# *A Novel Ensemble Representation Framework for Sentiment Classification*

Mengtao Sun
*Faculty of Information Technology and Electrical Engineering*
*Norwegian University of Sciences and Technology*
Aalesund, Norway
mengtao.sun@ntnu.no

Ibrahim A. Hameed
*Faculty of Information Technology and Electrical Engineering*
*Norwegian University of Sciences and Technology*
Aalesund, Norway
ibib@ntnu.no

Hao Wang
*Faculty of Information Technology and Electrical Engineering*
*Norwegian University of Sciences and Technology*
Gjøvik, Norway
hawa@ntnu.no

***Abstract*—Text representation has a critical impact on the accuracy of text classifiers which is imperative to be strengthened. On the other hand, the question of how the state-of-the-art embeddings outperform previous approaches cannot be well explained. To advance text representation and better understand the internal mechanism, we propose a novel end-to-end framework named *Ensemble Framework for Text Embedding* (EFTE), which weightedly combines diverse embeddings and simultaneously represents sentences' and tokens' features in a more reasonable way. According to the experimental results in sentiment classification, our proposed embedding apparently improves the effectiveness compared to six single embeddings. Moreover, the importance of each embedding in terms of EFTE integration and how different embeddings influence the results by classification are discussed.**

***Keywords— text representation, ensemble, sentiment analysis***

## I. INTRODUCTION

Text representation is highly influential to the performance of text classification. Those representation algorithms can be classified into three categories: Vector Space Model (VSM), Theme Model (TM) and Neural Network Model (NNM). According to [1], in NNM, a representation based on simple neural network outperforms all of the VSM and TM based approaches. Especially, a novel self-attentive structure [2] in NNM is growing rapidly and shows the best capabilities in feature extraction through numerous real-world tasks [3-5].

Apparently, single embedding is impossible to vectorize all the features from the text. A common approach is to concatenate two embeddings, as it is shown by equation (1), where $h$ and $h'$ are two types of embeddings.

$$F = h \oplus h' = \{h_1, h_2, \dots, h_n, h'_1, h'_2, \dots, h'_n\} \quad (1)$$

However, a token may be plausible in original algorithm but unable to be featured in their combination. For example, by one type embedding, a token "*bank*" can be represented as a "*river levee*", by another type of embedding as a "*financial organization*", both two embeddings are informative in separated but unfeatured in their combination. Moreover, the weights of each token should be normalized since multiple embeddings are in different value scales. To the best of our knowledge, there is no such good ensemble approach aiming at representations.

Several drawbacks appear when we try to find the best possible integration. Firstly, it is hard to deal with the level-gap. As it is highlighted by [6], texts comprise both sentence-level and tokens-level representations which respectively embeds in non-unified dimensional tensors. Secondly, it is problematic to assemble with irregular dimensions. Each model have unique standards on feature scope so it is impossible to align with each other. Thirdly, it is tedious to leverage the weights in unequal embeddings, as it is emphasized by [7], sentence embedding is more comprehensive than tokens. Similarly, it is irrational to put BERT (Bidirectional Encoder Representations from Transformers) [8] and ELMO (Embeddings from Language Model) [9] embeddings on an equal place. Thus, a novel structure is required to enable us to make individual embeddings working tendentiously.

Motivated by the line of fusion structure in many successful cases, we propose a novel ensemble framework to address these problems in this paper, namely *Ensemble Framework for Text Embedding* (EFTE). EFTE does not merely rely on given embeddings but is a general framework for diverse levels and types of embeddings. In this paper, EFTE is applied for Norwegian review classification. This is because Natural Language Processing (NLP) studies in Norwegian are insufficient. In particular, few sentiment analysis efforts have been put into Norwegian language. Practically, this method is applicable to any language.

BERT and ELMO show good advantages as backbone. First of all, BERT can well represent synonyms and polysemy and shows the state-of-the-art performance in classification [8]. Secondly, it has shown to be efficient in more than a hundred of languages [10]. In addition, ELMO is more capable to represent contextual semantics in time series [9] which has been successfully applied for emotional perceptron in Norwegian language [11]. Intuitively, the ensemble of BERT and ELMO can supplement information to each other and improve the final performance in classification.

In this paper, results of six simple Norwegian language embeddings are analyzed and compared with EFTE. Besides,

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

the importance of each embedding in terms of EFTE integration and how different embeddings influence the results by classification are discussed. Our contributions are summarized as follows:

- We design a novel ensemble structure called EFTE. It solve the problem of embeddings with non-aligned dimensions and automatically achieves the ensemble from diverse embedding.
- We make a series of experiments on Norwegian Review Corpus. The results show our mixture embedding outperforms other six single Norwegian embeddings.
- We evaluate the parameters in EFTE to measure the importance in separated embedding. We also force the weights distribution changes and discuss the influence of each embedding on results.

The rest of this paper is organized as follows: Section II investigate the recent developments of embeddings and ensembles. Section III presents the proposed structure and configuration of EFTE; reviewing of fusion operations in (A); the essential principle, route, and some questions solved in EFTE in (B); detailed description of how each layer works in EFTE in (C), and finally training methods of EFTE network are given in (D). Section IV presents the experiments on NoReC including data preprocessing, metrics, baseline models and results. Concluding remarks and discussions are given in Section V.

