ETC^2: Near-Attention Ensemble of Term Classification for Effective and Efficient Text Classification

Anonymous ACL submission

Abstract

Sequence models, particularly those leveraging transformer architectures, have recently dominated the Automatic Text Classification (ATC) field. These models represent words as dense contextual vectors composing the document (dense) representations. Though effective, these models are expensive for training (fine-tuning) and inference (prediction) time. Traditional bag-of-words approaches that directly represent a document as a single sparse vector are usually much more efficient but are not as effective as sequence models. Both model types commonly involve constructing a representation of the entire document before predicting its class, overlooking the importance of some individual word (co-)occurrences for the target task. This paper takes a completely different approach to the ATC task by promoting words as "first-class" citizens for ATC. In other words, our method, called ETC^2, directly classifies each term of a document – using an intricate combination of (i) frequentist information, (ii) explicit co-occurrence and context modeling, and (iii) (near-)attention layering. It then uses these predictions to estimate the document class. The proposed approach eliminates the need for a single document representation, thus enormously improving model efficiency. In our experimental evaluation, ETC^2 was as effective as (if not better) than the best Transformer baselines in the tested datasets, being up to 17x faster at inference (prediction) time than modern Transformer-based classifiers.

1 Introduction

Automatic text classification (ATC) plays a pivotal role in information systems, providing a systematic means to assign topical categories to diverse text units such as documents, social media posts, and news articles. As the volume of textual data escalates, efficient and accurate text classification methods become increasingly important.

ATC approaches have predominantly employed bag-of-words and, more recently, sequence-based models, each with strengths and weaknesses. Sequence models, particularly those leveraging transformer architectures (Vaswani et al., 2017; Yang et al., 2019; Sanh et al., 2019; Liu et al., 2019; Radford et al., 2018), have emerged as a dominant force in ATC (Cunha et al., 2021; de Andrade et al., 2023). These models represent each word as a dense vector and a document as a combination of these representations, offering a nuanced understanding of the text's semantic structure. However, such a rich representation comes at the price of a high computational demand, which might worsen if long text sequences are considered, posing a significant hurdle to their widespread applicability, particularly in resource-constrained environments. On the other hand, the traditional bag-of-word approaches rely on simpler representations that often achieve superior efficiency by directly encoding a document as a single (sparse) vector at the expense of non-top-notch effectiveness in some scenarios.

This duality in model choices has prompted researchers to explore novel methodologies that balance representation complexity and computational efficiency. *We follow this path.* Indeed, our main goal in this paper is to explore a trade-off between representation complexity and algorithm efficiency while, at the same time, achieving the same or superior predictive capability.

One common characteristic of the aforementioned Bag-of-words- and Sequence-based approaches, which may be seen as a limitation, is that they have to construct a representation of the entire document before predicting its class. This document-centric approach overlooks the importance of individual term (co-)occurrences, potentially hindering the models' ability to capture subtle nuances within the text. Recognizing this gap in the current research landscape, our study introduces a new framework designed to address this limitation and enhance the ATC effectiveness.

In this context, we present a novel "word-centric"

approach to ATC that goes beyond conventional document-level classification. Instead of treating the entire document as a monolithic entity, our proposed methodology, named **Ensemble of Term Classification for Efficient Text Classification (ETC^2)**, focuses on directly classifying each term within a contextualized bag of words and subsequently estimating the document class. This paradigm shift eliminates the need for a single document representation, allowing for more granular text analysis and improving the model's ability to discern intricate details within the data.

In a broad sense, ETC^2 demonstrates proficiency in recognizing discriminating occurrences of term contexts that contribute to distinguishing labels. Conversely, to identify the most discriminative co-occurrences, a loss function is applied to mitigate the impact of easy-to-classify documents – a high probability of the expected label means less importance in the loss – and emphasize the uncertain co-occurrences, blurring the biased ones in the decision function. This strategy compels the algorithm to discern the specific co-occurrences of each term responsible for the discrimination in the final class decision.

In a nutshell, our novel approach, depicted in Figure 1, encompasses a few embedding layers[1], a single non-parametric multi-headed near-attention layer, and a linear layer to represent the model. Incorporating multiple embedding layers enables the model to capture intricate semantic features within the documents, facilitating a more nuanced understanding of the text. The non-parametric multi-headed attention mechanism also empowers the model to efficiently attend to relevant information across different document parts, effectively leveraging local and global classification contexts. Finally, the linear layer is a robust representation, consolidating the learned features into a better predictor. By leveraging a streamlined architecture comprising these key components, our method achieves higher predictive performance for unknown documents, especially for large datasets.

