“Fake News” Detection: An Exploration of Using Natural Language Processing Methodologies to Identify the Degree and Presence of Misinformation

Florencia Froebel, Sarah Hoover, and Ifrah Javed

Abstract

With an increase of information-sharing platforms and widespread accessibility to them, anyone has the capability to share information and have it reach millions of people. Although beneficial, this also lends to widespread misinformation, coined as “Fake News”. This paper builds on current research to develop a tool that can utilize linguistic features to identify the degree of false information in a piece of text, specifically focusing on political claims in the LIAR dataset. Our results demonstrate that the encoder layer of the pre-trained T5 transformer model yields improvement over BERT-based architectures. We also show that contextual data are valuable features when applied to a six-way classification task, however incorporating emotion-based features does not provide any additional improvement.

1 Introduction

This section covers project motivation and highlights the problem statement.

Information travels incredibly quickly currently with tools such as social media. While this speed surely has its benefits, it does not necessarily mean that all the information is correct. This was very apparent during the COVID pandemic, with misinformation about the disease and treatment causing political strife, exacerbating the spread of disease and in some cases leading to death. Our goal is to develop a tool to help identify instances of misinformation to prevent the publication and rapid spread of it.

Incorrect claims, referred to as “Fake News” are increasingly common, given that there are widely accessible public platforms. Widespread misinformation can have severe consequences, especially in the case of incorrect claims being made regarding health issues and politics. Previous NLP-based approaches to this problem can be broadly categorized into two groups: information retrieval techniques to verify the underlying claim based on a source text and the linguistic features of the statement. In this paper, we focus primarily on linguistic features and emotion/tone features to determine the degree to which a statement is false.

We leverage the LIAR dataset, a benchmark dataset for fake news detection. Early work on the LIAR dataset employed CNN [1] and LSTM [2] architectures, while more recent work incorporates pre-trained transformers such as BERT. [3, 4]. The dataset was chosen for its fine-grained labels, which reflect the degree of falseness. Most prior work on LIAR has reported findings for both binary and six-way classification tasks. However state of the art performance for a six-way task is very low at around 36-49% [5,2,8]. Our work focuses on the six-way classification task because we believe it is important to capture the nuance in statements and to be able to identify the degree to which a statement is false. Further, linguistic characteristics are well-suited to distinguishing the degree of falseness since exaggerated language can be the difference between a true statement and a mostly-true one. We build on the prior work done with this dataset by applying the T5 pre-trained transformer to LIAR for the first time and providing a side-by-side comparison of modeling techniques using BERT and T5.

2 Relevant Background Work

This section covers relevant background research that informed the model building process.

Previous work has demonstrated that linguistic cues can help identify false statements. Rashkin, et al. found that false news articles and statements tend to use more exaggerated and vague language [6]. Research by Mackey, et al. also demonstrated that emotion-based features improved a classification task using full-length news articles [7].

Several well-performing models on the LIAR dataset have incorporated pre-trained transformers. The work by [3] and [5] achieved accuracies of

37.1% and 37.4%, respectively, for six-way classification using BERT. Other pre-trained transformers such as RoBERTa, GPT2, and Funnel have shown promise [8]. No known published research has applied T5 to LIAR, however Gurrapu, et al. showed that using T5 yielded better performance over BERT for a similar fake news classification task [4]. Based on this research, the scope of potential models was expanded to also iterate on a T5 model as this is a novel architecture application to the LIAR dataset.

3 Dataset

This section describes the dataset used to train and evaluate models.

The LIAR dataset consists of 12,386 political claims, labeled by experts at Politifact.com [1]. There are six different labels, listed in ascending order of untruthfulness: “true”, “mostly true”, “half true”, “barely true”, “false”, and “pants-fire”. The labels are relatively balanced, however “pants-fire” is the smallest category. A complete breakdown of the labels is provided in the appendix. Example statements are provided in Figure [1].