## II. Related Work

### A. Embedding

Text embedding is defined as a mapping from text to vectorization. The seminal embedding derive from Vector Space Model (VSM), each component corresponds to a word term, which is equivalent to representing text as a point in space. Text vector can not only be used to train classifiers, but also calculate the similarity between texts. Recently, a few works still focus on VSM, such as [12] and [13] that advanced the term frequency inversed document frequency (TF-IDF) algorithms by domain knowledge.

Researchers have tried to interpret text from the perspective of probability generation, i.e. Theme Model (TM). In this representation, each dimension stands for a "topic" formed as a group of word clustering. Here, the semantics are represented by each dimension which contains one perspective of explanatory information. Followed by Latent Dirichlet Allocation (LDA), many variants are developed. For instance, [14] used TF-IDF and LDA to enhance expert and intelligent systems based on similar words, especially when there is a scarcity of labeled texts. [15] employed both document-level and segment-level specific topic distributions to capture fine-grained differences in topic assignments. Their model also combines other empirical LDA-based models and shows the outcomes in 6 public datasets.

Neural Network Model (NNM) has developed rapidly over the last several years and has exceeded most statistical models. NNM came from Word2Vec [16] by Mikolov, and then further developed in Doc2Vec [17] and fastText [18]. However, basic NN is insufficient to produce embeddings. Many complex models are therefore proposed such as CNN [19], RNN embedding [20] and its variants including LSTM [21-22] and GRU [23-24] embeddings. ELMO is one of the best embeddings which encompasses two bi-LSTM layers for contextualize information. It is easy to be integrated to existing models, which significantly improved the results in 6 NLP problems [9]. Google in 2017 proposed a novel structure named Transformer [2] with multiple self-attentive layers for sequence-to-sequence applications, which is suitable for text representation. With the multiple self-attentive layers, the pre-trained model is possible to be fine-tuned with an additional output layer to significantly improve the performance in many applications [8].

### B. Ensemble structure

The essence of ensemble is based on the fusion structure in neural network [25]. Seq2seq model with attention mechanism makes the model automatically focus on concrete utterance from text [26]. They ingeniously create an attention distribution and combine it with encoder to choose different segments from each time step during training. Attention makes decoder understand which portion is important and should be concentrated. Followed by attention, many dynamic networks are designed, where the ground-truth is joint after feature extraction. For example, [27] has successfully equipped text summarization with the previous words by pointer-generator attention. [28] presented a new structure named “Hopping” to repeatedly train, calculate attention and add it into encoder.

Ensemble structure also associates with some applications. To name a few, [29] has successfully united emotion embedding, semantics embedding, and randomly initialized embedding to create a chat system, which generates human-like emotional responses. [30] has combined available multimodal cues from videos for the cross-modal video-text retrieval task. [31] has mixed LDA and GRU-CNN text representation for sentiment classification.

In this paper, we thoroughly study several fusion structures in aforementioned neural network and put a new integrated embedding for classification.

## III. Algorithms

### A. Preliminary operation in fusion

Current evidences support that fusion methods do not follow a fixed rule. For clearness, here we list the most common equations used in fusion neural network:

$$F = h \oplus h' \quad (1)$$

$$F = h + h' \quad (2)$$

$$F = h \circ h' \quad (3)$$

$$F = h^T W h' \quad (4)$$

$$F = |h - h'| \quad (5)$$

$$F = NN(h, h') \quad (6)$$

Where $F$ is a merged tensor, $h$ and $h'$ represent two different tensors in hidden layer, $\circ$ is element-wise multiplication, $W$ is trainable weight. $NN(\cdot,\cdot)$ denote a neural network with same dimensional output with $h$ and $h'$. The outcomes of equation (2) is applied in residual neural network

[32]. It is also possible to use multiple equations in serial or parallel calculation. For example, [33] has combined nine operations in parallel using equation (7):

$$F(c,m,q) = [c, m, q, c \circ q, c \circ m, |c-q|, |c-m|, c^T W q, c^T W q] \quad (7)$$

When the merged output is a scalar, such as a weight or probability, the calculation are:

$$S = h \cdot h' \quad (8)$$

$$S = \frac{h \cdot h'}{\|h\| \|h'\|} \quad (9)$$

Scalar $S$ in equation (9) is also known as cosine similarity which is used as weighting coefficient in attention [26].

