0 or 1

$$(x_{i1}, x_{i2}, y_i)$$

News Article

Datasets

- Real and Fake News Kaggle Dataset
  - 6335 News Articles (3174 Real and 3171 Fake)
  - Political News, especially 2016 US Presidential Elections
- ISOT Fake News Dataset
  - 44898 News Articles (21417 Real and 23481 Fake)
  - Political News, Government News, World News
- LIAR Dataset
  - 12791 Short Statements (2053-2454-2627-2103-2507-1047 from True to False)
  - Political and General Statements, Contextual Information

Preprocessing

- Column Preparation
  - Kaggle: label, title, text, titletext* [concatenation]
  - ISOT: label, text
  - LIAR: label*, text [top two true classes and top two false classes]
- Data Filtering
  - Drop examples with missing data
  - Trim textual data to first 200 words
  - ISOT: Trim Reuters datelines
- Dataset Splitting
  - Stratified

Models

Three Word **Embeddings**:

- BiLSTM with **Word2Vec** (300-dim)
- BiLSTM with **Word2Vec_2** (256-dim)
- BiLSTM with **GloVe** (300-dim): pretrained word representations trained on the global word-word co-occurrence matrices from Wikipedia and Gigaword-5
- BiLSTM with **ELMo** (256-dim): deeply contextualized word representation

Three **Transformers**:

- **BERT** (Bidirectional Encoder Representations from Transformers): deeply bidirectional and pretrained on the BooksCorpus+ English Wikipedia
- **ALBERT** (A Lite BERT): parameter-reduction techniques to lower memory consumption and improve training speed of BERT
- **DistilBERT**: 40% smaller yet 60% faster than BERT while retaining 97% of BERT's language capabilities

Hardware & Software Environment

- Google Colaboratory
- Pytorch 1.5 + TorchText 0.3 for model construction and GloVe implementation
- AllenNLP 0.9 for Elmo implementation
- SacreMoses 0.0 for tokenization before embedding layers
- Huggingface 2.9 for implementation of transformers

Hyperparameter Settings

Embedding Layers:

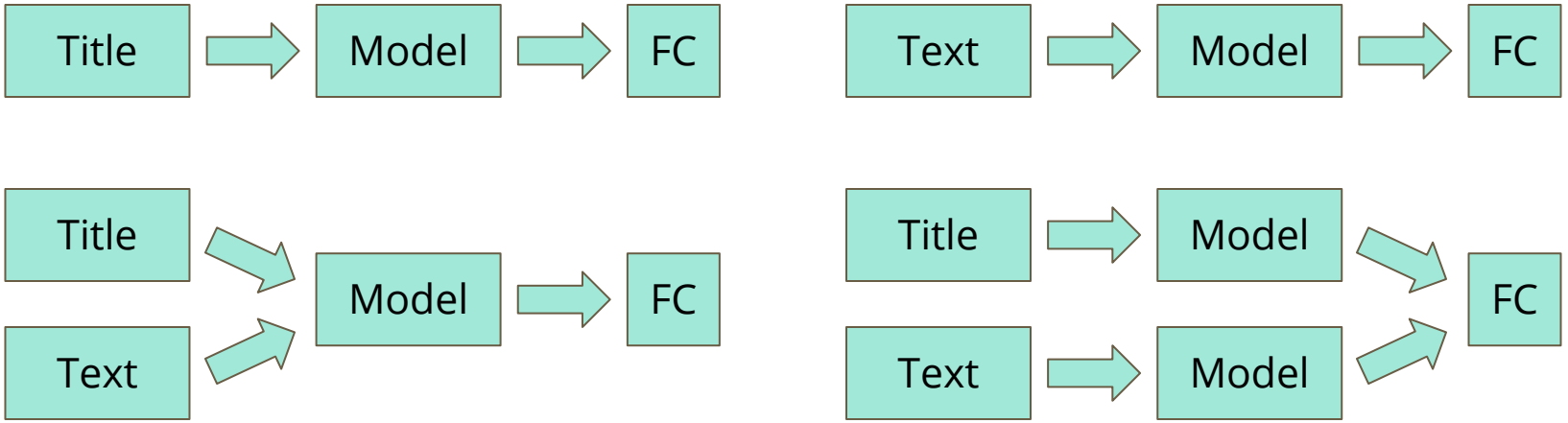
Embedding Layer	Word2Vec	Word2Vec_2	GloVe	ELMo
Tokenizer	Moses Tokenizer			
Input Restrictions	-			$\leq$ 50 char's
Embedding Dimensions	300	256	300	256
Encoder	1-layer BiLSTM with 128 dimensions			
Batch Size	32			
Learning Rate	1×10^{-3}			3×10^{-3}
Training Details	5 epochs with Adam optimizer and cross entropy loss			