To guide our research and evaluation of the potential ETC^2 advantages in terms of efficiency and effectiveness when compared to strong baseline sequence-based methods (Transformers), we focus on answering the following research questions:

- R1: How does the ETC^2 framework perform compared to traditional fine-tuned sequence classification methods across diverse datasets?
- R2: How does incorporating the proposed single near-Attention layer impact the model's ability to capture nuanced term contexts compared to the traditional inner product layer?
- R3: How stopwords removal and near-Attention contribute to ETC^2's efficiency/effectiveness?

We conduct an extensive experimental evaluation of the proposed ETC^2 framework, considering seven reference datasets and four strong sequence-based transformer baselines. Our experimental results show that ETC^2 stands out as the most efficient solution while maintaining comparable or even better predictive capabilities when compared to the baselines. In more detail, ETC^2 achieves state-of-the-art effectiveness results in most tested datasets, achieving up to 17x speedup gains in ATC prediction efficiency. Importantly, this high efficiency is primarily attained without compromising (or improving) effectiveness in most datasets.

Our proposed approach challenges the conventional wisdom in ATC. Our main contributions include: (i) introduction of the **Ensemble of Term Classification (ETC^2)** framework, a very novel ATC approach, which changes the way classification is thought and performed when compared with the state-of-the-art; (ii) development of an efficient BoW-Based algorithm designed to classify lengthy documents; (iii) proposal of a novel representation for contextualized BoW, achieved by leveraging the weighted co-occurrences of n-grams through a single near-Attention layer; (iv) extensive experimentation of the proposed framework.

2 Related Work

Static word embeddings have been pivotal in NLP tasks, offering pre-trained representations that capture semantic relationships among words. Word2Vec (Mikolov et al., 2013), with Continuous Bag of Words (CBOW) and Skip-gram models, are prominent examples of early word embedding techniques. GloVe (Pennington et al., 2014), another widely adopted static word embedding model, constructs word vectors by leveraging global co-occurrence statistics from a corpus, effectively capturing syntactic and semantic word relationships. FastText (Joulin et al., 2016) extends

[1]A layer in the context of neural networks refers to a functional unit that processes input data through a series of mathematical operations to produce output representations.

traditional word embeddings by incorporating subword information. While these models differ in operation, they all produce static Bag-of-Words (BoW) embeddings without considering contextual dependencies within the document. This static representation limits their ability to capture nuanced semantic relationships crucial for ATC.

State-of-the-art ATC methods, based on sequences of word models, leverage advanced pre-trained language models to enhance effectiveness. BERT (Vaswani et al., 2017)´s ability to capture contextual information bidirectionally led to substantial improvements in various NLP tasks.DistilBERT (Sanh et al., 2019) further explores the efficiency-effectiveness trade-off by distilling the knowledge from BERT into a smaller, distilled version, with a significantly reduced parameter count. RoBERTa (Liu et al., 2019) builds upon BERT's architecture with modifications such as dynamic masking during pre-training. Based on a different architecture, XLNet (Yang et al., 2019) introduces a permutation language modeling objective, combining the strengths of autoregressive and autoencoding models.

Models that integrate term co-occurrence relations within documents effectively capture contextualized terms. Unlike conventional bag-of-words embedding techniques, which overlook term co-occurrence and thus fail to contextualize terms adequately, newer Transformer-based models are explicitly designed to evaluate both term occurrence and absolute position within the document, facilitated by mechanisms like attention layers. This enables them to infer masked or subsequent tokens in a sequence, leading to a richer understanding of contextual semantics. However, this enhanced capability comes at a computational cost, particularly noticeable when processing large documents due to the intensive computations required.

By contrast, ETC^2 presents a novel ATC approach. While existing models excel in capturing complex contextual dependencies at the corpus or document level, ETC^2 prioritizes a more granular classification of individual term contexts within documents. This distinct focus on the (word, class) relationship enables ETC^2 to discern nuanced textual nuances as effectively as, or even surpassing, sequence models while maintaining the computational efficiency of BoW representations.

3 ETC^2 Framework

Let $D = \{(d_i, l_i)\}_N$ be a set with N documents (d_i) associated with (training) labels (l_i). The problem can be generalized as predicting each label for all unseen documents in D': $\arg\max_{l \in L} \Pr(l|d_i)$. To the best of our knowledge, all current ATC algorithms implement the previous expression to determine the label that maximizes the prediction probability for the unknown document d_i, based on a single representation for d_i. For instance, in sequence-models, usually the <CSL>-contextualized token represents d_i. This representation usually has $d_i \in \mathbb{R}^c$, with c (a constant) being the dimensionality of representation space. This representation size (c) is fixed and can be small and dense as in sequence models or large and sparse (i.e., c = vocabulary size) as in *TFIDF*-models.

We exploit a very different approach for the ATC task. ETC^2 explores the manifold representations intrinsic to a document. Here, we represent each document as $d_i \in \mathbb{R}^{kc}$, with k being a constant ($k < |d_i|$ for most documents[2]) and c a hyperparameter denoting the number of hidden units in the proposed architecture. This approach involves encapsulating the document through contextualized terms, wherein the term importance discerns the document class label following its co-occurrence discriminating power.