## B. Overview and definitions

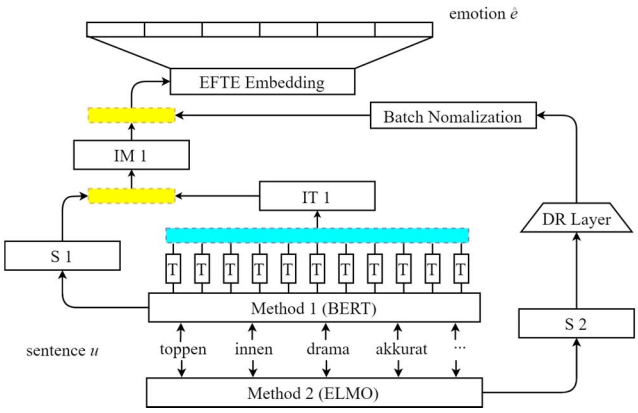


Fig. 1. Structure of EFTE network

Our task, here, is to perceive the emotion in a production review corpus. That is, given an input sentence $u$ with its corresponding emotion $e$, EFTE output prediction $\hat{e}$. The integrated embedding $y = \{y_1, y_2, \dots, y_m\}$ can be obtained by the hidden layer after training.

EFTE as a general framework is explained in Fig. 1. Method 1 yield text embedding at first. Tokens ($T_1$) are formed as Integration ($IT_1$) and then combines Sentence ($S_1$) to be $IM_1$ (Integrated embedding by Method 1). Subsequently, Method 2 produces Sentence ($S_2$) and feeds $S_2$ to Dimensional Reduction (DR) layer which compresses $S_2$ to match the dimension of $IM_1$. In this way, $IT_1$, $S_2$ and $IM_1$ are normalized to an identical dimension. Finally, EFTE embedding is received by the same fusion structure of $IT_1$ and $S_1$.

Hence, multiple embeddings complement with each other and guarantee the outcome of classification. Such embedding obviously contains more information compared to those obtained based on either BERT or ELMO from tokens and sentences. The weights in $T_1$-$T_1$, $IT_1$-$S_1$ and EFTE Embedding layer can also be manually controlled or monitored for analysis.

The essential components of EFTE are listed below:

- **BERT Layer.** This model yields $T_1$ and $S_1$.
- **$IT_1 \leftarrow T_1$ $T_1$.** This model integrates tokens $T_1$ to $IT_1$.
- **$IM_1 \leftarrow IT_1$ $S_1$.** This model is fed with $IT_1$, $S_1$ and yields $IM_1$.
- **ELMO Layer.** This model yields $S_2$.
- **DR Layer.** This layer is used for dimensional reduction which compresses dimension of $S_2$ to the same of $IM_1$.
- **EFTE Embedding $\leftarrow IM_1$ $S_2$.** $IM_1$, $S_2$, $IT_1$ have been in same level so that we straightforwardly apply the structure of $IT_1$-$S_1$ layer to obtain EFTE embedding.

## C. Detailed operations

*a) BERT:* We use BERT embedding as backbone. Given sentence $u$, tokens embedding $\{h_{bt}^1, h_{bt}^2, \dots, h_{bt}^{Lt}\}$ is calculated by the pretrained Norwegian language model released by [10].

Tokens in Norwegian are exemplified as follows:

| | |
|---|---|
| **Origin Text** | *toppen innen drama akkurat na !*<br>*(top in drama right now !)* |
| **Tokens** | *[CLS] toppen innen drama ak ##kura ##t na ! [SEP]* |

where [CLS] denotes starting token and presents sentence level embedding, defined as $h_{bs} = h_{bt}^1$. Semantical embedding $h_{bt} = \{h_{bt}^2, \dots, h_{bt}^9\}$ come from "*toppen*" to "*!*".

In literature, sentence level embedding $h_{bs}$ is used for classification. In EFTE, $h_{bt}$ is also considered to supplement tokens-level information.

*b) $T_1$ $T_1$ Layer:* This model feeds tokens $h_{bt}$ to output integration $h_{bi}$. As it is shown in Fig. 2, the darker triangle is, the more weight corresponds in token.

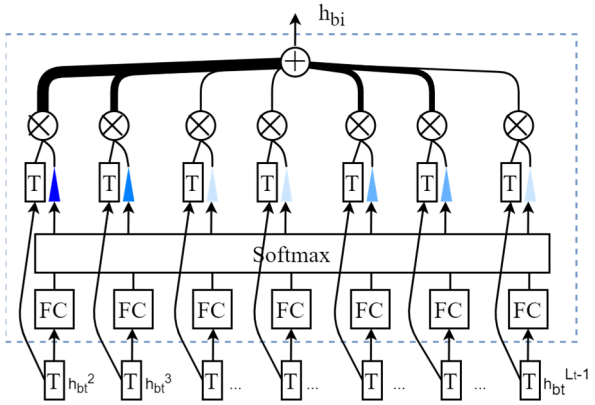


Fig. 2. $T_1$ $T_1$ Layer

Self-attentive layer derived from [34]. In this paper, we refer to their works but simply adopt a spectrum of feedforward networks to preserve the features from BERT, each network output a scalar. The integration $h_{bi}$ is calculated by equation (10) and (11).

$$h_{bi} = \sum_{i=2}^{L_t-1} G_i \cdot h_{bt}^i \quad (10)$$

$$G_i = softmax(W_{G_i} \cdot h_{bt}^i + b_{G_i})_{i\in[2,L_t-1]} \quad (11)$$

where each token $h_{bt}^i$ is limited by controlling matrix $G_i$. $W_{G_i}$ and $b_{G_i}$ are trainable parameters.

