Applied Intelligence (2022) 52:17465–17489
https://doi.org/10.1007/s10489-022-03281-1

Word-level human interpretable scoring mechanism for novel text detection using Tsetlin Machines

Bimal Bhattarai[1] · **Ole-Christoffer Granmo**[1] · **Lei Jiao**[1]

Accepted: 20 January 2022 / Published online: 2 April 2022

Abstract

Recent research in novelty detection focuses mainly on document-level classification, employing deep neural networks (DNN). However, the black-box nature of DNNs makes it difficult to extract an exact explanation of why a document is considered novel. In addition, dealing with novelty at the word level is crucial to provide a more fine-grained analysis than what is available at the document level. In this work, we propose a Tsetlin Machine (TM)-based architecture for scoring individual words according to their contribution to novelty. Our approach encodes a description of the novel documents using the linguistic patterns captured by TM clauses. We then adapt this description to measure how much a word contributes to making documents novel. Our experimental results demonstrate how our approach breaks down novelty into interpretable phrases, successfully measuring novelty.

Keywords Natural language processing · Novelty detection · Tsetlin Machine · Interpretable learning · Explainable AI

1 Introduction

The fundamental principle underlying machine learning classifiers is a generalization – the ability to form a decision boundary that differentiates new input into known classes. When training a supervised classifier, it is common to assume that the classes to be recognized are present both in the training and test data [49]. However, given an open world, training on all conceivable classes of input is impractical. This problem introduces the need for *novelty detection* – the task of spotting input classes that one has not seen before. The problem is particularly severe in text-based supervised classification due to the many-faceted nature of natural language, which gives rise to multiple application-dependent interpretations. Indeed, researchers have for a long time tried to address novelty detection in natural language. So far, no single best model has appeared. Indeed, the success of each model relies on the properties of each particular dataset.

The problem of novelty detection arises in many tasks, such as fault detection [16] and handwritten alphabet recognition [54]. In general, one applies novelty detection when it is required to know whether a given input is similar to or significantly different from the training data. For natural language text, the novelty detector should discern that a text does not belong to a predefined set of topics. Several challenges make such novelty detection particularly difficult:

1. Textual information tends to be diverse, composed of large vocabularies.
2. Language and topics are typically evolving, making the novelty detection problem dynamic [21].

Lately, the aforementioned challenges have manifested when using supervised learning to build chatbots, an application area that is gaining traction. A chatbot typically needs to handle the language of a multitude of users with evolving information requirements. As such, it must be able to determine when it is capable of answering a query and when it faces a new topic.

Majority of the existing literature on text-based novelty detection addresses one of the following granularity levels:

✉ Bimal Bhattarai
bimal.bhattarai@uia.no; bobsbimal58@gmail.com

Ole-Christoffer Granmo
ole.granmo@uia.no

Lei Jiao
lei.jiao@uia.no

[1] Centre for Artificial Intelligence Research (CAIR), University of Agder, Grimstad, Norway

1. Event-level techniques [4] perform topic detection and tracking on a stream of documents.
2. Document-level techniques [17] classify an incoming document as known or novel based on its content.
3. Sentence-level techniques [6] look for novel sentences within a particular document.

Usually, the sentences/documents are ranked based on some sort of similarity score, obtained from comparing them with previously seen sentences/documents. For instance, the Maximal Marginal Relevance model (MMR) proposed in [14] assigns low scores to previously seen sentences/documents, while assigning high scores to novel ones.

Figure 1 illustrates the problem of novelty detection, contrasting it against anomaly and outlier detection. Anomaly detection [15] concerns discovering anomalies, which are invalid data points. Outlier detection [3, 29], on the other hand, flags legitimate data points that deviate significantly from the mean. Finally, novelty detection [43] is the discovery of entirely new types of data points.

In contrast to prior work, we here focus on novelty detection at the word level. To this end, we propose a new interpretable machine learning approach for calculating novelty scores for the words within a sentence. The calculation is based on the linguistic patterns captured by a Tsetlin Machine (TM) in the form of AND-rules (i.e., conjunctive clauses). To the best of our knowledge, this is the first study of its kind on this problem.