ETC^2 represents the label/document posterior as the joint probabilities of term contexts (tc_j in d_i) and labels (l), as shown in Eq. 1.

$$\Pr(l|d_i) = \sum_{tc_j \in d_i} \Pr(l|tc_j) \Pr(tc_j|d_i) \quad (1)$$

Figure 1 presents a simplified diagram of the ETC^2 framework, delineating the representation of embeddings for terms (Q, K, and V) and specialized one-hot encoding for Term Frequency (TF) and Document Frequency (DF). ETC^2 constructs a contextualized term representation by considering the intricate co-occurrence relationships between terms within the document, named *co-occurrence probability* ($\Pr(to_1, to_2|d)$), which means the probability of both terms, to_1 and to_2, co-occurring together at document d. ETC^2 utilizes these probabilities to infer the posterior $\Pr(tc|d)$. Ultimately, the probability $\Pr(l|tc)$ and the final $\Pr(l|d_i)$ are inferred.

The term contexts operate in a shared space of terms, their frequencies (TF and DF), and co-occurrences, as described in Section 3.1. Next,

[2]We set k as the 90-percentile of training document sizes.

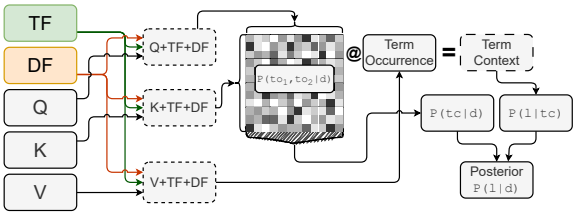


Figure 1: ETC^2 Overall Diagram.

in Section 3.2, we discuss how the label is inferred ($\Pr(l|tc_j)$) based on the importance of these terms' contexts ($\Pr(tc_j|d_i)$).

3.1 Term Occurrence and Term Context

ETC^2 builds the (partial) document representation by converting a text sequence into a set of terms t_j, formed by (i) unique uni- and bi-grams non-stopwords in the document d_i, (ii) their respective Term Frequency $TF_{(i,j)}$ within d_i, and (iii) the Document Frequency (DF_j) within the dataset. We embed the term occurrences by encoding the term, term frequency (TF), and document frequency (DF) values into specialized one-hot encodings.

While the term encoder uses the traditional one-hot-encoder, the TF and DF encoders consider the squared and logarithmic-scale rounded encoding, respectively. This results in terms with (scale-)comparable frequency distributions, whether within documents or across the corpus, being mapped to shared spaces. For instance, terms occurring 3, 4, or 5 times within the corpus are embedded equally, a.k.a., the same Document Frequency Bias, due to the encoding function.

$$\mathbf{Enc}(3) = \mathbf{Enc}(4) = \text{round}(\log_2(5)) = 2 \tag{2}$$

When considering scale as the encoding factor in DF, we establish a basic representation of term rarity akin to the Inverse Document Frequency (IDF), quantifying the proportion of term occurrences in documents. Similarly, our approach integrates the rarity scale of terms as a quantifiable factor.

Thus, all terms happening in just 1 document (DF=1) will have the same "bias" within documents (the parameters). The same applies to terms that occurred in 2 documents, which will share the global rarity bias. On the other hand, terms that occur between 370,728 documents and 741,455 documents will have the same bias (encoded to 19), and terms that happen in more than 741,455 documents will share the code 20 (the maximum supported occurrence).

Similarly, Term Frequency Bias embeds comparable frequencies, but based on the squared of $TF_{(j,i)}$, a.k.a. term frequency within the document. Table 1 shows an example of the TF and DF encodings.The sum of these embeddings (term, $TF_{(j,i)}$, and DF_j) captures the joint influence of terms on the view of term frequencies and rarity.

Code	0	1	2	3	5	10
$TF_{(j,i)}$	[1;2]	[3;6]	[7;12]	[13;20]	[31;42]	[91;110]
DF_j	[1]	[2]	[3;5]	[6;11]	[23;45]	[725;1448]

Table 1: Encoder to $TF_{(j,i)}$ and DF_j values.

