information

MDPI

Article

An Ensemble Framework for Text Classification

Eleni Kamateri * and Michail Salampasis

Department of Information and Electronic Engineering, International Hellenic University, Alexander Campus, P.O. Box 141, Sindos, 57400 Thessaloniki, Greece; msa@ihu.gr
* Correspondence: ekamateri@iee.ihu.gr

Abstract: Ensemble learning can improve predictive performance compared to the performance of any of its constituents alone, while keeping computational demands manageable. However, no reference methodology is available for developing ensemble systems. In this paper, we adapt an ensemble framework for patent classification to assist data scientists in creating flexible ensemble architectures for text classification by selecting a finite set of constituent base models from the many available alternatives. We analyze the axes along which someone can select base models of an ensemble system and propose a methodology for combining them. Moreover, we conduct experiments to compare the effectiveness of ensemble systems against base models and state-of-the-art methods on multiple datasets (three patent classification and two text classification datasets), including long and short texts and single- and/or multi-labeled texts. The results verify the generality of our framework and the effectiveness of ensemble systems, especially ensembles of classifiers trained on different data sections/metadata.

Keywords: ensemble learning; ensemble framework; text classification; patent classification

1. Introduction

The task of text classification varies from domain to domain, depending on the type of text that must be classified. Different types of text differ in terms of various features, such as length, structure, metadata, kinds of words/phrases, grammatical patterns, syntactic idiosyncrasies, writing style, language/word distribution, etc., and thus different techniques for their classification are possible.

One example is patent classification, a text classification task dealing with long patent text that is full of technical and legal terminologies and structured in a distinctive way. Similarly to other domains, current research efforts in patent classification mainly focus on the use of machine learning (ML) models, deep learning (DL) models [1–4], and, lately, large language models (LLMs) [5,6] to address the automated assignment of classification codes to patent text. A promising technique that can improve the performance of traditional learning models is the use of ensembles of models that combine the knowledge gained from multiple learning models [7]. Recently, ensembles of classifiers have been applied in patent classification, bringing significant improvements in terms of accuracy compared to their constituent classifiers [8,9].

Even though the application and features of patent classification differ from other text classification tasks, the underlying problem remains the same. Hence, ensembles of base classifiers similar to those applied in patent classification could potentially attain better performance in any text classification task. Indeed, various combinations of classifiers have been proposed in the literature to address text classification [10]. However, a roadmap methodology for designing ensemble systems does not yet exist. This is an important

Academic Editors: Shadi Banitaan and Mina Maleki

Received: 18 October 2024
Revised: 23 December 2024
Accepted: 6 January 2025
Published: 23 January 2025

Citation: Kamateri, E.; Salampasis, M. An Ensemble Framework for Text Classification. *Information* **2025**, *16*, 85. https://doi.org/10.3390/info16020085

problem, as ensemble systems can be developed in myriad ways, but an ensemble must have only a small, finite set of constituent base models. Therefore, a data scientist must make a difficult decision, taking into account domain-specific language, data-centric parameters, the techniques that are applicable, or even practical demands such as periodicity when updating base models.

To assist this process, an Ensemble Patent Classification Framework (EPCF) has been proposed to guide the creation of ensemble architectures for automated patent classification [11]. Its objective is to guide patent researchers in designing new ensemble patent classification systems or transforming existing patent classification systems into powerful ensemble systems. Although it has been devised for patent classification, the framework can be transferred to any text classification task.

The main contributions of this work are as follows:

1. To define a new general, domain-agnostic ensemble framework for text classification;
2. To validate the applicability of the framework;
3. To evaluate the effectiveness of combining classifiers' knowledge against base classifiers and state-of-the-art (SotA) methods on several text classification datasets, and especially the effectiveness of combining knowledge from different data sections.

The remainder of the paper is structured as follows: Section 2 describes the related work in ensemble methodologies and frameworks. Section 3 presents the ensemble framework for text classification. Section 4 introduces the design approaches followed for developing representative ensemble systems that will be evaluated within this paper. Section 5 describes the data collection methods used for the evaluation. Sections 6 and 7 focus on the evaluation of the proposed framework, consisting of both the experimental methodology and evaluation results. Finally, Section 8 discusses the experimental results and Section 9 concludes the paper.

2. Related Work

Text classification, also known as text tagging or text categorization, is a task in which one or many pre-defined classification codes, known as tags or categories, are assigned to a given text based on its content, denoting the topic, the sentiment polarity, etc. The underlying text can vary significantly, from a few words or phrases (e.g., sentiment analysis) to long documents (e.g., document classification).

Earlier works in text classification organize the task into two steps: (1) extracting features and (2) feeding extracted features into a classifier to obtain the final label. The classifier could be an ML or a DL algorithm. Over the last decade, DL models have surpassed classical ML models in a variety of text classification tasks, and the main focus has shifted to developing the most appropriate DL classifier [12]. Recently, LLMs have changed the classification paradigm and demonstrated improved performance [13].

Meanwhile, ensemble learning has been explored as a new approach to boosting the performance of DL models in general and text classification specifically [14,15]. Fundamentally speaking, an ensemble learning model can be applied in any domain/task where high-bias and high-variance base predictive models exist to combine them into a better-performing, more stable model. Ensembles of heterogeneous classifiers, which differ in terms of the type of classifier, have been used to improve classification performance in the assignment of categories to long documents [16,17] and shorter texts such as ticket text [18]. Ensembles of homogeneous classifiers, which use the same classifier but differ in the way they manipulate the data, have also been used for text classification, using either different partitions of a dataset [19] or different subspaces of the feature space [20] and then submitting them to a classifier. Hybrid approaches using both heterogeneous and homogeneous classifiers can also be found in the literature. Different classifiers leverage

complementary information and different feature representation methods to predict the topic of documents [10] or leverage different feature sets for sentiment classification [21].

Even though numerous research initiatives exist related to ensembles of classifiers for text classification, there are no or very few available studies explaining the options for creating an ensemble system. A simple representation of an ensemble framework was presented in [22]. Although the framework does not include a design methodology, the authors describe two basic ensemble composition approaches: (i) to differ in the training dataset, i.e., a homogeneous approach, or (ii) to differ in the type of the base classifier, i.e., a heterogeneous approach. They also provide guidelines for the creation of different training datasets, introducing the concept of the data partitions. Data partitioning is based on the multi-view characteristic of the data that have a natural separation of their features or can be described using different views of information [23]. Ensemble learning can exploit these multiple partitions or views of the data and perform better than base learners [24,25]. Last, a significant study reviewing the key factors of any ensemble framework, especially ensemble deep learning, is presented in [15]. These factors include the following: (i) the data sampling method, (ii) the diversity of base classifiers, whether they are sequential and each of them has an impact on the formation of the next classifier or they are parallel, (iii) the fusion method, and (iv) the heterogeneity of the involved base classifiers, whether homogeneous or heterogeneous. Although these characteristics are adequately described, they are not presented on a simplistic and illustrative basis to make it easy to put them together.