Problem definition In the supervised classification setting, i pre-labeled data points $D = \{(v_1, y_1), (v_2, y_2), \ldots, (v_i, y_i)\}$ is used for training. Here, v_i is the i^{th} input example and y_i is its class. The input v_i is an t-dimensional real-valued vector $(x_1, x_2, \ldots, x_o) \in \mathbb{R}^t$, where x_o refers to the o^{th} element of the vector. The class $y_i \in Y = \{1, 2, \ldots, C_l\}$, in turn, is an integer class index referring to one out of C_l classes. Learning a classifier entails constructing a classification function $f(v; D)$, $f : \mathbb{R}^t \rightarrow Y$, based on the data D. The function simply assigns a label y to the data point v. Our emphasis is novelty scoring, which can be seen as another function $z(v; D), z : \mathbb{R}^t \rightarrow \mathbb{R}$. The function computes a real-valued novelty score for input data point v, with the purpose of discerning new classes not found in Y. In this way, a classifier can return the correct class label while flagging novel examples. Considering each element in v to represent a specific word, this paper further extend the novelty detection by introducing a method for breaking down the overall score $z(v; D)$ for v into the contribution of each element x_o. By doing so, we break down novelty into interpretable phrases.

Paper contributions In this paper, we use the TM to construct conjunctive clauses in propositional logic. In this manner, we capture frequent patterns in the data D, which we then utilize to characterize the known classes Y comprehensively. The novelty score is then calculated based on examining the clauses that match the given input. By further looking into the composition of each clause, we are able to break down the novelty score into the contribution of the different phrases. This decomposition is based on training clauses for the novel data and then measuring the relative frequency of each word inside the clauses for the known classes, contrasted against the relative frequency obtained from the clauses of the novel class. These scores can, in turn, be adopted as input features to machine learning classifiers for novelty detection. Similarly, contextual scores can be calculated simply by inspecting each word's clauses, providing a local perspective for both novel and known classes.

The remainder of the paper is organized as follows. In Section 2, we first summarize related work before we present the details of the TM in Section 3. This forms the basis for our novelty description architecture, covered in Section 4. In Section 5, we present our empirical results, concluding the work in the last section.

2 Related work

Several studies have been carried out on supervised multiclass classification in a closed-world setting [5]. There is a dearth of work addressing open-world settings [33],

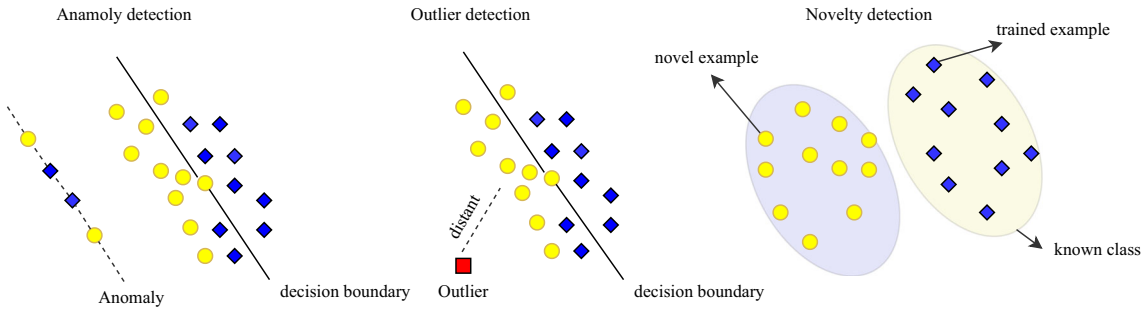


Fig. 1 Visualization of outlier detection, anomaly detection and novelty detection

with distance-based methods being one of the earliest approaches [28]. These approaches rely on nearest neighbor search, which introduces scalability issues when dealing with larger datasets. Another class of methods are based on single-class classifiers. These include One-Class SVM [50] and SVDD [55]. Further, the decision score from SVM has been used to produce a probability distribution for novelty detection [44]. As no negative training samples are used, single-class classifiers struggle with maximizing the class margin. To overcome the problem of One-Class SVMs, a new learning method named center-based similarity space (CBS) was proposed in [20], which transforms each document in a closed boundary to a central similarity vector that can be used in a binary classifier.

Probabilistic methods have also been utilized for novelty detection [43]. In [30], a technique to threshold the entropy of the estimated class probability distribution is proposed. In that method, choosing the entropy threshold needs prior knowledge. Additionally, the class probability distribution can be misleading when novel data points fall far from the decision boundary. In [32] and [46], an active learning model is proposed to both discover and classify novel classes during training. However, the appearance of novel instances during testing is not considered.