The term's context tc_j represents the integration of the probability of its self-occurrence to_j and its co-occurrence with other terms within the document ($to_k \in d_i$). Applying the Euclidean distance this process aims to evaluate the likelihood of both terms being situated in the same location within the spaces of occurrences and, to integrate these probabilities, we introduce a novel layer termed the near-Attention Layer, as depicted in Eq. 3.

$$tc_j = \sum_{to_k \in d_i} \Pr(to_j, to_k|d_i) to_k \tag{3}$$

This Layer, aligned with the conventional approach to representing term context, as used in Transformers, utilizes separate representations for key and query term encoding (multi-headily). It entails applying two normalization procedures under the Q/K-terms Euclidean Distance, as in Eq. 4.

$$\Pr(to_j, to_k|d_i) = \frac{\sigma\Big(\text{Norm}(to_j, to_k)\Big)}{\sum_{to_d \in d_i} \sigma\Big(\text{Norm}(to_j, to_d)\Big)} \tag{4}$$

$$\text{Norm}(to_j, to_k) = \frac{ndist(K_{to_j}, Q_{to_k}) - \mathbb{E}[ndist]}{\sqrt{\mathbb{V}\text{ar}[ndist] + \epsilon}}$$

$$ndist(K, Q) = \frac{b + \epsilon}{||K - Q||^2 + b + \epsilon} \tag{5}$$

where σ denotes the sigmoid function, $\mathbb{E}$ and $\mathbb{V}$ar represent the expected value and variance of the normalized Euclidean distance ($ndist$), respectively; b is the bias of the expected distance; and $||K_{to_j} - Q_{to_k}||^2$ the Euclidean distance between the occurrence representations of term j (key) and term k (query) within the document.

By replacing the conventional inner product typically used in most Transformers, we construct spaces with enhanced granularity, eliminating the need for multiple layers. Another key feature of this Layer includes considering the term's attention within the document, normalized by their distance to the average (layer norm), thereby significantly extending the attention range.

This approach allows the model to discern and quantify co-occurrences with higher discriminative power than others. However, to facilitate this, the model must prioritize attention towards terms co-occurrences within less easy-to-classify documents within the decision function. Hence, for

the ETC^2 framework to operate effectively, it must complement the co-occurrences with a focused loss function. In the next Section, we will detail how Focal Loss works and how to incorporate the near-Attention Layer into terms' context weight to infer the label posterior probability. As we shall see in our Ablation analysis (Section 4.4.1) the significance of employing near-attention in contrast to the conventional inner product is notorious.

3.2 Ensembling the Terms's Classifications

In ETC^2, the combination of weights attributed to co-occurrences of terms plays a crucial role in determining the final classification outcome. This Section explores the methodology of integrating these weights and the associated challenges in mitigating the influence of well-classified examples through focal Loss.

ETC^2 integrates two essential equations to predict term contexts and labels within a document. The probability of label l given document d_i is computed as the summation over all term contexts tc_j present in d_i, where each term context's likelihood is determined by the product of $\Pr(l|tc_j)$ and $\Pr(tc_j|d_i)$, see Eq. 1. For each term context tc_j, the probability of label l given the term context ($\Pr(l|tc_j)$) is obtained through a linear transformation followed by a softmax over the label set L:

$$\begin{aligned} \Pr(l|tc_j) &= \text{Softmax}_L(\text{Linear}(tc_j)) \\ \text{Linear}(X) &= W \times X + b \end{aligned} \tag{6}$$

where $W \in \mathbb{R}^{|L| \times c}$ and $b \in \mathbb{R}^c$ are learnable parameters. The softmax function, a standard component in ATC tasks, transforms the raw scores generated by the linear Layer into a probability distribution. The linear Layer is a critical component in the classification process, reflecting the likelihood of the raw term context X into meaningful class probabilities. This approach facilitates accurate and informed classification decisions.

Simultaneously, the probability of each term context tc_j given document d_i ($\Pr(tc_j|d_i)$) is calculated based on the conditional probabilities of term co-occurrences. This probability is determined by the ratio of the sum of the conditional probabilities of tc_j given all other term contexts tc_k within d_i to the sum of all possible combinations of term contexts $tc_{k'}$ and $tc_{k''}$ within d_i:

$$\Pr(tc_j|d_i) = \frac{\sum_k \Pr(tc_j|tc_k, d_i)}{\sum_{k',k''} \Pr(tc_{k'}|tc_{k''}, d_i)} \tag{7}$$

By combining these equations, ETC^2´s model dynamically evaluates the importance of each term context within the document and predicts the labels accordingly, thus enabling nuanced and contextually informed document classification.

One last challenge in leveraging co-occurrence weights is obscuring the influence of easy-to-classify documents during the classification process. Focal loss (Lin et al., 2017), a strategy commonly employed to address class imbalance and prioritize learning from hard examples, can inadvertently reinforce easy-to-classify documents' influence, thereby, their terms' co-occurrences. This Loss can lead to suboptimal classification outcomes by prioritizing class-exclusive (but non-discriminative) terms over more subtle contextual cues.

$$\text{FL}(\Pr(l|d_i)) = -(1 - \Pr(l|d_i))^\gamma \log(\Pr(l|d_i)) \tag{8}$$

In traditional cross-entropy Loss ($\gamma = 0$), well-classified examples often dominate the loss calculation, which can overshadow the learning process for minority classes or challenging instances. Focal Loss introduces a dynamic scaling factor, termed the focal parameter ($\gamma > 0$), which modulates the contribution of each example to the Loss based on its classification difficulty. As depicted in (Lin et al., 2017), Figure 2 illustrates an example of the corresponding losses for various gamma values, ranging from 0 to 5.