*c) $IT_1$ ▒ $S_1$ Layer:* Integrated token $h_{bi}$ and sentence $h_{bs}$ produce $h_b$ by this layer (depicted in Fig. 3). $h_b$ is computed by equation (12).

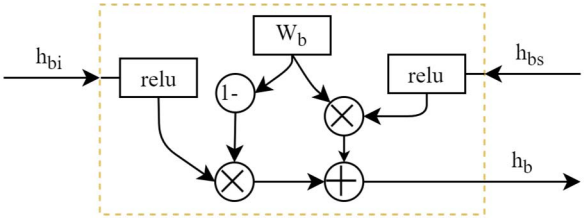


Fig. 3. $IT_1$ ▒ $S_1$ Layer

$$h_b = w_b \circ relu(h_{bi}) + (1 - w_b) \circ relu(h_{bs}) \quad (12)$$

where $w_b$ is a controlling parameter to balance token and sentence embeddings. $relu$ function is used to normalize each input.

*d) ELMO:* On the other route, sentence $u$ is embeded by ELMO and generate $\tilde{h}_{es}$, the calculations are concluded in equation (13-15)

$$I_1 = \overleftarrow{LSTM}_1(CharCNN(u)) \oplus \overrightarrow{LSTM}_1(CharCNN(u)) \quad (13)$$

$$I_2 = \overleftarrow{LSTM}_2(I_1) \oplus \overrightarrow{LSTM}_2(I_1) \quad (14)$$

$$\tilde{h}_{es} = \gamma(s_0 t + s_1 I_1 + s_2 I_2) \quad (15)$$

where $\overleftarrow{LSTM}$ and $\overrightarrow{LSTM}$ represent two direction. $CharCNN(u)$ is a character-based CNN for raw word vector, $s$ is $softmax$-normalized weights. Scalar $\gamma$ control the scale of $\tilde{h}_{es}$.

*e) DR Layer:* However, $\tilde{h}_{es}$ by ELMO is not in a same feature scope with BERT, dimension alignment is prerequisites before embedding ensemble. In general, the solution of inconsistent dimension is commonly conducted by padding and truncating. [35] described a Principal Component Analysis (PCA) neural network which has been successfully used in recent applications [36-37]. Thus, we apply PCA for dimension reduction $\tilde{h}_{es} \rightarrow h_{es}$ i.e. $Dim(h_{es}) = Dim(h_b)$.

Here, a batch normalization is used after dimension reduction. As it is mentioned in [9], the scale problem disastrously influents the results for many NLP tasks.

*f) $IM_1$ ▒ $S_2$ Layer:* Embedding $h_b$ and $h_{es}$ are in the same dimension after *DR layer*, the structure in Fig. 4 is therefore directly applied, i.e. feed $h_b$ and $h_{es}$ and recieve $y = \{y_1, y_2, \dots, y_m\}$, the calculation is shown in equation (16).

$$y = w_y \circ relu(h_b) + (1 - w_y) \circ relu(h_{es}) \quad (16)$$

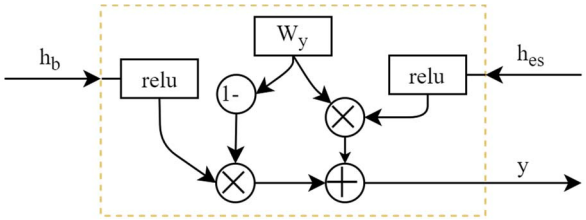


Fig. 4. $IM_1$ ▒ $S_2$ Layer

where the parameter $w_y$ controls the weight for each type of embedding.

## D. Training

Here, a pretrained BERT posted by Google which involves 110 million parameters and a pretrained ELMO posted by HIT-SCIR with 20 million parameters are directly applied. In order to preserve the information for further trainings, we froze the weight in BERT and ELMO layer and make $y$ connect a simple feedforward network with six outputs towards emotional degree. We calculate the cross entropy as the loss function, as it is shown by Equation (17).

$$Loss = -elog(\hat{e}) \quad (17)$$

where $e$ is the one-hot representation of the text emotion and $\hat{e}$ is the predicted emotion representation. The whole network can be jointly trained.

# IV. Experiments

## A. Preprocessing

Norwegian Review Corpus (NoReC) is presented by University of Oslo for training and evaluating models for document-level sentiment analysis [38]. Each document has a metadata which describes the extraction and emotion rate. According to the metadata, NoReC is also possible for training sentence-level network.

**NoReC**. This dataset consists of over 35,000 full-text reviews. Each review is rated with a numerical score on a scale of 1-6 for predicting positive or negative polarity for a given text. It derives from 8 websites consisting of the comprehensive reviews in 9 domains from 2003 to 2017 in Norway. Detailed information of NoReC is presented in Table I.