3. An Ensemble Framework for Text Classification

Following the design principles introduced by the EPCF, we propose an ensemble framework for text classification that is structured according to the following three dimensions.

Dimension 1: Fundamental Ensemble Components

This dimension consists of the two core components that are necessary to create an ensemble system of base classifiers, including the following:

- The *training dataset* is the training samples providing labeled ground truth pairs of inputs and expected outputs.
- The *base classifiers* are the learning models trained on the given training dataset.

Dimension 2: Modes of Heterogeneity

Following the general principle of ensemble learning, the base classifiers involved in an ensemble system should be usefully diverse and have independent, hence complementary, predicting capabilities [26–28]. This diversity is implemented in the second dimension, which guides researchers in designing different base classifiers and includes diversity in each fundamental ensemble component.

Dimension 2.a: Different training datasets—Data Heterogeneity

The training datasets employed by base classifiers can differ in many aspects, called views or partitions:

- The *horizontal data partitioning* creates different training datasets by resampling the entire training dataset into dataset partitions (also called bootstraps). By repeating the copying of random samples, representative new datasets are formed that resemble the population of the initial dataset. This is a statistical method for deriving robust estimates of population parameters like mean.
- The *vertical data partitioning* creates different training datasets by using various features of the original data, such as other modalities, e.g., the text and the accompanied graphics; different sections or metadata, e.g., the title's and the main body's text; or other data representations, e.g., TF-IDF and word embeddings.

- The *functional data partitioning* forms different training datasets using any function that splits the original data into subparts, e.g., different data partitions split the data based on the publication date or the labels.

Dimension 2.b: Different base classifiers—Classifier Heterogeneity

The base classifiers leveraged by an ensemble system can differ in the following design aspects:

- The *type of the base classifier* can be a traditional ML or a DL classification algorithm. Recently, a classification head/layer has been added at the end of a pre-trained LLM to predict the final label.
- The *architecture of the base classifier* can vary in several aspects, such as the type of network architecture, the number of layers/nodes, the number of filters, the loss function, the learning rate, the dropout rate, the weight initializations, and other hyper-parameters, e.g., the batch size, the number of epochs, etc.
- The *training method of the base classifier* can be a single or multi-label training method. Classifiers can be trained using the primary label or all available labels regardless of whether they are evaluated for a single label or for multiple labels. In the case of multi-label training, the labels can be represented as multi-hot encoded vectors or probability vectors [29].

Dimension 3: Selection and Fusion Techniques

This dimension defines the combination of base classifiers that produces the final prediction. In particular, the base classifiers of an ensemble system can be combined using different selection and fusion techniques, which can fall into two broad groups:

- The *averaging techniques* are score aggregation techniques that calculate the score based on a number of evidence types, e.g., voting, or simple data fusion algorithms, e.g., mean, medium, etc. A summary of different averaging techniques can be found in [30].
- The *meta-learning techniques* involve a meta-learning stage. The most common meta-learning techniques are stacking and the mixture of experts:
- In stacking, a meta-classifier is trained on features that are the outputs of base classifiers to learn how to combine their predictions best.
- In the *mixture of experts*, an expert classifier is trained on a sub-task and then a gating model is developed that learns which expert classifier to trust each time based on the input.

Figure 1 depicts the ensemble framework for text classification and illustrates its dimensions. Each slice of the cube corresponds to a different design approach. For example, selecting the sub-cube with the pattern creates an ensemble system consisting of base classifiers trained on different functional partitions, while the outcomes of base classifiers are combined using an averaging technique.

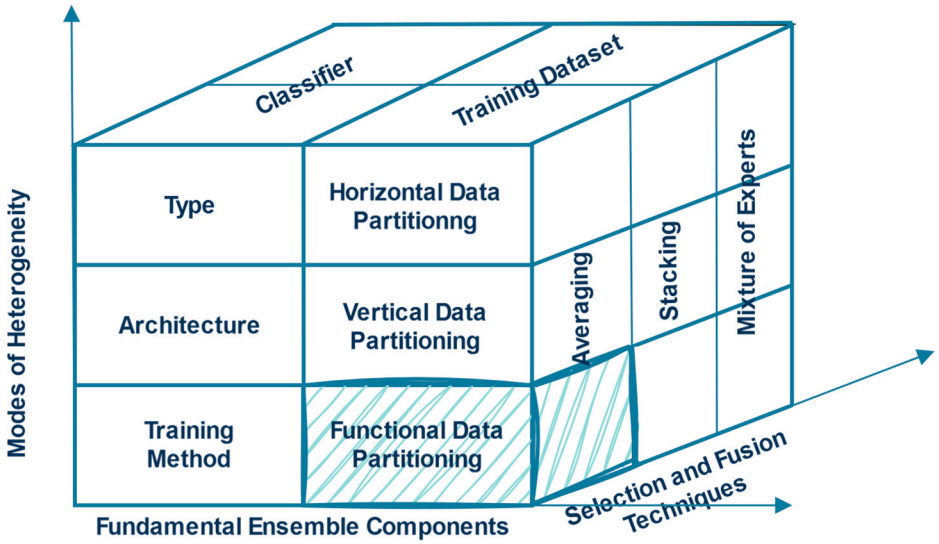

Figure 1. The ensemble framework for text classification.

4. Homogeneous vs. Heterogeneous Ensembles

When designing an ensemble system, there are two core approaches: a homogeneous or a heterogeneous approach. This section describes these two core design approaches, which will be evaluated in the following sections.

4.1. Homogeneous Approach

In the homogeneous approach, the base classifiers composing the ensemble system are identical in terms of the type, the architecture, and the training method, but they differ in the training dataset. Specifically, four representative variations of ensemble systems consisting of homogeneous classifiers trained on different training datasets are evaluated. These have been created following different data partitioning techniques:

- *Horizontal data partitioning—Bagging technique.* A well-known ensemble learning technique, named Bagging [31], is used for creating different subsets of the training dataset, called bootstraps, resampling randomly, usually with replacement, the entire training dataset.
- *Horizontal data partitioning—Adaboost technique.* Another ensemble learning technique to create different subsets of the training dataset is the boosting technique. AdaBoost [32], one of the most well-known boosting techniques, prioritizes misclassified samples during the resampling of the entire training dataset to form sequential base classifiers.
- *Vertical data partitioning—sections.* The training datasets consist of different sections/metadata, e.g., the title, the abstract, the detailed description or main body, etc. These sections can be used to train different base classifiers.
- *Functional data partitioning—label's representation.* A promising technique further described in [33] is to split the training dataset based on the labels' frequency. Different subsets of the training dataset are formed with samples having high-represented and/or low-represented labels. The threshold under/over for which a code is considered high- or low-represented can be set from 100 to 500 training samples depending on the dataset's size and labels' distribution.