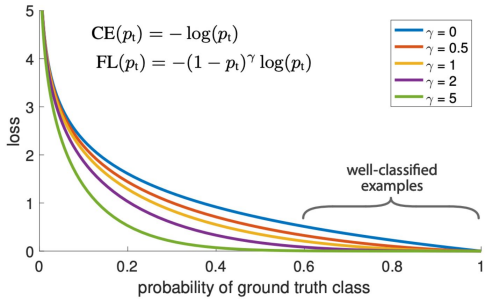


Figure 2: Focal Loss vs. Cross-Entropy Loss example.

ETC^2 stands out for its efficiency and simplicity, characterized by a minimalistic parameter footprint. With only embedding layers for Term (TF) and Document Frequency Bias (DF), Term Queries (Q), Keys (K), and Values (V), alongside parameters W and b, ETC^2 embodies a streamlined architecture that optimizes computational resources while maintaining robust performance.

The model captures the essential semantic co-occurrence information for effective term classification by leveraging embeddings for Q, K, and V. The judicious use of parameters W and b further enhances the model's expressiveness, enabling it to adapt to diverse text classification tasks while minimizing computational overhead. Consequently, ETC^2 presents itself as a lean yet potent ATC solution, offering a good balance among

Dataset	$\|L\|$	Balance	#Vocab	#Docs	DocLen	#T/Doc
sogou	5	**100.0**↑	**273363**↑	**510000**↑	**535.23**↑	**175.39**↑
20ng	20	94.32	176493	18906	266.96	139.84
wos11967	**33**↑	80.77	67978	**11967**↓	201.99	120.97
books	8	85.11	157526	33594	276.27	112.56
dblp	10	39.12	68127	38128	146.17	86.741
acm	11	**34.49**↓	**55761**↓	24897	64.105	40.719
agnews	**4**↓	**100.0**↑	90137	127600	**39.646**↓	**33.834**↓

Table 2: Statistics – **Balance** (%): class ratio in percentage for each dataset; **DocLen**: average document length in each dataset;**#T/Doc**: average number of unique tokens per document in each dataset. Arrows represent the column's highest and smallest values.

simplicity in its parameterization, effectiveness, and efficiency, as our experiments shall confirm.

4 Experimental Design and Results

We conducted experiments to assess ETC^2 effectiveness/efficiency for ATC. The evaluation encompassed seven well-established and diverse ATC benchmarks, including five small-medium datasets (with less than 100k documents), namely, 20 newsgroups (20ng), ACM, books, dblp, and Web of Science (wos11967), and two large datasets–SOGOU and AGNews (with more than 120k documents). Details for each dataset, including document size, class distribution, vocabulary size, and other characteristics, are outlined in Table 2.[3]

As baselines, we compared ETC^2 against the most traditional fine-tuned sequence modeling methods (Transformers) widely considered state-of-the-art in ATC, namely, BERT (Vaswani et al., 2017), RoBERTa (Liu et al., 2019), XLNet (Yang et al., 2019), and DistillBert (Sanh et al., 2019), with implementations from (Wolf et al., 2020). To evaluate the performance of each method, we employed the F1 micro and macro metrics, which summarize model effectiveness in terms of overall accuracy (micro) and F1 per class (macro), accounting for dataset skewness. As an efficiency metric, we consider prediction speed at test time–the number of predicted documents per second–, which considers the necessary computational resources to apply the learned model to unseen documents.

Although model training time is crucial for initial development, a model's true worth is demonstrated through its predictive prowess during deployment. These models must efficiently process incoming data to facilitate timely insights and actions across diverse applications. Since the number of labeled documents typically remains smaller than the total document count, focusing on prediction time underscores the necessity for responsive and scalable models.

The experimental procedure was conducted on twin machines featuring Intel Xeon E5-2686 v4 processors with eight virtual CPUs and 62GiB of RAM. Additionally, each machine was equipped with one NVIDIA Tesla V100 GPU boasting 16GiB of video memory.

Lastly, to ensure statistical soundness, we consider a setup based on a 10-fold cross-validation procedure for small/medium datasets and a 5-fold cross-validation for large datasets. Results correspond to the average of the test folds in each scenario. For assessing statistical significance, we employed a t-student test to compare the proposed ETC² method with the baseline methods, considering a 99% confidence level.