TABLE I. Description of Norec

| **Emotion** | ***1*** | ***2*** | ***3*** | ***4*** | ***5*** | ***6*** |
|---|---|---|---|---|---|---|
| **Proportion** | 1.2% | 6.7% | 18% | 33% | 36% | 6.0% |
| **Documents** | ***Training*** | 20000 | | | | |
| | ***Validation*** | 1000 | | | | |
| | ***Testing*** | 1000 | | | | |
| | ***Rest*** | | | | | |
| | 35,189 | | | | | |

We apply metadata and homogeneously select 22000 documents in six kinds of reviews from the dataset to make the proportion in accordance with the original distribution because some reviews only have seldom words. Documents are divided into training, validation and testing datasets as 20000, 1000 and 1000 respectively. The raw dataset contains three types of Norwegian dialects, namely, Nynorsk, Bokmål and mixed. We uniform them by langid.py to Bokmål. According to [38], the conversion accuracy reaches 100%.

Each sentence is then parsed into fine-grained tokens. The statistics are depicted in Fig. 5. Abscissa record the number of tokens in a sentence and the ordinate count of such sentence in the whole dataset. The length of most reviews is congested within 100 which is set to be the threshold of feature scope preprocessed by truncation and padding in the input layer.

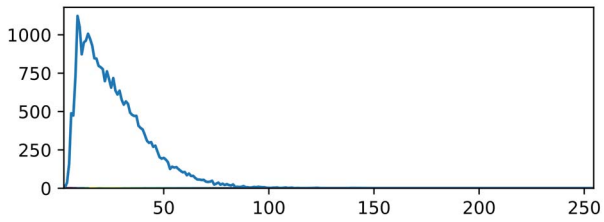


Fig. 5. Tokens Number in Training Dataset

## B. Metrics

We evaluate the effectiveness of embedding by confusion matrix (CM). The records in the data set are summarized in the form of a matrix according to two criteria from the real category and classification model.

$$CM = \begin{bmatrix} TP & FP \\ FN & TN \end{bmatrix} \tag{18}$$

Where each column represents the predicted emotion and each row records the real belonging of the data. $TP$ and $FP$ are the true and false proportions of prediction in positive emotion, $TN$ and $FN$ are true and false of negative prediction.

The Precision ($P$), recall ($R$), Accuracy ($Acc$) and $F1$ score of the model corresponding to each category can be calculated in equation as evaluation indicators (19-22).

$$P = \frac{TP}{TP+FP} \tag{19}$$

$$R = \frac{TP}{TP+FN} \tag{20}$$

$$Acc = \frac{TP+TN}{TP+FN+FP+FN} \tag{21}$$

$$F1 = \frac{2TP}{2TP+FP+FN} \tag{22}$$

Where $P$ is the probability of positive samples among all predicted positive samples. $R$ is the coverage of predicted positive samples in actual positive samples. $Acc$ is a percentage of correct predictions from total samples. As Accuracy and Recall always conflict in results, they are difficult to be simply used for model comparison. $F1$ is used as a harmonic average of the Accuracy and Recall.

## C. Baseline and parameters

Six types of embeddings are adopted as baselines. The essential comparison is held by BERT and ELMO. We also apply the ELMO-BERT Concatenation (EBC), which is deemed as the most general combination method.

Besides, we utilize several traditional embeddings, which are known as three primitive patterns: randomly initialized Word2Vec, CNN and GRU model. All models have the same fully-connection structure in the classification layer. The model parameters and feature shapes of EFTE are shown in Table II.

TABLE II. PARAMETERS IN EFTE

| Layer | Parameters | Shape |
|---|---|---|
| **BERT** | Frozen | None,100,768 |
| **$T_1$-$T_1$** | 769*100 (Parallel) | (None,1) *100 |
| **$IT_1$-$T_1$** | 1 | None, 768 |
| **ELMO** | Frozen | None, 1024 |
| **DR** | 0 | None, 768 |
| **Batch Normalization** | 1536 | None, 768 |
| **$IM_1$-$S_2$** | 1 | None, 768 |
| **FC** | 4614 | None, 6 |
| **Total** | 81514 | - |

## D. Results and discussion

Each sentence has six classes ranking the attitude from negative to positive (1 to 6). The results are analyzed with a 6*6 confusion matrix and each value from the matrix is used to

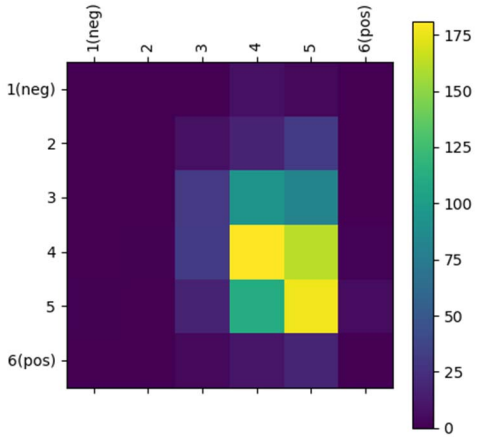


Fig. 6. Confusion Matrix of EFTE in Classification

TABLE III. COMPARISION BY EMBEDDINGS IN CLASSIFICATION

| Method | *Acc* | *R* | *P* | *F*1 |
|---|---|---|---|---|
| **Word2Vec** | 0.379 | **0.38** | 0.26 | 0.31 |
| **CNN** | 0.377 | **0.38** | 0.28 | 0.31 |
| **GRU** | 0.371 | 0.37 | 0.31 | 0.32 |
| **ELMO** | 0.268 | 0.27 | 0.31 | 0.28 |
| **BERT** | **0.384** | **0.38** | 0.32 | 0.31 |
| **EBC** | 0.355 | 0.35 | 0.29 | 0.31 |
| **EFTE** | 0.379 | **0.38** | **0.36** | **0.36** |

calculate the indicators by equations (19-22). The actual classification results are shown in Fig. 6 with 1000 testing example by EFTE using the above parameter configuration.