DNNs have recently been used to address the problem of novelty detection. In [61], a two-class SVM classifier is adopted to categorize known and novel classes. An adversarial sample generation (ASG) framework [23] is used to generate positive and negative samples. Similarly, [37] employs generative adversarial networks (GANs), where the generator produces a mixture of known and novel data. The generator is trained with so-called feature matching loss, and the discriminator performs simultaneous classification and novelty detection. In computer vision, the problem of novel image detection is addressed by introducing the concept of open space risk [49]. This is achieved by reducing the half-space of a binary SVM classifier with two parallel hyperplanes that bound the positive region. Although the binary SVM reduces the positive region to half-spaces, their open space risk is still infinite. In [5], a method called OpenMAX is proposed, which estimates the probability of an input belonging to a novel class. In general, the major weaknesses of these methods are high computational complexity and uninterpretable inference. A state-of-the-art GAN-based method for unsupervised outlier detection called Single-Objective Generative Adversarial Active Learning (SO-GAAL) and Multi-Objective Generative Adversarial Active Learning (MO-GAAL) was proposed in [41]. The method is based on a min-max game between a generator and a discriminator. The training process of the generator is paused before convergence to synthesize outliers, which is subsequently used to train the discriminator to recognize the outliers. However, the method is primarily designed for high-dimensional data, requiring extensive problem-specific hyperparameter tweaking. The unsupervised learning method COPOD [40] is a more recent approach that is inspired by copulas for modeling multivariate data distributions. In comparison to other methods, COPOD is computationally efficient, interpretable, and is unaffected by feature dimension. However, the method fails to handle complex features and intricate nonlinear relations.

Apart from the studies on the document-level novelty detection, novelty detection at the event level arises from topic detection, which focuses on the online event and story detection [38]. The study at the event level primarily consists of clustering algorithms that measure the closeness of incoming events or stories to one of the clusters depending on a pre-defined threshold. Novelty detection at the sentence level was investigated in Text Retrieval Conferences (TREC) by highlighting sentences that include novel information given a topic and a list of documents [52]. Based on TREC, many studies have been conducted on novelty detection at sentence level [56, 63], including term translations, Principal Component Analysis (PCA) vectors, Support Vector Machine (SVM) classification, named entities patterns, etc. Likewise, a few approaches have been introduced for learning sentence embeddings, including SkipThought [36], Conceptual Sentence Embedding [58], and FastSent [31]. However, these approaches on embeddings are very dependent on the domain-specific downstream tasks. Recently introduced powerful language models, such as ELMo [42] and BERT [18], have been successful for transfer learning and they are able to learn dynamic sentence embedding in an unsupervised manner.

In [22], a unified attention architecture is proposed to deal with vector representations of text input in NLP. The authors investigate how information can be retrieved from attention in NLP. Further, [51] checks whether the attention weights provide any interpretability by manipulating the weights in pretrained text classification models. They used an intermediate representation erasure method to demonstrate that attention weights are unreliable predictors of the relative significance of the specific input. They thus do not accurately explain the model's decision-making. Additionally, [53] employed a novel approach for visualizing the attention score for each token. This is the first study on interpretability analysis by visualizing and scoring at the word level. However, as explained in [34], the scoring acquired using attention methods does not provide a meaningful explanation. A more advanced scoring method known as Masked Language Model (MLM) [48] uses pretrained MLM to score sentences using pseudo-log-likelihood scores (PLLs), which involves masking each token one by one. The method becomes unsuitable for scoring the entire tokens of the dataset as the computational complexity rises.

Likewise, recent keyword extraction (KE) algorithms such as YAKE [13] and KeyBERT [26] are also used to extract the top-scoring tokens from the trained model. To the best of our knowledge, in novelty detection, there exists no such method to measure each word's contribution to the novelty. In this study, we expand the study on novelty detection with a method for scoring each word's contribution to the overall novelty, which offers a clear view to the researchers for the reasoning and the interpretation of the results that the algorithm offers.