4.2. Heterogeneous Approach

In the heterogeneous approach, the base classifiers are trained on the same partition of the training dataset, but they differ in the design characteristics of the classifier. Specifically,

three representative variations of ensemble systems are tested consisting of heterogeneous classifiers differentiating in the following:

- The *type ofthe base classifier.* Base classifiers may differ in the learning model. For example, different classification algorithms, such as RNN, CNN, or BERT, can be leveraged.
- The *architecture of the base classifier.* Base classifiers may differ how they structure the network architecture, with myriads of parameters available to differentiate their predictive behavior.
- The *training method of the base classifier.* Base classifiers can be trained by having as a target the primary category or the combined list of primary and secondary categories assigned by annotators [29]. For example, in patent test collections, such as the CLEFIP-0.54M and WIPO-alpha, the assigned classification codes have different priorities. The main classification code is the primary code of a patent document since it will be later classified and searched with this code. Secondary classification codes, called further, are also assigned to a patent document that corresponds to other relevant features of the invention that are not the most representative of the essential prior art, but they are still considered useful. Base classifiers can be trained in these datasets using the main or all classification codes. There are more origins of heterogeneity, e.g., classification codes can be represented as a multi-hot encoding or probability distribution vector. The probabilities can be evenly or unevenly assigned to categories based on additional annotator information, such as confidence and disagreement [29].

5. Data Collections

Five benchmark datasets—the CLEFIP-0.54M, the WIPO-alpha, the USPTO-2M, the Web of Science (WOS), and the EURLEX57K—have been used to evaluate the applicability of the proposed ensemble framework and the effectiveness of ensemble systems.

5.1. CLEFIP-0.54M

The CLEFIP-0.54M [34] contains the English patents of the CLEF-IP 2011 test collection, which have been extracted with the condition to have (i.e., not being empty) the main classification code, the EN abstract, the EN description, the EN claims, the EN title, the applicants, and the inventors. In addition to the main classification code, all (main and further) IPC classification codes at the subclass (third) level of the IPC 5+ level hierarchy are available. The dataset contains 541,131 patents classified in 731 main (and 810 main and further) subclass codes. Moreover, the text of the EN abstract, the EN description, and the EN claims has undergone a pre-processing, removing any character that is not alphabetic and English stop words.

5.2. WIPO-Alpha

The WIPO-alpha is an English patent database issued in 2002 by the World Intellectual Property Organization (WIPO). It is a data collection of about 75K XML documents distributed over 30,000 codes in the fifth level and 5500 codes in the fourth level, e.g., there is only one patent with the "A01C00502" code in the fifth level and seven patents with the "A01C005" code in the fourth level. For our experiments, we use the codes of the third level, known as subclass codes, which amount to 451 main codes and 633 all codes (main and further).

5.3. USPTO-2M

The USPTO-2M is a large-scale patent classification dataset made publicly available by Li et al. [3,35]. The dataset includes the title, the abstract, and the subclass labels (multi-label) for each patent. The dataset contains 2,000,147 patents classified in 637 categories

from 2006 to 2015. The same or subparts of this dataset have been used by other studies, such as in [36], which removes the low-represented labels with a frequency of less than 100 documents and finally considers 544 labels.

5.4. Web of Science (WOS-5736, WOS-11967, and WOS-46985)

The Web of Science (WOS) [37] is a document classification dataset of 46,985 scientific papers with 134 categories and 7 parent categories, which have been made available by the Web of Science. Each document contains two fields, the abstract, and the keywords, provided by the authors. The WOS-5736 and WOS-11967 are two subsets of the WOS-46985. The WOS-11967 contains 11,967 documents with 35 categories and 7 parent categories and the WOS-5736 contains 5736 documents with 11 categories and 3 parent categories.

5.5. EURLEX57K

The EURLEX57K [38] contains 57K English EU legislative documents from the EUR-LEX portal tagged with ~4.2K labels (concepts) from the European Vocabulary (EUROVOC). Each legislative document is provided in a JSON file containing information for a legal act (EU Directive, Regulation, Decision), as published in the Eur-Lex portal. The entire content of each legal act can be represented solely by its title, header, recitals, main body, and attachments.

Some statistics about the datasets are presented in Table 1.

Table 1. Statistics of the benchmark datasets.

Dataset	Document Type	Task(s)	#Labels	#Train/Test
CLEFIP-0.54M	Patent documents	Single and multi-label	731 main and 810 main and further	487,018/54,113
WIPO-alpha	Patent documents	Single and multi-label	451 main and 633 main and further	46,324/28,926
USPTO-2M *	Patent documents	Multi-label	544	1,947,223/49,888
WOS-5736	Scientific publications	Single-label	11	5162/574
WOS-11967			35	10,770/1197
WOS-46985			134	42,286/4699
EURLEX57K	Legislative documents	Multi-label	~4.2K	45,000/6000

* Keeping only the labels found in more than 100 patent documents.

6. Evaluate the Applicability of the Ensemble Framework

We conduct a first set of experiments to exemplify the ensemble framework's potential and evaluate the effectiveness of ensemble systems against base classifiers. Specifically, we create representative instantiations of the ensemble framework, demonstrating different ways to design an ensemble system and evaluating the effectiveness of combining knowledge against base classifiers.

6.1. Experimental Methodology in the CLEFIP-0.54M Dataset

We made our code publicly available so that other researchers could easily reproduce our experiments. Moreover, we provide some important details about our experimental parameters in this section. The first set of experiments is conducted on the CLEFIP-0.54M dataset.

6.1.1. Base Classifier

Concerning the training data, for each training sample, the first 60 words of the abstract section are used to train the base classifiers. These 60 words are mapped to embeddings using a domain-specific pre-trained language model with 300 dimensions [4].

All base classifiers employ a simple classification algorithm. This is a bidirectional LSTM (Bi-LSTM), which has been proven in a previous study to attain the best accuracy for single-label patent text classification when different RNN and CNN base classifiers, including 1D-CNN, GRU, LSTM, Bi-GRU, and Bi-LSTM, are compared to create an ensemble system [9,11]. The network architecture of base classifiers is depicted in Figure 2.

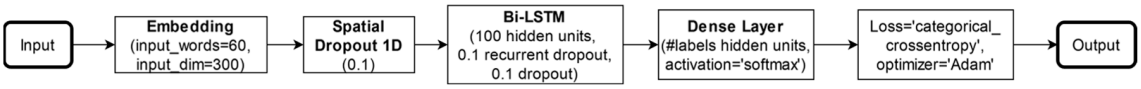

Figure 2. Bi-LSTM network architecture adopted by base classifiers of ensemble systems.

The CLEFIP-0.54M contains both the main and all IPC classification codes at the subclass level; thus, both the single- and multi-label training methods can be used. In this set of experiments, each base classifier is trained to target only the main classification code, which is encoded using the one-hot encoding scheme.

Moreover, for all experiments in this first set, the batch is set to 128 and the epochs to 15.