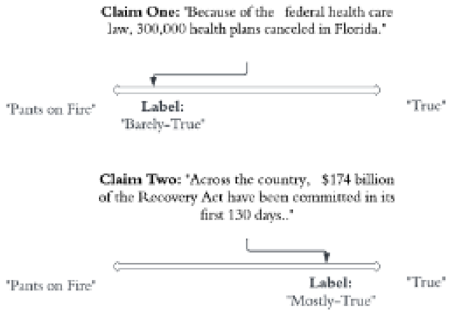

Claim One: "Because of the federal health care law, 300,000 health plans canceled in Florida."
"Pants on Fire"
Label:
"Barely-True"
"True"
Claim Two: "Across the country, $174 billion of the Recovery Act have been committed in its first 130 days.."
"Pants on Fire"
Label:
"Mostly-True"
"True"

Figure 1: Examples of claims in the dataset and their corresponding labels.

The dataset also includes the name of the individual or entity who made the claim, the political party they are associated with, and keywords identifying the topic of the claim (e.g., healthcare). These additional features are referred to collectively as the statement metadata.

The written justification for each label is also available, as extracted and published by [9] and dubbed the “LIAR-Plus” dataset. This justification was written by the person determining the label at the time they evaluated the claim. We decided to exclude the justification from our analysis because it does not provide an accurate representation of information that is available at the time a statement is made.

4 Overall Approach

The overall approach was to compare the type of model being used and generate additional features, specifically, features that categorize the emotions present in the claim. Model metrics are outlined as well.

A BERT-based model with a fully connected layer and classification head was selected as the baseline model, given the growing popularity of pre-trained transformers and the previous work on LIAR using BERT. Our strategy was to improve performance by progressively adding features to our baseline, while adding these same features to a T5-based model.

In addition to evaluating model types, we built and incorporated features that give non-linguistic information. This included providing context for the claims from the available metadata, and deriving additional features from the claims themselves. Research done by Rashkin, et al. [6] and Mackey, et al. [7] showed model improvement on a similar task using features like this.

4.1 Feature Building

We fed six sets of features into our models.

Feature Set One: This is the “Claim”, which is just text.

Feature Set Two: This is the “Claim” as well as the concatenated metadata from the other column in the dataset (claim topic, speaker of the claim, and party affiliation of the claim speaker).

Feature Set Three: Two packages, EmoLex [10] and LIWC [11], were used to extract information about emotion and other lexical features present in the claim. These features were selected based on previous work done by [9]. These packages quantify the amount of certain emotions present in the claim. For example, a feature added by using these quantifies the amount of “disgust” present in the claim text. These features were treated as additional inputs to the model. They were concatenated to either the CLS or pooler token for BERT, or a pooled vector from the last hidden state of the T5 encoder. This feature set includes the “Claim”, the concatenated metadata, and the features derived from EmoLex and LIWC.

Feature Set Three Subsets: LIWC and EmoLex collectively added nearly 150 features to the

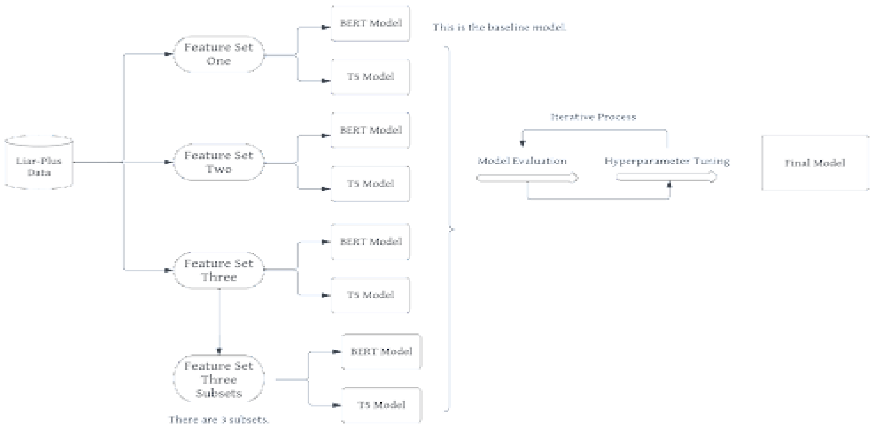


Figure 2: Diagram of model building strategy. Each feature set is defined in Section 4.1.

dataset, resulting in a sparse matrix. Further iterations utilized subsets of these features along with the claim and concatenated metadata. A list of extreme emotions was identified to focus our analysis, including "fear", "joy" and "disgust". We further split this list into a feature set with just the extreme positive emotions and another feature set with just the extreme negative emotions. These feature sets are outlined in the appendix.