3 Tsetlin machine (TM) architecture

The TM, proposed in [24], is a recent approach to pattern classification, regression, and novelty detection [1, 8, 25]. It captures the frequent patterns of the learning problem using conjunctive clauses in propositional logic. Each clause is a conjunction of literals, where a literal is a propositional/Boolean variable or its negation. Recent research reports that the TM performs competitively with state-of-the-art deep learning networks in text classification [7, 47, 59, 60] along with parallel and asynchronous architecture [2] for faster learning across diverse tasks. Further, theoretical studies have uncovered robust convergence properties [35, 62].

A basic TM accepts a vector $X = (x_1, \ldots, x_o) \in \{0, 1\}^o$ of o Boolean features as input. For text input, it is typical to booleanize the text to form a Boolean set of words, as suggested in [7]. The input features, together with their negated counterparts, $\bar{x} = \neg x = 1 - x$, form a literal set $L = \{x_1, \ldots, x_o, \neg x_1, \ldots, \neg x_o\}$. For classification problems, the sub-patterns associated with the classes are captured by the TM using m conjunctive clauses C_j^+ or C_j^-. The $j = 1, \ldots, m/2$ subscript denotes the clause index, while the superscript indicates the *polarity* of a clause. In brief, half of the clauses are assigned positive polarity, i.e., C_j^+, and the other half are assigned negative polarity, i.e., C_j^-. The positive polarity clauses vote for the input belonging to the class favored by the TM, while the negative polarity clauses vote against that class, that is, for other classes.

A clause $C_j^\xi, \xi \in \{-, +\}$, is formed by ANDing a subset $L_j^\xi \subseteq L$ of the literal set. That is, the set of literals for clause C_j^ξ with polarity ξ can be written as:

$$C_j^\xi(X) = \bigwedge_{l \in L_j^\xi} l = \prod_{l \in L_j^\xi} l. \tag{1}$$

The clause evaluates to 1 if and only if all the literals of the clause also evaluate to 1. For example, the clause $C_j^\xi(X) = x_1 x_2$ consists of the literals $L_j^\xi = \{x_1, x_2\}$ and outputs 1, if $x_1 = x_2 = 1$. The final classification decision is obtained by subtracting the negative votes from the positive votes, and then thresholding the resulting sum using the unit step function u:

$$\hat{y} = u\left(\sum_{j=1}^{m/2} C_j^+(X) - \sum_{j=1}^{m/2} C_j^-(X)\right). \tag{2}$$

For example, the classifier $\hat{y} = u(x_1\bar{x}_2 + \bar{x}_1 x_2 - x_1 x_2 - \bar{x}_1\bar{x}_2)$ captures the XOR-relation.

For learning, the TM employs a team of Tsetlin Automata (TA), one TA per literal $l \in L$. Each TA performs one of two actions: either *include* or *exclude* its designated literal. Each clause statistically forwards the feedback to its individual TA. The TM employs Type I and Type II feedback. These feedback types control the reward, penalty or inaction received by TAs depending on six factors: (1) target output ($y = 0$ or $y = 1$), (2) clause polarity, (3) clause output ($C_j = 0$ or 1), (4) literals value ($x = 1$, or $\neg x = 1$), (5) vote sum, and (6) the current state of the TA. Type I feedback is designed to produce frequent patterns, while Type II feedback increases the discriminating power of the patterns (see [25] for details). The feedback guides the complete system of TAs towards a Nash equilibrium. At any point in the training process, we have m conjunctive clauses per class, half of them positive and half of them negative. These can be retrieved and deployed upon completion of training.

4 Novelty description

By novelty description, we mean the task of characterizing novel textual content at the word level. For instance, the known content may be reviews of mobile phones, while the novel content could be reviews of grocery stores. For this example, one may define the novel content using words associated with grocery stores. However, describing novelty at the word level is nontrivial because the meaning of words varies depending on the context they appear in. For example, consider the word "bat". This word typically manifests in two distinct contexts- it can denote either "animal" or "sports". Likewise, the word "bank" can refer to "river bank" or "cash bank". That is, when contextual meaning is considered, the novelty of the word "bat" and "bank" can be different based on their respective uses. As a result, measuring and describing novel content is a challenging problem.