4.1 Effectiveness Results

Micro and MacroF1 results presented in Table 3 show that ETC^2 excels in both effectiveness metrics, being one of the overall best and most consistent methods across all datasets. ETC^2 is the best single method (single winner) in three out of seven datasets (20ng, wos11967, Sogou), in terms of both Micro and MacroF1, being tied with all transformers in first place in the ACM dataset in terms of MicroF1. Overall, considering seven datasets and two metrics, ETC^2 has twenty-four wins and ten ties out of the 56 baseline comparative results (2 metrics $\times$ 4 baselines $\times$ 7 datasets). In other words, 60% of the time, ETC^2 is better or equal to some Transformer. Comparatively, BERT–the best Transformer–is the best single method in only two datasets. Same for Roberta. None of them are as consistent as ETC^2.

Indeed, even when (statistically) failing to win over other methods in some datasets, ETC^2 effectiveness is still very competitive with the best Transformers, losing by very small margins, even without the expensive step of exploiting external pre-training data. For instance, when considering MicF1, ETC^2's losses against the best baselines achieve no more than 2.0% in agnews (against Roberta), 1.7% in dblp (also against Roberta) and 2.8% in books (against Bert). Furthermore, in dblp, ETC^2 statistically ties with Distillbert and in books with Roberta and XLNet. Overall, ETC^2's effectiveness is very similar, if not better, than that of most Transformers in most tested datasets.

The excellent ETC^2 results, especially in the

[3]Class ratio indicates class skewness within the dataset ($AVG\_Len\_Classes/MAX\_Len\_Classes$).

	F1 - Micro					F1 - Macro				
Dataset	**ETC**	**BERT**	**DistilBert**	**RoBERTa**	**XLNet**	**ETC**	**BERT**	**DistilBert**	**RoBERTa**	**XLNet**
sogou	**97.1**$^{(0.1)}$	95.5$^{(0.2)}$▽	95.5$^{(0.1)}$▽	95.6$^{(0.1)}$▽	95.5$^{(0.1)}$▽	**97.1**$^{(0.1)}$	95.5$^{(0.2)}$▽	95.5$^{(0.1)}$▽	95.6$^{(0.0)}$▽	95.5$^{(0.1)}$▽
20ng	**91.0**$^{(0.7)}$	88.0$^{(0.9)}$▽	87.3$^{(0.7)}$▽	87.1$^{(1.0)}$▽	87.7$^{(0.7)}$▽	**90.7**$^{(0.7)}$	87.5$^{(0.9)}$▽	86.9$^{(0.7)}$▽	86.5$^{(1.0)}$▽	87.2$^{(0.7)}$▽
wos11967	**90.0**$^{(0.9)}$	87.4$^{(1.0)}$▽	87.2$^{(1.1)}$▽	87.5$^{(1.0)}$▽	87.2$^{(0.9)}$▽	**89.6**$^{(0.9)}$	87.0$^{(1.0)}$▽	86.8$^{(1.1)}$▽	87.1$^{(1.0)}$▽	86.9$^{(0.9)}$▽
acm	**79.3**$^{(0.8)}$	**79.9**$^{(0.8)}$●	**79.4**$^{(0.8)}$●	**78.8**$^{(0.8)}$●	**79.3**$^{(0.8)}$●	68.1$^{(1.7)}$	**71.1**$^{(1.2)}$△	70.2$^{(1.2)}$△	70.2$^{(1.8)}$●	70.5$^{(2.1)}$●
books	86.6$^{(0.7)}$	**89.1**$^{(0.6)}$△	**88.0**$^{(0.5)}$△	87.2$^{(0.6)}$●	87.3$^{(0.7)}$●	86.7$^{(0.6)}$	**89.1**$^{(0.5)}$△	88.0$^{(0.5)}$△	87.2$^{(0.6)}$●	87.4$^{(0.8)}$●
dblp	82.4$^{(0.8)}$	83.5$^{(0.6)}$△	83.2$^{(0.6)}$●	**83.8**$^{(0.8)}$△	**83.7**$^{(0.7)}$△	80.6$^{(1.0)}$	81.9$^{(0.8)}$△	81.4$^{(0.8)}$●	**82.3**$^{(0.81)}$△	82.1$^{(0.8)}$△
agnews	92.5$^{(0.2)}$	94.2$^{(0.1)}$△	94.1$^{(0.1)}$△	**94.4**$^{(0.2)}$△	94.1$^{(0.3)}$△	92.5$^{(0.2)}$	94.2$^{(0.1)}$△	94.1$^{(0.1)}$△	**94.4**$^{(0.2)}$△	94.1$^{(0.3)}$△

Table 3: Average Micro/Macro F1 results (and standard deviations) obtained from the experiments. The best results, including statistical ties for each dataset are highlighted in bold. Symbols ▽, △and ●indicate results that are significantly lower, higher than or tied with the ETC² results, respectively (with a p-value < 0.01).