6.1.2. Design Details of Homogeneous Ensemble Classifiers

- *Horizontal data partitioning—Bagging technique:* Random resampling is used to create the different training datasets, while the number of base classifiers varies from 3 to 5 and to 7.
- *Horizontal data partitioning—Adaboost technique:* Weights are set to concentrate on training samples that have been classified incorrectly. The number of sequential base classifiers varies from 3 to 5 and to 7 and their final prediction is taken by adding the weighted prediction of every classifier.
- *Vertical data partitioning—sections*: In addition to the abstract, we use the title, description, and claims sections of the patent document to train different base classifiers.
- *Functional data partitioning—labels' representation*. Two base classifiers are created: one that is trained on patent documents of high-represented classification codes and one that is trained on patent documents of low-represented classification codes. The threshold under which a code is considered low-represented is set to 500 patent documents after an initial exploration of the accuracy attained as the patent frequency of codes increases.

6.1.3. Design Details of Heterogeneous Ensemble Classifiers

- The *type of the base classifier.* Different base classifiers are used in addition to a Bi-LSTM classifier. We use a bidirectional GRU (Bi-GRU), an LSTM, and a GRU classifier with similar network architectures. Details of the DL models can be found in [9].
- The *architecture of the base classifier.* Classifiers may differ in many ways concerning the network architecture. In this experiment, classifiers differ in the number of hidden units they use, adding another base classifier with 200 units.
- The *training method of the base classifier.* In the case of a heterogeneous ensemble using a different training method for base classifiers, these are trained to target either the main (i.e., single-label training method) or all IPC classification codes (i.e., multi-label training method). For the representation of all IPC classification codes, a greater probability of 0.6 is assigned to the main classification code, while the remaining

probability of 0.4 is evenly distributed among all other IPC classification codes, i.e., further classification codes. For example, if there are 4 IPC classification codes (1 main and 3 further), they are assigned a probability of 0.6, 1.33, 1.33, and 1.33, respectively. In these cases, the Kullback–Leibler (KL) divergence loss is used instead of the categorical cross-entropy.

6.1.4. Combination Method

For all ensemble variations presented in this paper, the outcome probabilities of base classifiers are fused using an averaging technique for each label to produce the final prediction.

The exception is the case of base classifiers that are trained on different labels' representations and present a significant deviation in the accuracy achieved. In this case, a meta-classifier is used to combine their outcomes, learning to distinguish their input and combine their outcomes appropriately.

The meta-classifier's network architecture is structured in two dense layers, as depicted in Figure 3. The first dense layer consists of several neurons, equal to the number of base classifiers multiplied by the number of labels and activated with a RELU function, while the second dense layer consists of neurons equal to the number of labels and activated with a sigmoid function.

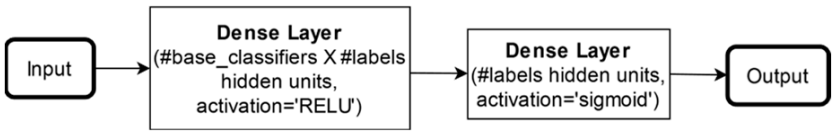

Figure 3. Meta-classifier.

6.2. Results in the CLEFIP-0.54M Dataset

The base classifiers and the ensemble system created from the combination of these base classifiers are evaluated in terms of predicting the correct main classification code (single-label classification task).

Tables 2–4 present the performance of representative variations of ensemble systems consisting of homogeneous base classifiers and the improvement achieved when applying the ensemble learning compared with the accuracy achieved by their ensemble constituents, i.e., the base classifiers, when they operate individually.

Table 2. Ensemble systems of homogeneous base classifiers using different horizontal data partitions.

	Bagging Technique			**Adaboost Technique**		
Base classifier #1	61.74%	61.93%	61.83%	63.73%	63.86%	63.73%
Base classifier #2	62.08%	61.78%	61.86%	61.98%	62.14%	61.98%
Base classifier #3	62.01%	61.74%	62.03%	62.05%	61.79%	62.01%
Base classifier #4		61.92%	61.89%		62.16%	62.02%
Base classifier #5		62.09%	61.94%		61.75%	61.82%
Base classifier #6			61.97%			62.19%
Base classifier #7			61.93%			61.88%
Ensemble	64.85%	65.70%	65.90%	65.31%	65.76%	66.07%
Improvement	2.91%	3.81%	3.98%	2.72%	3.42%	3.84%

Table 3. Ensemble systems of homogeneous base classifiers using different vertical data partitions.

Sections	
Base classifier #1 trained on titles	59.58%
Base classifier #2 trained on abstracts	63.76%
Base classifier #3 trained on descriptions	66.46%
Base classifier #4 trained on claims	64.56%
Ensemble	70.54%
Improvement	6.95%

Table 4. Ensemble systems of homogeneous base classifiers using different functional data partitions.

Labels' Representation	
Base classifier #1 trained on low-represented labels	9.37% (65.72%)
Base classifier #2 trained on high-represented labels	63.91% (68.02%)
Ensemble	68.15%
Improvement	4.39%

Specifically, Table 2 presents the accuracy of base classifiers trained on 3, 5, and 7 bootstraps created (i) using random resampling of the training dataset (Bagging technique) and (ii) using weighted resampling of the training dataset assigning a greater weight on misclassified samples (Adaboost technique). Moreover, Table 2 presents the accuracy of the ensemble system consisting of these base classifiers and the improvement achieved by the ensemble system compared with the average accuracy achieved by its constituent base classifiers.

Moreover, in Table 2, the AdaBoost and Bagging techniques of horizontal data partitioning achieve similar performance, although they manipulate the data differently. More specifically, AdaBoost initiates with slightly improved performance, as the first base classifier is trained on the entire training dataset (bootstrap). In contrast, the performance of later base classifiers is decreased as misclassified instances are preferred in respective bootstraps, resulting in a similar total performance with the Bagging technique, which creates random, unbiased bootstraps. In addition, we observe that the number of base classifiers (and bootstraps) plays a role in the total performance of the ensemble system. Specifically, the more classifiers, the higher accuracy achieved.

Table 3 depicts the accuracy of base classifiers trained on different vertical partitions, particularly different features of the patent document, including the title, abstract, description, and claims section. Moreover, it presents the accuracy of the ensemble system created by these base classifiers. As we can see, a significant improvement is recorded when we combine these base classifiers into an ensemble system, reaching an increase of 6.95% compared with the average accuracy achieved by base classifiers, confirming that the more diverse the base classifiers are, as the ones presented in Table 3, the better accuracy is achieved.

Table 4 presents the accuracy of base classifiers trained on different training subparts. Specifically, the first classifier is trained on patents with low-represented labels and the second is trained on patents with high-represented labels. Then, their outcomes are combined using a stacking method, i.e., the meta-classifier depicted in Figure 3. The main accuracy (outside brackets) is achieved when the entire testing set is evaluated, consisting of patents belonging to both groups of low- and high-represented labels, while the accuracy inside brackets is achieved when the classifier is tested only on patents that the classifier is trained to be able to handle, e.g., on patents of low-represented labels. Moreover, Table 4

presents the accuracy of the ensemble system created by these base classifiers. As we can see, the improvement of the ensemble system is 4.39% compared with the accuracy of a base classifier trained on the entire dataset (63.76%).