In general, one can detect and characterize novel content by contrasting against the probability of observing textual content X, given that the content is known. We denote this probability distribution by $p_{known}(X)$. Assume that the corresponding probability distribution $p_{novel}(X)$ for

novel content also is available. Then, the optimal novelty detection test for a given false positive rate (α) can be obtained by thresholding the likelihood ratio $p_{novel}(X)$ / $p_{known}(X)$ [39].

Since neither $p_{known}(X)$ or $p_{novel}(X)$ are available to us, we must estimate them using training examples. Inspired by the work in [9] on Semi-Supervised Novelty Detection (SSND), we use two sets of examples. One set represents known content, while the other represents novel content. We obtain these sets by employing a binary classifier that can distinguish between known and novel content, such as the one we proposed in [8].

4.1 Identifying novel word candidates

In our approach, we begin by training a TM on input texts represented as Boolean bag-of-words, i.e., as word *sets*. A propositional variable represents each word in the vocabulary, capturing the presence/absence of the corresponding word in the input text. We group the texts into two classes, *Known* and *Novel*. The first represents known content, and the second represents novel content. Our task is to describe how the second group of text is novel at the word level. To this end, we begin by identifying novel word candidates, followed by scoring and ranking the words based on their contribution to novelty.

Figure 2 shows our architecture for identifying novel word candidates. As seen, upon training, we obtain the clauses of the two classes, *Known* and *Novel*. We extract all the words included in the clauses for each class. Each clause contains a combination of both plain ($\mathcal{P}_L$) and negated ($\mathcal{N}_L$) words. As such, the plain and the negated words serve two different roles. The plain words characterize the corresponding class, while the negated words characterize the other class. We exploit this property as follows, building two bag-of-words (BOW). The first is a bag of known words, referred to as $\mathcal{B}_\mathcal{K}$, and the second is a bag of novel words, referred to as $\mathcal{B}_N$.

For class *Known*, we perform the following procedure:

- We consider the words included in positive clauses first. Here, the plain words $\mathcal{P}_L$ are added to the bag of known words $\mathcal{B}_K$, while the negated words are placed in the bag of novel words $\mathcal{B}_N$.
- For negative clauses, we do the opposite. The plain words $\mathcal{P}_L$ are added to the novel words bag $\mathcal{B}_N$. The negated words $\mathcal{N}_L$, on the other hand, are added to the known word bag $\mathcal{B}_K$.

The above procedure is inverted for class *Novel*:

- For the positive clauses, the plain words $\mathcal{P}_L$ are added to the novel word bag $\mathcal{B}_N$, while the negated words are added to the known word bag $\mathcal{B}_K$.
- Conversely, for the negative clauses, the plain words are added to $\mathcal{B}_K$, characterizing the known class, while the negated words $\mathcal{N}_L$ are added to $\mathcal{B}_N$.

4.2 Scoring word novelty

With the word bags $\mathcal{B}_K$ and $\mathcal{B}_N$ available, we calculate novelty scores at the word level as follows. From the unique words in the bags $\mathcal{B}_K$ and $\mathcal{B}_N$, we produce two corresponding word sets, $\mathcal{S}_K$ and $\mathcal{S}_N$. Assume these respectively contain K and N unique words:

$$\begin{aligned} \mathcal{S}_K &= \{s_1, s_2, \ldots, s_k, \ldots, s_K\}, \\ \mathcal{S}_N &= \{s_1, s_2, \ldots, s_n, \ldots, s_N\}. \end{aligned} \tag{3}$$

Here, s_k represents a specific word in the set $\mathcal{S}_K$, while s_n represents a specific word in the set $\mathcal{S}_N$.