Speed$_{(docs/s)}$	**ETC**	**BERT**	**DistilBert**	**RoBERTa**	**XLNet**
20ng	**875.87**$^{(31.6)}$	175.69$^{(2.97)}$▽	286.81$^{(7.12)}$▽	171.21$^{(3.4)}$▽	78.564$^{(0.623)}$▽
wos11967	**1129.0**$^{(7.75)}$	186.78$^{(0.521)}$▽	314.35$^{(2.2)}$▽	187.15$^{(1.12)}$▽	80.995$^{(1.02)}$▽
acm	**2493.9**$^{(47.0)}$	210.33$^{(0.922)}$▽	386.99$^{(1.41)}$▽	216.84$^{(1.14)}$▽	86.951$^{(0.206)}$▽
books	**1282.7**$^{(15.8)}$	177.15$^{(3.0)}$▽	293.52$^{(0.72)}$▽	176.06$^{(0.699)}$▽	78.692$^{(0.218)}$▽
dblp	**1414.8**$^{(14.2)}$	196.95$^{(1.56)}$▽	344.03$^{(1.61)}$▽	198.51$^{(1.94)}$▽	84.034$^{(0.176)}$▽
agnews	**3672.4**$^{(22.0)}$	215.96$^{(1.38)}$▽	403.61$^{(1.47)}$▽	221.82$^{(1.5)}$▽	87.231$^{(0.928)}$▽
sogou	94.268$^{(0.109)}$	122.51$^{(0.0)}$△	**174.64**$^{(6.81)}$△	111.47$^{(3.26)}$△	68.334$^{(0.0336)}$▽

Table 4: Prediction/Inference Time (Speed) Comparative Results.

datasets in which it is the sole winner in both Micro and MacroF1 - Sogou, 20ng, and wos11967–may be explained by the large vocabulary and high density (#terms/doc) of these datasets. We can see in Table 2 that the first two datasets have the largest vocabularies, while the three datasets have the highest density among all. Large vocabularies and higher densities greatly benefit ETC^2 by enabling it to capture a wider range of linguistic nuances and semantic intricacies in textual data. With an expanded vocabulary, ETC^2 can represent a broader array of terms, improving its ability to discern subtle contextual cues and relationships within documents. Consequently, ETC^2 achieves superior performance in ATC, where understanding language syntactic co-occurrences is crucial for accurate predictions and insights.

4.2 Prediction/Inference Time (Speed)

Table 4 shows the ETC^2's prediction (inference) time against those of the Transformers regarding the number of documents classified per second. The higher the value, the faster the method. As highlighted, we consider prediction time even more important than training time, which can be done in batch and is run only once. Prediction at test time, on the other hand, involves the practical application of the method and is supposed to occur an unlimited number of times.

In this scenario, the superiority of ETC^2 is glaring. It is the sole winner in 6 out of 7 datasets, losing only in Sogou by a small margin. The speedup gains over some baselines achieve up to 42x, such as against XLNet, the slowest method. Compared to BERT, the most effective method, and the fastest Transformer, the speedup gains at prediction time ranges from 5x (in 20ng) to 17x (in agnews).

The remarkable efficiency in prediction speed can be attributed to its simple architecture and document representation approach. In essence, the model's design and method of representing documents are straightforward and streamlined, allowing for faster processing and inference during prediction tasks. This simplicity reduces computational overhead and enables the model to make predictions swiftly and efficiently.

In sum, when combining the effectiveness results with the efficiency ones, ETC^2 is the method of choice if the ATC task requires a fast and effective classifier that delivers top-notch effectiveness comparable to the best transformer architectures at speed only similar to the simplest and fastest of the Bow-based classifiers.

4.3 Visualization of ETC^2 Inner Workings

Figure 3 illustrates the reduced dimensional representation of terms contexts occurring in four documents from the first 20ng class, aimed at showcasing ETC^2's ability to discern groups of significant terms. The document selection process adhered to specific criteria: the correct prediction with the highest and least expected probability and the wrong prediction with the highest and least expected probability, all drawn from the test examples. Despite a few misclassifications, the analysis reveals that most impactful terms remain identifiable, underscoring ETC^2's capacity to discern terms with substantial discrimination power.

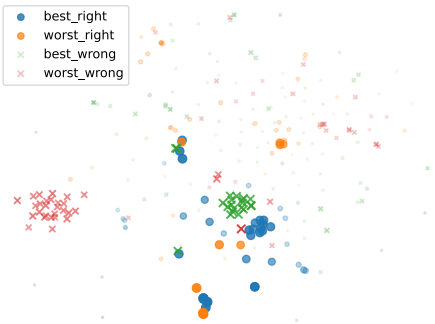
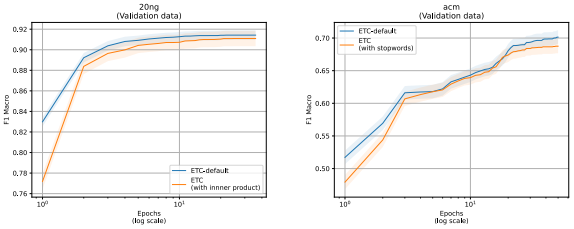


Figure 3: 2D representation of the terms' contexts. Point size means expected class probability $\Pr(l|tc_j)\Pr(tc_j|d_i)$.