Table 5 presents the performance results of representative variations of ensemble systems consisting of heterogeneous base classifiers. Base classifiers differ in (i) the type, (ii) the network architecture, and (iii) the training method. All techniques can improve the ensemble system's measured accuracy compared with base classifiers. The highest improvement is recorded (2.48%) when using different types of base classifiers.

Table 5. An ensemble system of different base classifiers.

Type		**Architecture**		**Training Method**	
Bi-LSTM Bi-GRU LSTM GRU	63.76% 63.45% 63.08% 63.41%	Bi-LSTM—100 units Bi-LSTM—200 units	63.76% 64.21%	Single-label training method Multi-label training method	63.76% 63.37%
Ensemble	65.90%	Ensemble	65.59%	Ensemble	65.38%
Improvement	2.48%	Improvement	1.61%	Improvement	1.82%

Overall, the accuracy measures are improved when an ensemble technique is applied compared to base classifiers. The optimal improvement is attained using homogeneous classifiers, particularly when combining knowledge from base classifiers working on different features (6.95%), namely the title, abstract, description, and claims.

Based on this observation, we decided to expand our experiments and train base classifiers and then combine their outcomes for the rest of the sections, including the title, the description, and the claims. Thus, we create an ensemble system for each section, and then we aggregate the predictions of ensemble systems for each section, creating an ensemble of these ensembles. The accuracy of ensemble systems for each section and the accuracy of the ensemble system consisting of these ensembles is presented in Table 6. Moreover, the table presents the improvement achieved compared with the accuracy of a base classifier trained on the entire dataset (63.76%). As shown in Table 6, the best accuracy, reaching 71.10% with an improvement of 11.39%, is attained when base classifiers are trained on over- and under-represented samples for each section, and thereafter, the results of ensemble systems for each section are averaged.

Table 6. Ensemble of ensemble systems for each patent section.

	Method	**Accuracy of Ensemble Systems (ES)**				**Accuracy of Ensemble of ES**	**Improvement**
		Title	**Abstract**	**Description**	**Claims**		
Homogeneous	Bagging technique	61.31%	65.90%	68.36%	66.38%	70.80%	+7.04%
	Adaboost technique	61.38%	66.07%	68.67%	66.65%	70.93%	+7.17%
	Labels' representation	62.46%	68.15%	71.10%	68.88%	75.15%	+11.39%
Heterogeneous	Type	62.21%	66.34%	68.94%	66.91%	71.19%	+7.43%
	Architecture	61.56%	65.59%	68.06%	66.43%	71.57%	+7.81%
	Training method	60.43%	65.38%	67.69%	65.86%	70.70%	+6.94%

7. Evaluate the Effectiveness of Ensemble Systems

To validate the effectiveness of ensemble systems against SotA methods, we conduct another set of experiments. In this set, we train the same Bi-LSTM base classifiers on different vertical data partitions and combine their knowledge to create an ensemble system. Then, we compare the base classifiers and their ensemble systems with SotA methods.

At this point, it should be mentioned that the incentive for comparing ensemble systems with SotA methods is to demonstrate the improvement that can be achieved using ensemble systems compared to base classifiers, which can have comparable performance to SotA methods. In no way have we aimed to compete with recent SotA methods, and because of this, only indicative examples of SotA methods are included rather than an exhaustive list.

7.1. Experimental Methodology for Comparisons with SotA

The second set of experiments is conducted on several well-established text classification datasets: (i) two from the patent domain, including the WIPO-alpha and the USPTO-2M, and (ii) two from the text classification domain, including the WOS (the three versions of the WOS: WOS-5736, WOS-11967, and WOS-46985) and the EURLEX57K.

Depending on the data collection, different numbers of words and sections are used to train the base classifiers. In USPTO-2M, the first 100 words from the title section, the abstract section, and the concatenated title and abstract section are used. In the WIPO-alpha, the first 60 words from the title, the abstract, the description, and the claims section are used. In the EURLEX57K, the first 100 words from the title, the header, the recitals, the main body, and the attachments are used. In the WOS, the first 180 words from the keywords and the abstract section are used.

In the case of patent data collections, the feature words are mapped to word embeddings using the same domain-specific pre-trained language model as the one used in the previous section. For the WOS and EURLEX, the pre-trained GLOVE language model with 300 dimensions is used [39].

Concerning the type of classifier, all base classifiers employ a Bi-LSTM with similar architecture and hyper-parameters as those depicted in Figure 2.

In the USPTO-2M and EURLEX57K, each document is assigned more than one label, namely IPC subclass codes and EUROVOC categories, respectively (multi-label classification task). Since there is no evidence to imply that the labels' order plays a specific role, we follow a multi-label training method, and probabilities are evenly assigned among codes/categories. In the WOS, each document is assigned a single label, namely the scientific category. Thus, a single-label training method is followed, and each label is encoded using the one-hot encoding scheme. Last, the WIPO-alpha contains both the main and all IPC subclass codes, and thus, base classifiers can be trained to target the main label or all available labels. For this set of experiments, each base classifier is trained to target all IPC classification codes, which are encoded to a vector using different probabilities based on their priority (as described in Section 6.1.3). In the case of the single-label training method, the categorical cross-entropy loss function is used, while for the multi-label training method, the KL divergence loss function is used instead.

Last, the batch is set to 128 and epochs are set to 30 for the WIPO-alpha, the WOS, and the EURLEX57K, while the epochs are set to 15 for the USPTO-2M.

The outcome probabilities of the base classifiers trained on each section are fused using an averaging technique.

7.2. Results for Comparisons with SotA

In the WOS, the base classifiers and the ensemble system of these base classifiers are evaluated on the accuracy of predicting the correct scientific category. Table 7 presents the accuracy of base classifiers trained on the text coming from each section and the accuracy of the ensemble system of these base classifiers, averaging the predictions for each label across all sections. Our ensemble method achieved the best accuracy in the case of the smaller variation of the WOS (WOS-5736) and an accuracy comparable to SotA methods for the other two variations (WOS-11967 and WOS-46985).

Table 7. Experiments on the WOS-5736, WOS-11967, and WOS-46985. The light gray rows present the performance of base classifiers trained on the keyword text and abstract text, respectively, while the dark gray row presents the performance of the ensemble system of these base classifiers.

Method		WOS-5736	WOS-11967	WOS-46985
		Accuracy	Accuracy	Accuracy
SotA	CNN [40]	70.46%	83.29%	88.68%
	RNN [40]	72.12%	83.96%	89.46%
	HDLTex [41]	76.58%	86.07%	90.93%
Ensemble (Sections)		79.17%	82.79%	90.24%
Base Classifiers (Bi-LSTM)	Keywords	53.69%	69.26%	70.56%
	Abstract	76.04%	75.36%	89.72%

In the WIPO-alpha, three evaluation metrics are measured [42], the accuracy of top prediction (Top 1 vs. main), the accuracy of all categories (Top 1 vs. main + further), and the accuracy of three guesses (Top 3 vs. main). Table 8 presents these metrics for the base classifiers trained on different vertical partitions, the title, abstract, description, and claims, and the accuracy of the ensemble system of these base classifiers. As we can see, the ensemble method outperforms the SotA methods for all measures.