Transformers:

Transformer	BERT	ALBERT	DistilBERT
Tokenizer	BERT Tokenizer	ALBERT Tokenizer	DistilBERT Tokenizer
Input Restrictions	$\leq$ 128 tokens		
Dimensions	768		
Batch Size	16		
Learning Rate	2×10^{-5}		
Training Details	5 epochs, Adam optimizer, cross entropy loss		

Experiments (Phase 1)

- Grid Search (Model x Version)
  - Model = Word2Vec, Word2Vec_2, GloVe, ELMo, BERT, ALBERT, DistilBERT
  - Version = Title, Text, Title-Text, Title+Text
  - Train-Valid-Test Split Ratio = 72-18-10



Experiments (Phase 2)

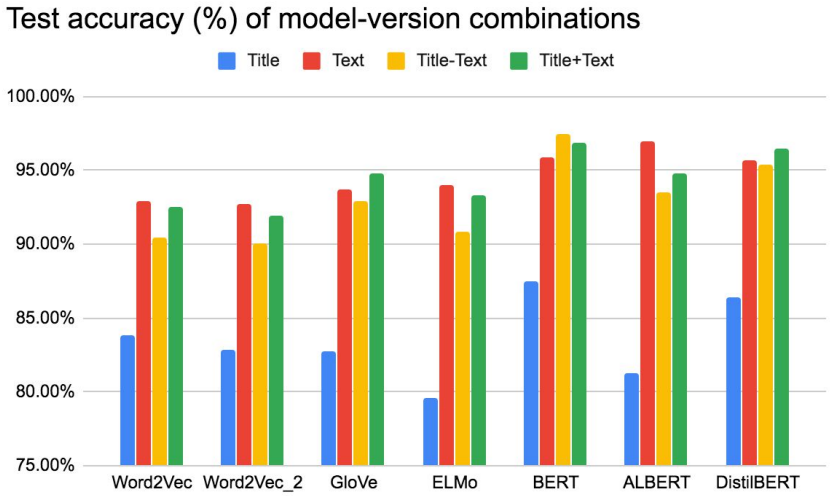
- Grid Search (Model x Ratio)
  - Model = Word2Vec, GloVe, DistilBERT
  - Version = Text
  - Ratio = 90, 80, 70, 60, 50, 40, 30, 20, 10

$$\text{Train-Valid-Test Split Ratio} = (0.8X)\text{-}(0.2X)\text{-}(100 - X)$$

Experiments (Phase 3)

- Grid Search (Model x Training x Dataset)
  - Model = Word2Vec, GloVe, DistilBERT
  - Version = Text
  - Training = Pretraining only, Training only, both Pretraining and Training
  - Dataset = ISOT Fake News Dataset, LIAR Dataset
- Pretraining on Real and Fake News Dataset
  - Train-Valid-Test Split Ratio = 80-20-0
- Training on ISOT Fake News Dataset
  - Train-Valid-Test Split Ratio = 2-0-98
- Training on LIAR Dataset
  - Train-Valid-Test Split Ratio = 72-18-90

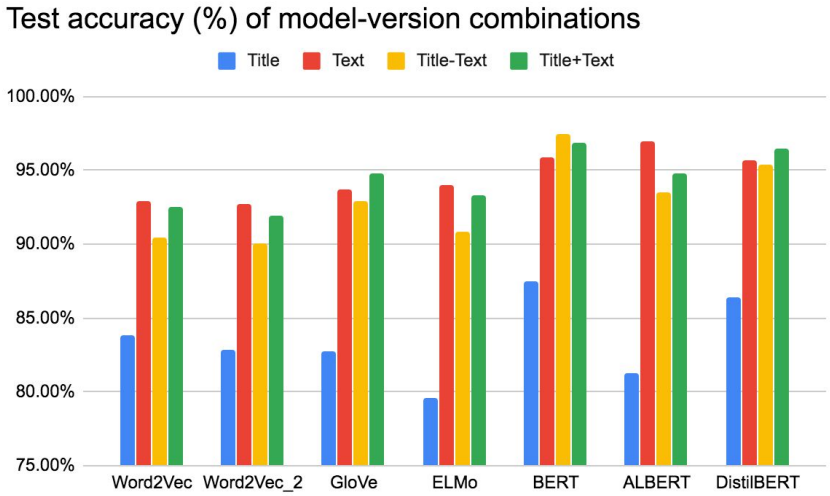
Discussion (Phase 1)