((a)) 20ng Ablation. ((b)) ACM Ablation.

Figure 4: Ablation Analyzes.

4.4 Ablation Analysis

To assess the impact of each component within the ETC^2 framework, we run a series of experiments using 20ng and the ACM datasets, in which we either outperformed or achieved comparable results with all baseline methods. We employ the validation data in each step of the folded cross-validation process for this assessment, using MacroF1 as the evaluation metric. The reported values represent the highest F1 scores achieved up to the current point (epoch) over the training process, thereby enabling a comprehensive assessment of the effectiveness of each strategy. Confidence intervals were computed with a 95% confidence level for the average values obtained across the validation folds. Figure 4 encapsulates the outcomes derived from the datasets and some strategic approaches employed in the study.

4.4.1 Inner Product vs Distance-Based

Figure 4(a) presents a visual representation of the comparative analysis between the proposed near-attention layer and the traditional inner product method. The graphical depiction accentuates the substantial influence of the near-attention layer, especially notable in the early stages of the training process. As training progresses, the near-attention layer consistently outperforms the traditional inner product approach, maintaining its superior performance over the entire training process. Despite eventual convergence in performance between the two methods at the end of the training process, statistical analysis confirms the continued superiority of the near-attention layer in facilitating enhanced model performance. Furthermore, this analysis reveals the (future) possibility of an early stop in training due to a strong start and rapid convergence of ETC^2, which can help diminish training costs.

4.4.2 Stopwords Removal Analysis

Figure 4(b) shows the impact of stopwords removal in the ACM training. ETC^2 strongly relies on the terms' discriminating capability; thus, maintaining only terms in the model that can help the decision process is crucial for effectiveness and efficiency. The figure shows a notable impact on the model's final effectiveness. Particularly, the absence of stopwords significantly influences the ETC^2's outcomes throughout the training process, especially at the initial and final training stages.

5 Conclusion and Future Work

We proposed ETC^2, a novel ATC model that completely rethinks the task by promoting documents' terms as first-class citizens in the decision process and taking their collective opinion to make a final (class) prediction. Other innovations of our solution include exploring a frequentist approach, explicit co-occurrence and context modeling, near-attention layers, dynamic dropouts, and focal loss. All these innovations together were essential for allowing ETC^2 to achieve comparable (or superior) effectiveness compared to the best Transformers while preserving the efficiency of Bow-based approaches, as demonstrated in our comprehensible experimentation. Future research will explore pre-training methodologies tailored specifically for ETC^2, leveraging its efficiency. Indeed, our gains in efficiency motivate assessing the scalability and robustness of ETC^2 in handling new extensive textual corpora. Finally, the good performance on denser documents suggests that semantic document expansion, such as in (Viegas et al., 2019), may produce good results.

Limitations

Despite relevant contributions, our proposed ETC^2 framework has some limitations. One is apparent in datasets with lower document densities. This observation stems from ETC^2's requirement to construct distinct class contexts, which becomes challenging when documents are located in denser regions that are difficult to tell apart due to the lack of information (low density). Conversely, in datasets with high densities, exemplified by Sogou, the speed performance of the tokenizer is inferior compared to traditional methods. This disparity arises primarily from the tokenizers used in most sequence-based models, which limit the number of tokens processed per document. Our proposal has also room for improvement in ATC tasks that are highly ambiguous (i.e., with terms with high ambiguity), such as sentiment analysis or spam detection.

References

Washington Cunha, Vítor Mangaravite, Christian Gomes, Sérgio Canuto, Elaine Resende, Cecilia Nascimento, Felipe Viegas, Celso França, Wellington Santos Martins, Jussara M Almeida, et al. 2021. On the cost-effectiveness of neural and non-neural approaches and representations for text classification: A comprehensive comparative study. *Information Processing & Management*, 58(3):102481.

Cláudio Moisés Valiense de Andrade, Fabiano Belém, Washington Cunha, Celso França, Felipe Viegas, Leonardo Rocha, and Marcos André Gonçalves. 2023. On the class separability of contextual embeddings representations - or "the classifier does not matter when the (text) representation is so good!". *Inf. Process. Manag.*, 60(4):103336.

Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hérve Jégou, and Tomas Mikolov. 2016. Fasttext.zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651*.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. 2017. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. 2018. Improving language understanding by generative pre-training.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Felipe Viegas, Sérgio D. Canuto, Christian Gomes, Washington Luiz, Thierson Rosa, Sabir Ribas, Leonardo Rocha, and Marcos André Gonçalves. 2019. Cluwords: Exploiting semantic word clustering representation for enhanced topic modeling. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, WSDM 2019, Melbourne, VIC, Australia, February 11-15, 2019*, pages 753–761. ACM.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32.