4.2 Model Building

We iterated on BERT and T5 based models in parallel. We first trained a traditional T5 model using the text-to-text architecture. We trained a model using only the encoder layer from T5 to replicate the structure of our baseline. Further iterations to the T5 model built on this encoder-only version in order to incorporate the numeric features. Then, we tested the six input sets described in Section 4.1. The overall flow of the experiments can be seen in Figure [2].

4.3 Evaluation Criteria

We created a 60/20/20 split for the train, test and validation datasets. The success of each model created was tracked by the following metrics: Validation Accuracy, Test Accuracy, Precision, Recall and F1 Score. The main metrics of interest to compare models were the Weighted F1 score. Validation accuracy was used for hyperparameter tuning and to check for overfitting. All of these metrics are available in the Appendix, with the exception of validation accuracy.

5 Results and Analysis

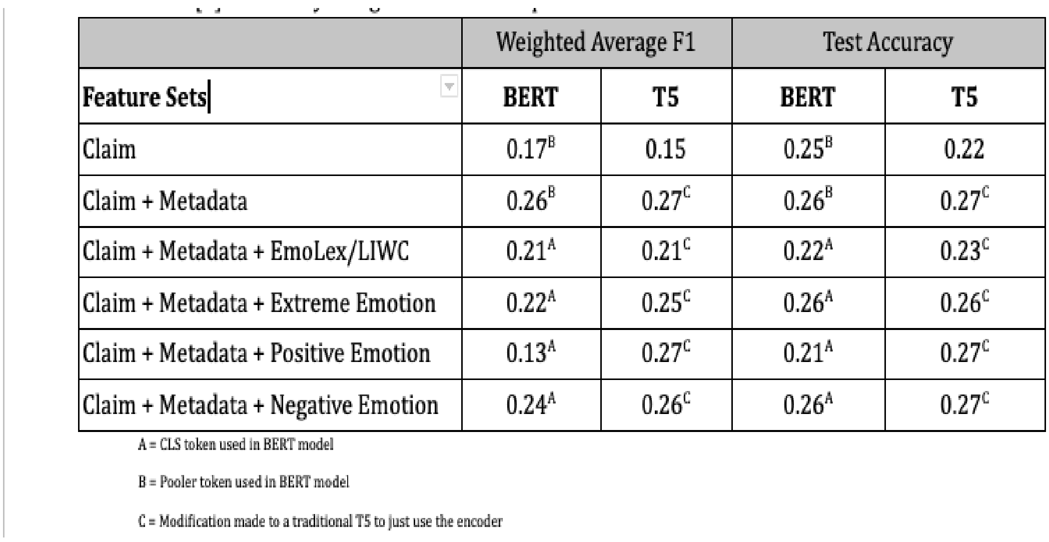
The F1 score and test accuracy are reported for the most informative models below in Table. Refer to the Appendix for a more complete list of the total model iterations run. All metrics shown in the table are generated on the same train/test and validation sets.

Each set of features has a corresponding BERT-based model and a T5 model. All BERT-based models were tested with both the CLS token and a pooler token, and the better-performing model was selected. Results are shown in Figure [3].

These low metrics are to be expected, given the prior work done on this dataset. For example, the original CNN architecture proposed by Wang when creating the LIAR dataset reached an accuracy of only 27%. Most of the highest-performing models incorporate the label justification from LIAR-plus, which we deliberately excluded from our analysis. To our knowledge, only Samadi, et al. have achieved an accuracy above 30% on a six-way classification task for this dataset without incorporating the justification [8].

The baseline model which is just the claim text fed into a basic BERT model had a test accuracy of 25% and an F1 score of 0.17. This model failed to make any predictions in the "pants-fire", "barely-true", and "true" categories. The basic T5 model with the same input had a worse accuracy at 22% as well as a slightly lower F1 score at 0.15. However, the corresponding T5 model produced predictions for all labels except the "pants-fire" class.