The color in confusion matrix represents the quantity of examples ranging from dark blue to bright yellow. It seems that our algorithm cannot well discriminate the emotion in Norwegian review. However, the intensity of emotion between two adjacent labels are obscure and their results are only slightly misled to the neighboring level. For example, level 3 are predicted as level 4, such error are tolerable and ineluctable for human sentimental expression [39-40].

In such a situation, the performance of our ensemble structure exceeds other six embeddings in general, illustrated in Table III. Notably, Recall, Precision and *F*1 scores surpass others, especially with 4% more than second best model GRU in *F*1 score and 4% overtaking BERT in Precision. In terms of Accuracy, the proposed EFTE embedding comes in the second place.

For the time-consumption as depicted in Fig. 7, EFTE becomes stable at the 25$^{th}$ epoch which is the fastest in convergence among the four kinds of embeddings. It is remarkable that, w.r.t. *F*1 score, our embedding apparently outperforms others. It is also shown that, after the 25$^{th}$ epoch, EFTE starts overfitting but still remains as the leader in *F*1 metrics.

In order to evaluate the inner weighting of tokens, we pass two sentences in radical polarity to the well-trained EFTE network and observe the output of self-attention in $T_1$ ▒ $T_1$ layer, which are shown in the heatmap in Fig. 8, where Fig. 8 (a) presents example for positive review in level 6, Fig. 8 (b) shows an instance of negative review in level 1. EFTE can automatically highlight their emotional encoding that "*toppen*" and "*unødvendig*" are selected in positive review and negative review respectively.

***Toppen*** *innen drama akkurat nå !*

***Top*** *in drama right now !*

(a) Positive review in attention

*Kroneksemplet på et* ***unødvendig kulturprodukt***

*Typical example on an* ***unnecessary cultural product***

(b) Negative review in attention

Fig. 8. Self-Attention in $T_1$-$T_1$ layer

After training, we freeze all the parameters and manually adjust the weights in two ensemble layers to observe the accuracy in NoReC classification. Table IV shows the accuracy in different sentence and word proportion. Table V shows the accuracy with different ELMO and BERT weights. When we tune one layer, the other layer keeps default training values. The proportion of each selection layer starts from 10% to 100%.

In Tables IV and V, **Acc** does not show a monotonic change, the best proportion is 90% in sentence and 10% in tokens. We also find that EFTE shows the best accuracy at the most percentage of BERT (70%-80%) compared to (20%-30%) in case of ELMO. In automatic training, weights are restrained at 87% of sentence and 13% of tokens embedding and constricted