Table 8. Experiments on the WIPO-alpha. The light gray rows present the performance of base classifiers trained on the title, abstract, description, and claims text, respectively, while the dark gray row presents the performance of the ensemble system of these base classifiers.

Method		Top 1 vs. Main	Top 1 vs. Main + Further	Top 3 vs. Main
SotA	Bi-GRU [4]	49%	-	-
	CNN [43]	55.02%	-	-
	Bi-GRU [44]	53.76%	62.65%	76.97%
Ensemble (Sections)		58.36%	67.90%	82.41%
Base Classifiers (Bi-LSTM)	Title	44.77%	53.28%	68.50%
	Abstract	50.18%	60.06%	75.72%
	Description	54.07%	63.32%	78.74%
	Claims	50.26%	58.99%	74.79%

In the USPTO-2M and EURLEX57K, the base classifiers and the ensemble system of these base classifiers are evaluated on the precision, the recall, and the F1-score at the top K predicted labels where K is equal to one (P@1, R@1, and F1@1). These evaluation metrics are depicted in Tables 9 and 10, respectively.

Table 9 depicts the evaluation metrics in the USPTO-2M dataset. Specifically, the table presents the P@1, R@1, and F1@1 for the base classifiers trained on different vertical partitions, including the title, the abstract, and their concatenation, respectively, and the same metrics for the ensemble system of these base classifiers. As we can see, the ensemble

method underperforms or performs comparably to the SotA methods for all measures, which are mainly based on BERT and XLNet models.

Table 9. Experiments on the USPTO-2M. The light gray rows present the performance of base classifiers trained on the title, abstract, and concatenation of title and abstract text, respectively, while the dark gray row presents the performance of the ensemble system of these base classifiers.

	Method	P@1	R@1	F1@1
SotA	PatentBERT [5]	80.61%	54.33%	64.91%
	DeepPatent [3]	73.88%	-	-
	PatentNet XLNet [36]	86%	42.9%	57.2%
Ensemble (Sections)		80.16%	41.71%	54.87%
Base Classifiers (Bi-LSTM)	Title	70.64%	36.76%	48.36%
	Abstract	77.20%	40.18%	52.85%
	Concat. title and abstract	79.02%	41.12%	54.09%

Table 10. Experiments on the EURLEX57K. The light gray rows present the performance of base classifiers trained on the title, header, recitals, main body, and attachments text, respectively, while the dark gray row presents the performance of the ensemble system of these base classifiers.

	Method	P@1	P@5	R@1	R@5
SotA	BiGRU-ATT [45]	89.90%	65.40%	20.40%	68.50%
	HAN [46]	89.40%	64.30%	20.30%	67.50%
	BIGRU-LWAN [47]	90.70%	66.10%	20.50%	69.20%
	BERT-BASE [38]	92.20%	68.70%	20.90%	71.90%
Ensemble (Sections)		89.07%	64.78%	17.60%	64.02%
Base classifiers (Bi-LSTM)	Title	84.83%	61.37%	16.77%	60.65%
	Header	85.35%	61.29%	16.87%	60.57%
	Recitals	83.10%	60.21%	16.43%	59.50%
	Main body	81.45%	59.94%	16.10%	59.23%
	Attachments	52.67%	38.42%	10.41%	37.97%

Table 10 depicts the evaluation metrics in the EURLEX57K dataset. Specifically, the table presents the P@1, R@1, P@5, and R@5 for the base classifiers trained on different vertical partitions, including the title, header, recitals, main body, and attachments, respectively, and the same metrics for the ensemble system of these base classifiers. As we can see, the ensemble method achieved improvements compared to the base classifiers, while it performed comparably to the SotA methods, which are mainly based on BiGRU, BERT, and HAN.

In the experiments with the WOS and WIPO-alpha datasets (Tables 7 and 8), the base classifiers utilize a Bi-LSTM classification algorithm, which is similar to the classification algorithms applied in the respective SotA methods, such as CNN, RNN, and GRU. Therefore, the comparison was made on an equal basis. On the other hand, in the experiments with the USPTO-2M and EURLEX57K datasets (Tables 9 and 10), we penalize our ensemble model since we compare it with SotA methods utilizing the BERT and XLNet classifiers.

8. Discussion

The primary objective of this paper was to present the design principles for developing ensemble systems for text classification. Ensemble learning is a technique that can achieve considerable gains in any individual model's prediction performance, producing results with high variance. Following the principles described in the proposed ensemble

framework, data scientists have a valuable roadmap for developing base classifiers and using them as building blocks for effective and more robust ensemble systems. Although we selected a simple classification algorithm, i.e., the Bi-LSTM, to demonstrate the benefits of ensemble systems, someone can apply the ensemble technique on more complex classification algorithms, such as BERT or XLNet.

We can compare the operation of ensemble models with the process of experts reviewing a research publication or a research proposal. The more reviewers assigned to the evaluation, the more likely it is that a reliable decision will be made, and all different aspects that could affect a decision will be detected and taken into account.

Hence, an ensemble system combines the results of base models into a more stable and better-performing model. Ensemble systems can be applied in any task where base models show high bias and/or high variance. Additionally, ensembles can improve performance if multiple signals exist, and their fusion can be combined. Finally, ensemble systems are better fit in professional domains where the gains of increased reliability are worth the extra effort in terms of time and resources, such as medical, legal, or patent domains.

One representative example from the patent field is when a reclassification is required due to changes in IPC/CPC. Many patent documents should be reclassified. This update should be carried out with the highest possible degree of reliability and confidence, as misclassification will have significant consequences. In this scenario, it would be worth any additional effort (in time and resources) to employ an ensemble.

Finally, as shown in the experiments, the performance gains achieved using the ensemble technique can be compared with those attained with SotA methods for text classification, e.g., transformers-based models. Hence, ensemble architectures consisting of simpler base models can be compared and even counterbalance the use of newer models, offering an alternative solution in cases of computational constraints.

8.1. Document Classification

Documents include many sources of information that can be used as signals by classifiers. For example, a patent document contains thousands of words that can be used as features. Using all available document features for training a classification model may be neither efficient nor effective. Therefore, strategies for addressing multiple features are required. Ensemble architectures, consisting of numerous base classifiers, can be trained on different features and combine their outcomes appropriately to produce the final prediction. This way, they can exploit many features, achieving better results, and counterbalance the failure of some.

Moreover, despite its ability for parallelization, the attention mechanism used by transformers-based models makes it difficult to process long texts, e.g., the BERT model handles 512 tokens, omitting a significant portion of content that may potentially point to additional labels due to the locality of concepts that is evident in long documents. Appropriate ensemble strategies are needed to adapt LLMs for long documents, splitting them into smaller chunks, processing each chunk individually and later combining the outcomes of all chunks to produce a final prediction [48].