We next incorporated the available metadata. These text-based features were concatenated with the main claim and fed into each model. These fea-



	Weighted Average F1		Test Accuracy	
Feature Sets	**BERT**	**T5**	**BERT**	**T5**
Claim	0.17^B	0.15	0.25^B	0.22
Claim + Metadata	0.26^B	0.27^C	0.26^B	0.27^C
Claim + Metadata + EmoLex/LIWC	0.21^A	0.21^C	0.22^A	0.23^C
Claim + Metadata + Extreme Emotion	0.22^A	0.25^C	0.26^A	0.26^C
Claim + Metadata + Positive Emotion	0.13^A	0.27^C	0.21^A	0.27^C
Claim + Metadata + Negative Emotion	0.24^A	0.26^C	0.26^A	0.27^C

A = CLS token used in BERT model

B = Pooler token used in BERT model

C = Modification made to a traditional T5 to just use the encoder

Figure 3: Summary of significant models produced.

tures improved the F1 score by 0.09 for the BERT model and by 0.12 for T5. With these models, the T5 performed marginally better than the BERT model. The concatenated metadata was used as the input to further iterations on the model due to this improvement in performance.

BERT and T5 performed equally well with the addition of LIWC and EmoLex. These features improved performance compared to the baseline model, however performance declined compared to using only the concatenated metadata as input. We sought to improve the utility of the LIWC and EmoLex features by focusing on a subset of 16 features most relevant to our task. This change marginally improved model performance compared to including the full range of features. These 16 features were further split into emotions that have a positive connotation and emotions that have a negative connotation. Focusing on just the positive emotions considerably worsened performance for BERT, however T5 improved slightly. While not shown in the results table, removing the metadata from the BERT model with positive emotions produced an F1 score of 0.21 - much closer to previous iterations. Including only negative emotion-based features improved both BERT and T5 over the extreme emotions.

The T5 based model performed better than its BERT model companion in all but two experiments. A particularly interesting finding is that employing only the encoder layer from T5 yielded an improvement over BERT, however utilizing the full encoder-decoder architecture did not outperform the BERT baseline.

In several experiments, T5 was able to predict a broader range of classes than BERT. For example, the BERT model with concatenated metadata and positive emotions did not predict any labels in the "barely-true" category. In contrast, the T5 equivalent was able to predict some of the "barely-true" labels correctly.

Overall, the T5 model with the claim and concatenated data as feature inputs had the best performance. While focusing on a subset of emotions produced better results than including the full range, the emotion-based features did not provide any value beyond the claim + concatenated metadata on their own.

6 Conclusion

Through model iteration there are three main conclusions that can be drawn: 1) Using the encoder layer of T5 for modeling the Liar dataset yields a marginal increase in performance over BERT models 2) Adding in contextual information about the text in a concatenated form improves performance the most 3) Including a targeted list of extreme emotions, especially ones with a negative connotation, yield a significant improvement in model performance over including a more complete range. In terms of future work, there is room for improvement. Future work could build on this research by incorporating additional features in a tabular fashion using TaBERT [12] and TaBT5 [13]. In particular, TaBT5 would be a promising avenue to explore since it could facilitate the implementation of the full encoder-decoder architecture of T5.

7 Acknowledgements

We would like to thank Daniel Cer, Jennifer Zhu, Mark Butler, and Aswin Thiruvengadam for their helpful input during office hours.

8 References

1. William Yang Wang. 2017. “Liar, Liar Pants on Fire: A New Benchmark Dataset for Fake News Detection." In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 422–426, Vancouver, Canada. Association for Computational Linguistics.

2. Yunfei Long, Qin Lu, Rong Xiang, Minglei Li, and Chu-Ren Huang. 2017. "Fake News Detection Through Multi-Perspective Speaker Profiles." In Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers), pages 252–256, Taipei, Taiwan. Asian Federation of Natural Language Processing.

3. Mehta, D., Dwivedi, A., Patra, A. et al. "A transformer-based architecture for fake news classification." Soc. Netw. Anal. Min. 11, 39 (2021). https://doi.org/10.1007/s13278-021-00738-y

4. S. Gurrapu, L. Huang and F. A. Batarseh, "ExClaim: Explainable Neural Claim Verification Using Rationalization," 2022 IEEE 29th Annual Software Technology Conference (STC), Gaithersburg, MD, USA, 2022, pp. 19-26, doi: 10.1109/STC55697.2022.00012.

5. Shaily Bhatt, Naman Goenka, Sakshi Kalra, and Yashvardhan Sharma. 2021. "Fake News Detection: Experiments and Approaches Beyond Linguistic Features." Data Management, Analytics and Innovation, pages 113-128, https://doi.org/10.48550/arXiv.2109.12914.