Version Analysis:

- Text, Title+Text > Title-text > Title
- Title is the most inferior, the headline of most fake news are made to look similar to real news to attract attention
- Title-Text performs better than Title but fell short against Text and Title+Text, the title takes up space which could be made for more text instead
- Text and Title+Text performed the best, the text in the Title+Text version dominates the title

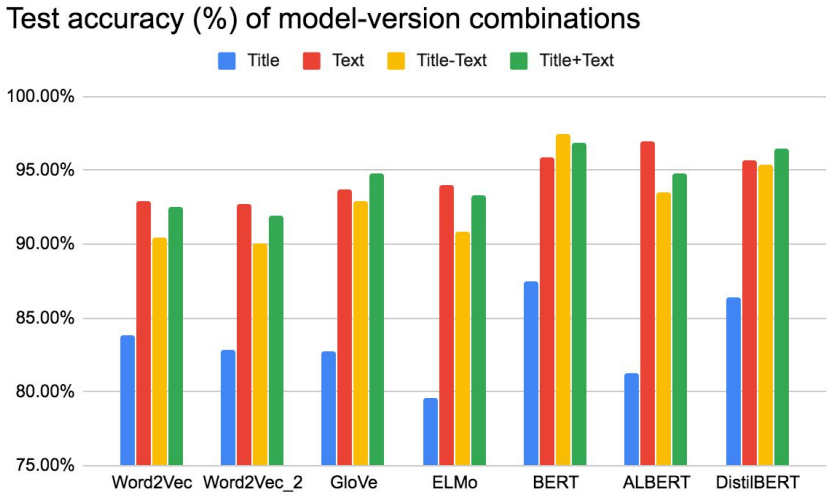
Discussion (Phase 1)



Embedding Analysis:

- GloVe generally performed better, it is pretrained on Gigaword-5 which contains mostly news information
- ELMo performed better than the Word2Vec baselines but worse than GloVe because of frozen weights during training and lower vector dim 256 compared to Glove's 300
- Word2Vec with 300 dimensions outperforms Word2Vec_2 with 256 dimensions by a slight margin due to higher representational power

Discussion (Phase 1)



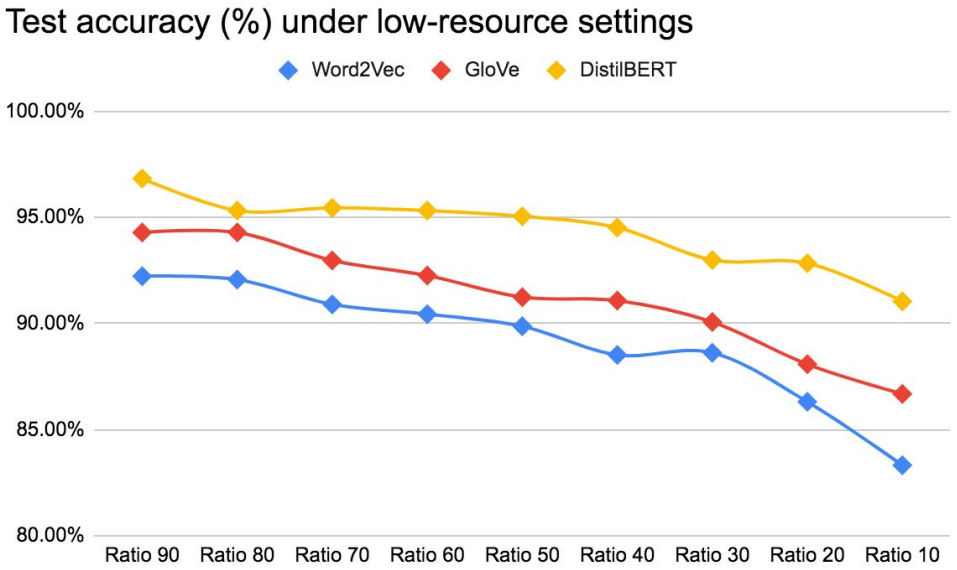
Transformer Analysis:

- BERT had the best performances in all versions, it contains the most parameters
- ALBERT although performs generally well, its parameter reduction techniques might have caused it to lose some representational power
- DistilBERT performed better than ALBERT in most cases but slightly worse than BERT, its primary advantage is being faster and cheaper to train than BERT while also keeping most of BERT's characteristics

Discussion (Phase 2)

Ratio Settings:

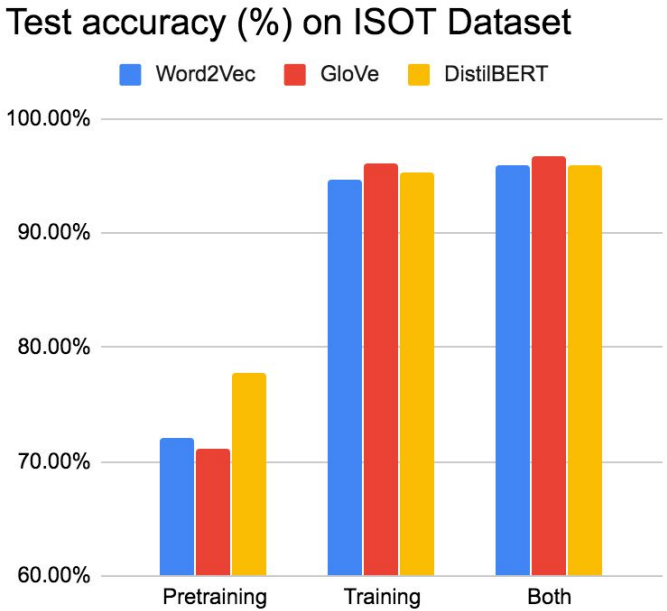
- With less training data, the performance of the models will decrease
- For each ratio setting: DistilBERT > GloVe> Word2Vec
- To achieve same level of performance, GloVe 80% less data than Word2Vec and 70% less data than GloVe



Discussion (Phase 3)

ISOT Fake News Dataset:

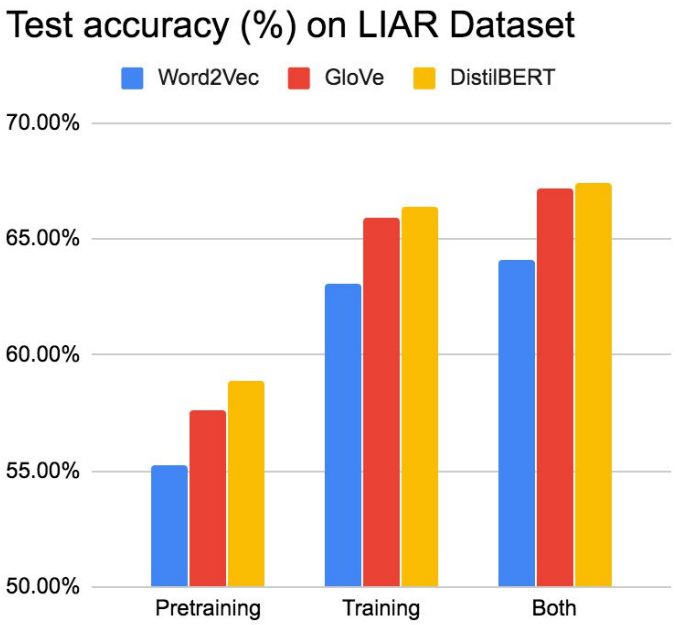
- pretrain+finetuning > train > pretrain
- pretraining and then fine-tuning worked better than training without any pretraining, however only by a slight margin
- The two datasets vary in news topics
- Pretraining without fine-tuning is insufficient: even the best-performing DistilBERT model gives under 80% test accuracy



Discussion (Phase 3)

LIAR Dataset:

- pretrain-train model > train model > pretrain
- The models perform significantly worse on this dataset than the previous ISOT dataset
- Shows that Real and Fake News Kaggle Dataset and LIAR Dataset is vastly dissimilar and hence pretraining does not really help



Conclusion

Investigated different **embeddings** and **transformers** on fake news detection

- **Phase 1**: Text and Title+Text versions work equally well due to the Text dominance, GloVe outperforms other embedding approaches, BERT performs the best in transformers, with DistilBERT slightly behind
- **Phase 2**: DistilBERT can robustly withstand low resource settings even when we have only several hundreds of examples (10% of the original dataset)
- **Phase 3**: pretraining on a similar dataset helps but only minimally, main requirements for good pretraining are for the two datasets to be very similar and both of high quality