8.2. Efficiency vs. Complexity

Complexity is translated in time and resources, including memory and CPU/GPU. The imposed complexity of ensemble models lies in the number of base models and the combination technique used to combine their outcomes. Thus, the complexity may vary significantly, from a simple ensemble system composed of two base models whose results are combined using an average function to a complicated ensemble system consisting of

many base models whose results are combined in a complicated way, e.g., using stacking techniques, the selective combination of models, etc.

Let us give a specific example comparing the SotA classifiers' complexity with the ensemble classifier's complexity in Table 10. To achieve that, we quantify the complexity of the development of a system using a simple system complexity metric, counting the number of nodes and the one-way interactions they have. The complexity of SotA classifiers is always one since only one classifier is involved. The complexity of developing the ensemble system increases linearly as a new base classifier is added, reaching six, as five base classifiers use a simple combination method, such as the averaging technique, to fuse their results.

Concerning the complexity of deploying ensemble systems (inference), base classifiers can operate in parallel; thus, no additional delays are added to the total inference time provided that their results are fused using a simple combination technique. Therefore, no significant scalability concerns may be raised by adopting an ensemble solution instead of a single-model solution.

Last, as mentioned above, ensemble architectures of simpler base models can achieve comparable results with SotA methods. Therefore, ensemble systems can be a suitable substitute for recent resource-intensive models, e.g., transformers-based models, in cases of computational constraints.

9. Conclusions

We proposed an ensemble framework for text classification that can be used to structure the design of new ensemble systems. Following the framework's principles, researchers can define the base classifiers of ensemble systems and how to include diversity across them. Moreover, we have shown that ensembles of classifiers outperform base classifiers and perform comparably to similar previous work on text classification.

In this work, we have focused on long and short texts from patents and scientific and legislative documents. Yet, our framework should be easily adaptable to other genres and domains. Further improvements for text classification can be expected from an ensemble of classifiers combining additional textual information, e.g., unexplored sections, or combining parts that articulate the topic(s) better, e.g., essential parts or significant words. Finally, as ensembles of classifiers improve the attained performance of base classifiers, transforming existing text classification models with simple and low-cost ensemble techniques, like combining the knowledge from all sections, may be a promising direction to improve the text classification performance.

Author Contributions: Conceptualization, E.K. and M.S.; methodology, E.K. and M.S.; software, E.K.; validation, E.K.; formal analysis, E.K.; investigation, E.K.; resources, E.K.; data curation, E.K.; writing—original draft preparation, E.K.; writing—review and editing, E.K. and M.S.; visualization, E.K.; supervision, M.S.; funding acquisition, E.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research work was supported by the Hellenic Foundation for Research and Innovation (HFRI) under the HFRI PhD Fellowship grant (Fellowship Number: 10695).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The code used for running the experiments of this article are available in the following GitHub repository: https://github.com/ekamater/Ensemble-Framework-for-Text-Classification (accessed on 18 October 2024).

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Grawe, M.F.; Martins, C.A.; Bonfante, A.G. Automated patent classification using word embedding. In Proceedings of the 2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA), Cancun, Mexico, 18–21 December 2017; pp. 408–411.
2. Xiao, L.; Wang, G.; Zuo, Y. Research on patent text classification based on word2vec and LSTM. In Proceedings of the 2018 11th International Symposium on Computational Intelligence and Design (ISCID), Hangzhou, China, 8–9 December 2018; Volume 1, pp. 71–74.
3. Li, S.; Hu, J.; Cui, Y.; Hu, J. DeepPatent: Patent classification with convolutional neural networks and word embedding. *Scientometrics* **2018**, *117*, 2. [CrossRef]
4. Risch, J.; Krestel, R. Domain-specific word embeddings for patent classification. *Data Technol. Appl.* **2019**, *53*, 108–122. [CrossRef]
5. Lee, J.-S.; Hsiang, J. Patent classification by fine-tuning BERT language model. *World Pat. Inf.* **2020**, *61*, 101965. [CrossRef]
6. Pujari, S.C.; Friedrich, A.; Strötgen, J. A multi-task approach to neural multi-label hierarchical patent classification using transformers. In *Advances in Information Retrieval: 43rd European Conference on IR Research, ECIR 2021, Virtual Event, 28 March–1April 2021, Proceedings, Part I 43*; Springer International Publishing: Cham, Switzerland, 2021; pp. 513–528.
7. Zhou, Z.-H.; Wu, J.; Tang, W. Ensembling neural networks: Many could be better than all. *Artif. Intell.* **2002**, *137*, 239–263. [CrossRef]
8. Benites, F.; Malmasi, S.; Zampieri, M. Classifying patent applications with ensemble methods. *arXiv* **2018**, arXiv:1811.04695.
9. Kamateri, E.; Stamatis, V.; Diamantaras, K.; Salampasis, M. Automated Single-Label Patent Classification using Ensemble Classifiers. In Proceedings of the 2022 14th International Conference on Machine Learning and Computing (ICMLC), Guangzhou, China, 18–21 February 2022; pp. 324–330.
10. Hong, Z.; Wenzhen, J.; Guocai, Y. An effective text classification model based on ensemble strategy. *J. Phys. Conf. Ser.* **2019**, *1229*, 012058. [CrossRef]
11. Kamateri, E.; Salampasis, M.; Diamantaras, K. An ensemble framework for patent classification. *World Pat. Inf.* **2023**, *75*. [CrossRef]
12. Minaee, S.; Kalchbrenner, N.; Cambria, E.; Nikzad, N.; Chenaghlu, M.; Gao, J. Deep learning—Based text classification: A comprehensive review. *ACM Comput. Surv. (CSUR)* **2021**, *54*, 1–40. [CrossRef]
13. Sun, X.; Li, X.; Li, J.; Wu, F.; Guo, S.; Zhang, T.; Wang, G. Text Classification via Large Language Models. *arXiv* **2023**, arXiv:2305.08377.
14. Mohammed, A.; Kora, R. A novel effective ensemble deep learning framework for text classification. *J. King Saud Univ.-Comput. Inf. Sci.* **2022**, *34*, 8825–8837. [CrossRef]
15. Mohammed, A.; Kora, R. A comprehensive review on ensemble deep learning: Opportunities and challenges. *J. King Saud Univ.-Comput. Inf. Sci.* **2023**, *35*, 757–774. [CrossRef]
16. Larkey, L.S.; Croft, W.B. Combining classifiers in text categorization. In Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Zurich, Switzerland, 18–22 August 1996; pp. 289–297.
17. Boroš, M.; Maršík, J. Multi-label text classification via ensemble techniques. *Int. J. Comput. Commun. Eng.* **2012**, *1*, 62–65. [CrossRef]
18. Anderlucci, L.; Guastadisegni, L.; Viroli, C. Classifying textual data: Shallow, deep and ensemble methods. *arXiv* **2019**, arXiv:1902.07068.
19. Dong, Y.-S.; Han, K.-S. A comparison of several ensemble methods for text categorization. In Proceedings of the 2004 IEEE International Conference on Services Computing (SCC), Shanghai, China, 15–18 September 2004; pp. 419–422.
20. Gangeh, M.J.; Kamel, M.S.; Duin, R.P. Random subspace method in text categorization. In Proceedings of the 2010 20th International Conference on Pattern Recognition (ICPR), Istanbul, Turkey, 23–26 August 2010; pp. 2049–2052.
21. Xia, R.; Zong, C.; Li, S. Ensemble of feature sets and classification algorithms for sentiment classification. *Inf. Sci.* **2011**, *181*, 1138–1152. [CrossRef]
22. Jurek, A.; Bi, Y.; Wu, S.; Nugent, C. A survey of commonly used ensemble-based classification techniques. *Knowl. Eng. Rev.* **2014**, *29*, 551–581. [CrossRef]
23. Xu, C.; Tao, D.; Xu, C. A survey on multi-view learning. *arXiv* **2013**, arXiv:1304.5634.
24. Gonçalves, C.A.; Vieira, A.S.; Gonçalves, C.T.; Camacho, R.; Iglesias, E.L.; Diz, L.B. A novel multi-view ensemble learning architecture to improve the structured text classification. *Information* **2022**, *13*, 283. [CrossRef]
25. Kumar, V.; Minz, S. Multi-view ensemble learning: An optimal feature set partitioning for high-dimensional data classification. *Knowl. Inf. Syst.* **2015**, *49*, 1–59. [CrossRef]
26. Hu, X. Using rough sets theory and database operations to construct a good ensemble of classifiers for data mining applications. In Proceedings of the 2001 IEEE International Conference on Data Mining, San Jose, CA, USA, 29 November–2 December 2001; pp. 233–240.