6. Hannah Rashkin, Eunsol Choi, Jin Yea Jang, Svitlana Volkova, and Yejin Choi. 2017. "Truth of Varying Shades: Analyzing Language in Fake News and Political Fact-Checking." In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 2931–2937, Copenhagen, Denmark. Association for Computational Linguistics.

7. A Mackey, Susan Gauch, and Kevin Labille. 2021. "Detecting fake news through emotion analysis." In Proceedings of the 13th International Conference on Information, Process, and Knowledge Management. 65–71.

8. Mohammadreza Samadi, Maryam Mousavian, Saeedeh Momtazi. 2021 "Deep contextualized text representation and learning for fake news detection." 2021. Information Processing Management, Volume 58, Issue 6, https://doi.org/10.1016/j.ipm.2021.102723.

9. Tariq Alhindi, Savvas Petridis, and Smaranda Muresan. 2018. "Where is Your Evidence: Improving Fact-checking by Justification Modeling." In Proceedings of the First Workshop on Fact Extraction and VERification (FEVER), pages 85–90, Brussels, Belgium. Association for Computational Linguistics.

10. Saif M Mohammad and Peter D Turney. 2010. "Emotions evoked by common words and phrases: Using mechanical turk to create an emotion lexicon." In Proceedings of the NAACL HLT 2010 workshop on computational approaches to analysis and generation of emotion in text, pages 26–34. Association for Computational Linguistics.

11. James W Pennebaker, Martha E Francis, and Roger J Booth. 2001. "Linguistic inquiry and word count: Liwc 2001." Mahway: Lawrence Erlbaum Associates, 71(2001):2001.

12. Pengcheng Yin, Graham Neubig, Wen-tau Yih, and Sebastian Riedel. 2020. CoRR, Vol. abs/2005.01856. https://doi.org/10.48550/arXiv.2005.08314

13. Ewa Andrejczuk, Julian Martin Eisenschlos, Francesco Piccinno, Syrine Krichene, and Yasemin Altun. "Table-To-Text generation and pre-training with TabT5." 2022. https://doi.org/10.48550/arXiv.2210.09162

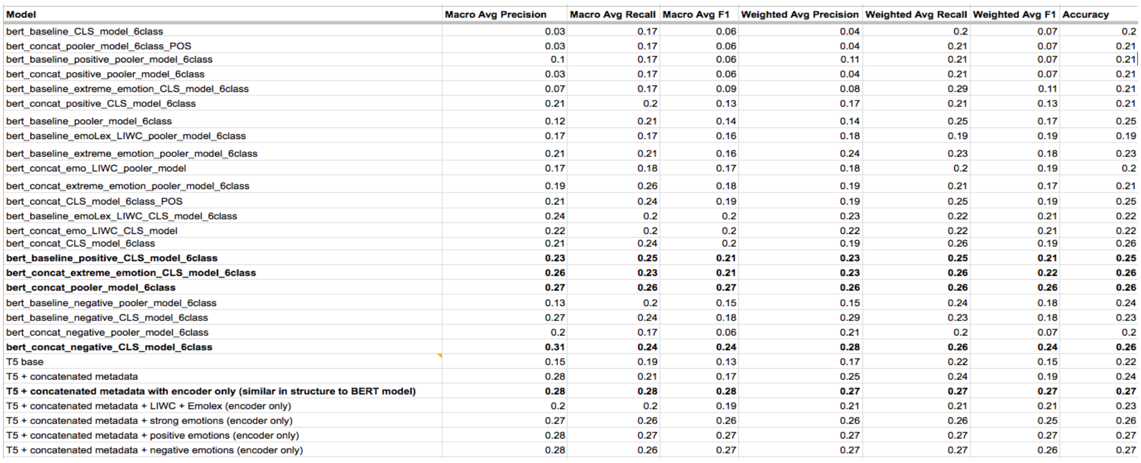
9 Appendix

Label	Number of Samples
"pants-fire"	1,050
"false"	2,511
"barely true"	2,108
"half true"	2,638
"mostly true"	2,466
"true"	2,063

The distribution of labels in the LIAR dataset.