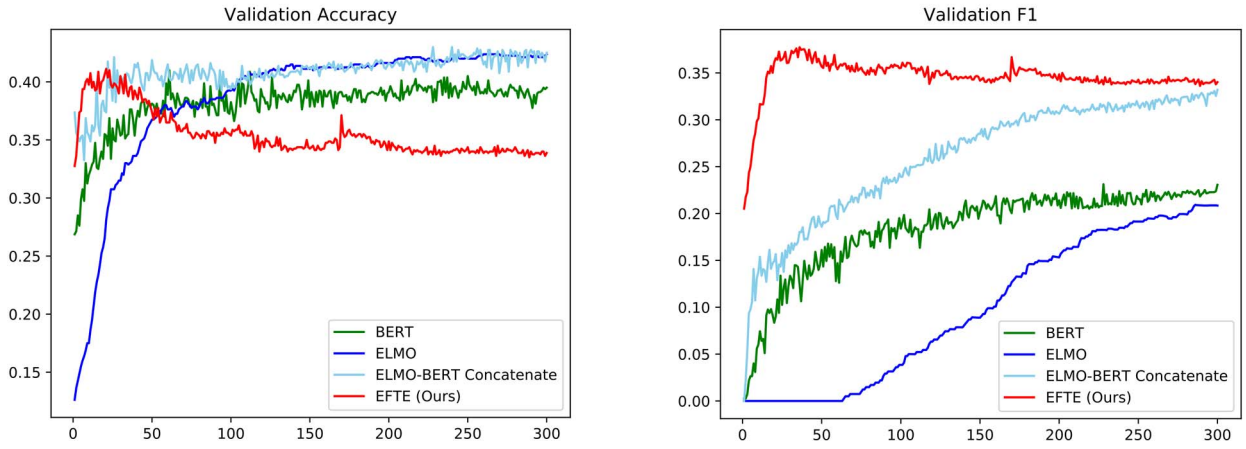


Fig. 7. Supervision in Validation by Accuracy and F1 Score

TABLE IV. INFLUENCE OF SENTENCE AND TOKEN EMBEDDINGS

| Sentence | Word | Acc |
|---|---|---|
| 10% | 90% | 0.351 |
| 20% | 80% | 0.356 |
| 30% | 70% | 0.356 |
| 40% | 60% | 0.36 |
| 50% | 50% | 0.361 |
| 60% | 40% | 0.36 |
| 70% | 30% | 0.363 |
| 80% | 20% | 0.367 |
| **Auto: 87%** | **Auto: 13%** | 0.37 |
| **90%** | **10%** | **0.371** |
| 100% | 0% | 0.366 |

TABLE V. INFLUENCE OF BERT AND ELMO EMBEDDINGS

| BERT | ELMO | Acc |
|---|---|---|
| 10% | 90% | 0.296 |
| 20% | 80% | 0.308 |
| 30% | 70% | 0.32 |
| 40% | 60% | 0.346 |
| 50% | 50% | 0.361 |
| **Auto: 57%** | **Auto: 43%** | 0.368 |
| 60% | 40% | 0.369 |
| **70%** | **30%** | **0.394** |
| **80%** | **20%** | **0.394** |
| 90% | 10% | 0.387 |
| 100% | 0% | 0.377 |

in 57% of BERT and 43% of ELMO. Parameters may include other unexplainable causes like the conflicts between different types of embeddings. With the results of ELMO-BERT ensemble in EFTE, the importance is prone to BERT which is 57% over 43% of ELMO.

## V. CONCLUSION

In this paper, we proposed a novel framework for Norwegian representation, EFTE, to enhance the results of classification. We applied BERT and ELMO and considered tokens and sentences level information for integration. Experimental results showed that our proposed algorithm outperforms six single embeddings in terms of $F1$ score. We also analyzed the main components of EFTE and showed how different embeddings are connected together.

We are planning to further explore other types of embeddings, such as the combination of LDA and BERT. In addition, we plan to collaborate with Norwegian linguists to extract Norwegian language feature in embeddings.

## REFERENCES


[1] M. Korpusik, Z. Liu, and J. Glass. "A Comparison of Deep Learning Methods for Language Understanding." Proc. Interspeech 2019 (2019): 849-853.

[2] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. "Attention is all you need." In Advances in neural information processing systems, pp. 5998-6008. 2017.

[3] N. Jiang, and M. Marneffe. "Evaluating BERT for natural language inference: A case study on the CommitmentBank." EMNLP-IJCNLP, pp. 6088-6093. 2019.

[4] J. Lee, and J. Hsiang. "Patent Claim Generation by Fine-Tuning OpenAI GPT-2." arXiv preprint arXiv:1907.02052 (2019).

[5] T. Klein, and M. Nabi. "Learning to Answer by Learning to Ask: Getting the Best of GPT-2 and BERT Worlds." arXiv preprint arXiv:1911.02365 (2019).

[6] K. Kowsari, K. J. Meimandi, M. Heidarysafa, S. Mendu, L. Barnes, and D. Brown. "Text classification algorithms: A survey." Information 10, no. 4 (2019): 150.

[7] K. Fan, H. Li, and X. Jiang. "An Improved Adaptive and Structured Sentence Embedding." In 2019 International Conference on Smart Grid and Electrical Automation (ICSGEA), pp. 199-203. IEEE, 2019.

[8] J. Devlin, M. Chang, K. Lee, and K. Toutanova. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).

[9] M. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. "Deep contextualized word representations." arXiv preprint arXiv:1802.05365 (2018).

[10] S. Wu, and M. Dredze. "Beto, Bentz, Becas: The Surprising Cross-Lingual Effectiveness of BERT." arXiv preprint arXiv:1904.09077 (2019).

[11] D. Zeman, J. Hajič, M. Popel, M. Potthast, M. Straka, F. Ginter, J. Nivre, and S. Petrov. "CoNLL 2018 shared task: multilingual parsing from raw text to universal dependencies." In Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies, pp. 1-21. 2018.

[12] Z. Zhu, J. Liang, D. Li, H. Yu, and G. Liu. "Hot Topic Detection Based on a Refined TF-IDF Algorithm." IEEE Access 7 (2019): 26996-27007.

[13] S. M. H. Dadgar, M. S. Araghi, and M. M. Farahani. "A novel text mining approach based on TF-IDF and Support Vector Machine for news classification." In 2016 IEEE International Conference on Engineering and Technology (ICETECH), pp. 112-116. IEEE, 2016.

[14] M. Pavlinek, and V. Podgorelec. "Text classification method based on self-training and LDA topic models." Expert Systems with Applications 80 (2017): 83-93.

[15] H. Amoualian, W. Lu, E. Gaussier, G. Balikas, M. R. Amini, and M. Clausel. "Topical coherence in lda-based models through induced segmentation." In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 1799-1809. 2017.