27. Kuncheva, L.I.; Whitaker, C.J. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Mach. Learn.* **2003**, *51*, 181–207. [CrossRef]
28. Brown, G.; Wyatt, J.; Harris, R.; Yao, X. Diversity creation methods: A survey and categorisation. *Inf. Fusion* **2004**, *6*, 5–20. [CrossRef]
29. Wu, B.; Li, Y.; Mu, Y.; Scarton, C.; Bontcheva, K.; Song, X. Don't waste a single annotation: Improving single-label classifiers through soft labels. In *Findings of the Association for Computational Linguistics: EMNLP 2023*; Association for Computational Linguistics: Singapore, 2023; pp. 5347–5355.
30. Paltoglou, G.; Salampasis, M.; Satratzemi, M. Simple adaptations of data fusion algorithms for source selection. In Proceedings of the 31th European Conference on Information Retrieval, Toulouse, France, 6–9 April 2009; pp. 497–508.
31. Freund, Y.; Schapire, R.E. Experiments with a new boosting algorithm. In Proceedings of the 13th International Conference on International Conference on Machine Learning, ICML'96, Bari, Italy, 3–6 July 1996; Volume 96, pp. 148–156.
32. Bühlmann, P. Bagging, boosting and ensemble methods. In *Handbook of Computational Statistics*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 985–1022.
33. Kamateri, E.; Salampasis, M. Ensemble Method for Classification in Imbalanced Patent Data. In Proceedings of the 4th Workshop on Patent Text Mining and Semantic Technologies (PatentSemTech 2023 in Conjunction with SIGIR 23), Taipei, Taiwan, 27 July 2023.
34. CLEFIP-0.54M. Available online: https://github.com/ekamater/CLEFIP-0.54M (accessed on 18 October 2024).
35. USPTO-2M. Available online: https://github.com/JasonHoou/USPTO-2M (accessed on 18 October 2024).
36. Haghighian Roudsari, A.; Afshar, J.; Lee, W.; Lee, S. PatentNet: Multi-label classification of patent documents using deep learning based language understanding. *Scientometrics* **2021**, *127*, 207–231. [CrossRef]
37. Kowsari, K.; Brown, D.; Heidarysafa, M.; Jafari Meimandi, K.; Gerber, M.; Barnes, L. Web of Science Dataset. Mendeley Data. 2018. V6. Available online: https://data.mendeley.com/datasets/9rw3vkcfy4/6 (accessed on 18 October 2024).
38. Chalkidis, I.; Fergadiotis, M.; Malakasiotis, P.; Androutsopoulos, I. Large-Scale Multi-Label Text Classification on EU Legislation. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL 2019), Florence, Italy, 28 July–2 August 2019.
39. Pennington, J.; Socher, R.; Manning, C.D. Glove: Global vectors for word representation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 25–29 October 2014; pp. 1532–1543.
40. Zhang, X.; Zhao, J.; LeCun, Y. Character-level convolutional networks for text classification. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 649–657.
41. Kowsari, K.; Brown, D.E.; Heidarysafa, M.; Meimandi, K.J.; Gerber, M.S.; Barnes, L.E. Hdltex: Hierarchical deep learning for text classification. In Proceedings of the 2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA), Cancun, Mexico, 18–21 December 2017; IEEE: Piscataway, NJ, USA, 2017; pp. 364–371.
42. Fall, C.J.; Törcsvári, A.; Benzineb, K.; Karetka, G. Automated categorization in the international patent classification. *ACM SIGIR Forum* **2003**, *37*, 10–25. [CrossRef]
43. Abdelgawad, L.; Kluegl, P.; Genc, E.; Falkner, S.; Hutter, F. Optimizing neural networks for patent classification. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*; Springer International Publishing: Cham, Switzerland, 2019; pp. 688–703.
44. Aroyehun, S.T.; Angel, J.; Majumder, N.; Gelbukh, A.; Hussain, A. Leveraging label hierarchy using transfer and multi-task learning: A case study on patent classification. *Neurocomputing* **2021**, *464*, 421–431. [CrossRef]
45. Xu, K.; Ba, J.; Kiros, R.; Cho, K.; Courville, A.; Salakhudinov, R.; Zemel, R.; Bengio, Y. Show, attend and tell: Neural image caption generation with visual attention. In Proceedings of the 32th International Conference on Machine Learning, Lille, France, 6–11 July 2015; pp. 2048–2057.
46. Yang, Z.; Yang, D.; Dyer, C.; He, X.; Smola, A.; Hovy, E. Hierarchical attention networks for document classification. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego, CA, USA, 12–17 June 2016; pp. 1480–1489.
47. Mullenbach, J.; Wiegreffe, S.; Duke, J.; Sun, J.; Eisenstein, J. Explainable prediction of medical codes from clinical text. *arXiv* **2018**, arXiv:1802.05695.
48. Gao, S.; Alawad, M.; Young, M.T.; Gounley, J.; Schaefferkoetter, N.; Yoon, H.J.; Wu, X.-C.; Durbin, E.B.; Doherty, J.; Stroup, A.; et al. Limitations of transformers on clinical text classification. *IEEE J. Biomed. Health Inform.* **2021**, *25*, 3596–3607. [CrossRef] [PubMed]