Feature Name	Subgroup
fear	Negative
anger	Negative
anticipation	Negative
trust	Positive
surprise	–
positive	Positive
negative	Negative
sadness	Negative
disgust	Negative
joy	Positive
power	–
emo_pos	Positive
emo_neg	Negative
emo_anx	Negative
emo_anger	Negative
emo_sad	Negative

The features selected for extreme, positive, and negative emotion analysis from LIWC and EmoLex.



Model	Macro Avg Precision	Macro Avg Recall	Macro Avg F1	Weighted Avg Precision	Weighted Avg Recall	Weighted Avg F1	Accuracy
bert_baseline_CLS_model_6class	0.03	0.17	0.06	0.04	0.2	0.07	0.2
bert_concat_pooler_model_6class_POS	0.03	0.17	0.06	0.04	0.21	0.07	0.21
bert_baseline_positive_pooler_model_6class	0.1	0.17	0.06	0.11	0.21	0.07	0.21
bert_concat_positive_pooler_model_6class	0.03	0.17	0.06	0.04	0.21	0.07	0.21
bert_baseline_extreme_emotion_CLS_model_6class	0.07	0.17	0.09	0.08	0.29	0.11	0.21
bert_concat_positive_CLS_model_6class	0.21	0.2	0.13	0.17	0.21	0.13	0.21
bert_baseline_pooler_model_6class	0.12	0.21	0.14	0.14	0.25	0.17	0.25
bert_baseline_emoLex_LIWC_pooler_model_6class	0.17	0.17	0.16	0.18	0.19	0.19	0.19
bert_baseline_extreme_emotion_pooler_model_6class	0.21	0.21	0.16	0.24	0.23	0.18	0.23
bert_concat_emo_LIWC_pooler_model	0.17	0.18	0.17	0.18	0.2	0.19	0.2
bert_concat_extreme_emotion_pooler_model_6class	0.19	0.26	0.18	0.19	0.21	0.17	0.21
bert_concat_CLS_model_6class_POS	0.21	0.24	0.19	0.19	0.25	0.19	0.25
bert_baseline_emoLex_LIWC_CLS_model_6class	0.24	0.2	0.2	0.23	0.22	0.21	0.22
bert_concat_emo_LIWC_CLS_model	0.22	0.2	0.2	0.22	0.22	0.21	0.22
bert_concat_CLS_model_6class	0.21	0.24	0.2	0.19	0.26	0.19	0.26
bert_baseline_positive_CLS_model_6class	**0.23**	**0.25**	**0.21**	**0.23**	**0.25**	**0.21**	**0.25**
bert_concat_extreme_emotion_CLS_model_6class	**0.26**	**0.23**	**0.21**	**0.23**	**0.26**	**0.22**	**0.26**
bert_concat_pooler_model_6class	**0.27**	**0.26**	**0.27**	**0.26**	**0.26**	**0.26**	**0.26**
bert_baseline_negative_pooler_model_6class	0.13	0.2	0.15	0.15	0.24	0.18	0.24
bert_baseline_negative_CLS_model_6class	0.27	0.24	0.18	0.29	0.23	0.18	0.23
bert_concat_negative_pooler_model_6class	0.2	0.17	0.06	0.21	0.2	0.07	0.2
bert_concat_negative_CLS_model_6class	**0.31**	**0.24**	**0.24**	**0.28**	**0.26**	**0.24**	**0.26**
T5 base	0.15	0.19	0.13	0.17	0.22	0.15	0.22
T5 + concatenated metadata	0.28	0.21	0.17	0.25	0.24	0.19	0.24
T5 + concatenated metadata with encoder only (similar in structure to BERT model)	**0.28**	**0.28**	**0.28**	**0.27**	**0.27**	**0.27**	**0.27**
T5 + concatenated metadata + LIWC + Emolex (encoder only)	0.2	0.2	0.19	0.21	0.21	0.21	0.23
T5 + concatenated metadata + strong emotions (encoder only)	0.27	0.26	0.26	0.26	0.26	0.25	0.26
T5 + concatenated metadata + positive emotions (encoder only)	0.28	0.27	0.27	0.27	0.27	0.27	0.27
T5 + concatenated metadata + negative emotions (encoder only)	0.28	0.26	0.27	0.27	0.27	0.26	0.27

A comprehensive table for all evaluations and model building.