[16] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. "Distributed representations of words and phrases and their compositionality." In Advances in neural information processing systems, pp. 3111-3119. 2013.

[17] Q. Le, and T. Mikolov. "Distributed representations of sentences and documents." In International conference on machine learning, pp. 1188-1196. 2014.

[18] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov. "Bag of tricks for efficient text classification." arXiv preprint arXiv:1607.01759 (2016).

[19] Z. Zheng, L. Zheng, and Y. Yang. "A discriminatively learned CNN embedding for person reidentification." ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM) 14, no. 1 (2018): 13.

[20] P. Wang, Y. Qian, F. K. Soong, L. He, and H. Zhao. "Word embedding for recurrent neural network based TTS synthesis." In 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 4879-4883. IEEE, 2015.

[21] O. Melamud, J. Goldberger, and I. Dagan. "context2vec: Learning generic context embedding with bidirectional lstm." In Proceedings of the 20th SIGNLL conference on computational natural language learning, pp. 51-61. 2016.

[22] M. E. Peters, W. Ammar, C. Bhagavatula, and R. Power. "Semi-supervised sequence tagging with bidirectional language models." In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, pp. 1756–1765. 2017.

[23] C. Xiao, T. Ma, A. B. Dieng, D. M. Blei, and F. Wang. "Readmission prediction via deep contextual embedding of clinical concepts." PloS one 13, no. 4 (2018): e0195024.

[24] T. Bansal, D. Belanger, and A. McCallum. "Ask the gru: Multi-task learning for deep text recommendations." In Proceedings of the 10th ACM Conference on Recommender Systems, pp. 107-114. ACM, 2016.

[25] A. Zadeh, M. Chen, S. Poria, E. Cambria, and L. Morency. "Tensor fusion network for multimodal sentiment analysis." arXiv preprint arXiv:1707.07250 (2017).

[26] B. Dzmitry, K. Cho, and Y. Bengio. 2015. Neural machine translation by jointly learning to align and translate. In Proceedings of the Third International Conference on Learning Representations (ICLR).

[27] A. See, P. J. Liu, and C. D. Manning. "Get To The Point: Summarization with Pointer-Generator Networks" In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, pp. 1073–1083. 2017.

[28] F. Liu, and J. Perez. "Gated end-to-end memory networks." In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers, pp. 1-10. 2017.

[29] W. Wei, J. Liu, X. Mao, G. Guo, F. Zhu, P. Zhou, and Y. Hu. "Emotion-aware Chat Machine: Automatic Emotional Response Generation for Human-like Emotional Interaction." In Proceedings of the 28th ACM International Conference on Information and Knowledge Management, pp. 1401-1410. ACM, 2019.

[30] N. C. Mithun, J. Li, F. Metze, and A. K. Roy-Chowdhury. "Learning joint embedding with multimodal cues for cross-modal video-text retrieval." In Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval, pp. 19-27. ACM, 2018.

[31] L. Luo. "Network text sentiment analysis method combining LDA text representation and GRU-CNN." Personal and Ubiquitous Computing (2018): 1-8.

[32] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 770–778, 2016.

[33] A. Kumar, O. Irsoy, P. Ondruska, M. Iyyer, J. Bradbury, I. Gulrajani, V. Zhong, R. Paulus, and R. Socher. "Ask me anything: Dynamic memory networks for natural language processing." In International conference on machine learning, pp. 1378-1387. 2016.

[34] Z. Lin, M. Feng, C. N. dos Santos, M. Yu, B. Xiang, B. Zhou, and Y. Bengio. "A structured self-attentive sentence embedding." arXiv preprint arXiv:1703.03130 (2017).

[35] T. H. Chan, K. Jia, S. Gao, J. Lu, Z. Zeng, and Y. Ma. "Pcanet: A simple deep learning baseline for image classification?" IEEE transactions on image processing, 24(12):5017–5032, 2015

[36] S. Wang, L. Chen, Z. Zhou, X. Sun, and J. Dong. "Human fall detection in surveillance video based on PCANet." Multimedia tools and applications 75, no. 19 (2016): 11603-11613.

[37] Y. Li, X. Wu, and J. Kittler. "L1-2D 2 PCANet: a deep learning network for face recognition." Journal of Electronic Imaging 28, no. 2 (2019): 023016.

[38] Velldal, E., Øvrelid, L., Eivind Alexander Bergem, C. S., Touileb, S., and Jørgensen, F. (2018). "NoReC: The Norwegian Review Corpus." In Proceedings of the 11th edition of the Language Resources and Evaluation Conference, pages 4186–4191, Miyazaki, Japan.

[39] E. Mower, A. Metallinou, C. Lee, A. Kazemzadeh, C. Busso, S. Lee, and S. Narayanan. "Interpreting ambiguous emotional expressions." In 2009 3rd International Conference on Affective Computing and Intelligent Interaction and Workshops, pp. 1-8. IEEE, 2009.

[40] V. S. Ferreira, L. R. Slevc, and E. S. Rogers. "How do speakers avoid ambiguous linguistic expressions?." Cognition 96, no. 3 (2005): 263-284.