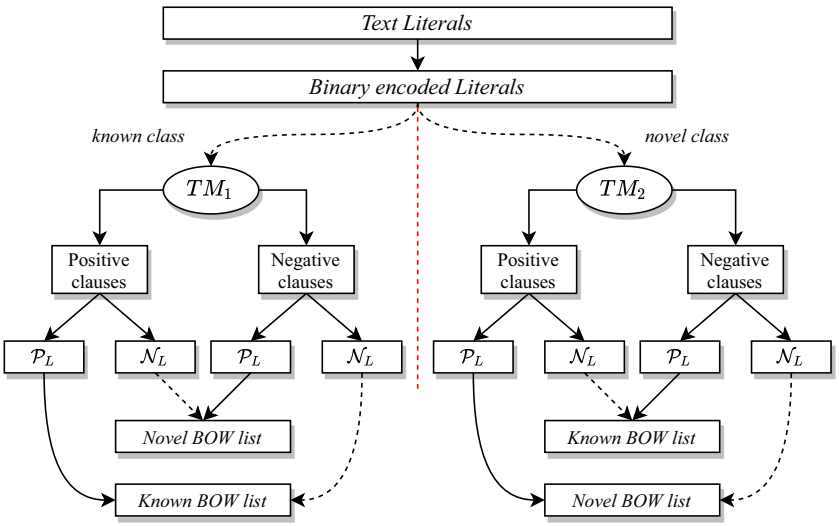


Fig. 2 Tsetlin Machine architecture for generating word sequences

We next estimate the occurrence probability p_{s_i} of each word s_i in $\mathcal{S}_K$, from the known class. The estimate is based on the relative frequency of s_i in the word bag $\mathcal{B}_K$ as given by (4):

$$p_{s_i}^{\mathcal{K}} = \frac{\mathcal{F}_i^{\mathcal{K}}}{\sum_{k=1}^{K} \mathcal{F}_k^{\mathcal{K}}}. \tag{4}$$

Here, $\mathcal{F}_i^{\mathcal{K}}$ is the frequency of word s_i in $\mathcal{B}_K$, i.e., the number of times that word s_i has the appropriate role in one of the clauses (as defined in the previous section). To prevent infinite or zero scores, we assume that every word has a minimum frequency of 1. In the following, we denote the set of relative frequencies for the words from $\mathcal{B}_K$ by $p_{\mathcal{K}}$, while $p_{\mathcal{N}}$ is the set of relative frequencies for the words from $\mathcal{B}_N$, as captured by (5):

$$\begin{aligned} p_{\mathcal{K}} &= \{p_{s_1}^{\mathcal{K}}, p_{s_2}^{\mathcal{K}}, \ldots, P_{s_K}^{\mathcal{K}}\}, \\ p_{\mathcal{N}} &= \{p_{s_1}^{\mathcal{N}}, p_{s_2}^{\mathcal{N}}, \ldots, p_{s_N}^{\mathcal{N}}\}. \end{aligned} \tag{5}$$

The calculation of the novelty score for each word depends on whether $s_i \in \mathcal{S}_K$, $s_i \in \mathcal{S}_N$, or both, as shown in (6):

$$Score(s_i) = \begin{cases} \frac{p_{s_i}^{\mathcal{N}}}{p_{s_i}^{\mathcal{K}}} & \text{if } s_k \in \mathcal{S}_K \cap \mathcal{S}_N, \\ 0 & \text{if } s_k \in \mathcal{S}_K \setminus \mathcal{S}_N, \\ \infty & \text{if } s_k \in \mathcal{S}_N \setminus \mathcal{S}_K. \end{cases} \tag{6}$$

Here, $p_{s_i}^{\mathcal{N}}$ and $p_{s_i}^{\mathcal{K}}$ denote the estimated occurrence probabilities of the word s_i from $p_{\mathcal{N}}$ and $p_{\mathcal{K}}$, respectively. The score defines how much a word contributes in a sentence/document to make it novel. That is, a higher score signals higher novelty and vice versa. Figure 3 shows the resulting TM-based architecture and flow of information for the above scoring approach.

Additionally, we also propose a contextual scoring approach to capture multiple word meanings determined by context. We presume that words that appear in the same clause are related semantically, and accordingly, we use clause co-occurrence of words to measure semantic relations. The intent is to differentiate between, for example, the meaning of "apple" in "apple phone" and the meaning of "apple" in "apple fruit". We achieve this through leveraging clauses that capture "apple" and "phone" in combination with other clauses that capture "apple" and "fruit".

The scoring is again performed in two steps:

1. Rather than measuring the frequency of individual words, we now measure frequency of co-occurrence among the TM clauses. For instance, let us consider the word pair (s_1, s_2) and novel class, associated with a total number of m clauses. The frequency of the word pair occurring together in the clauses is then given as:

$$p_{s_1,s_2}^{\mathcal{N}} = \frac{\mathcal{F}_{s_1,s_2}^{\mathcal{N}}}{m}. \tag{7}$$

   Here, $F_{s_1,s_2}^{\mathcal{N}}$ is the number of times the word pair occur together across the m clauses of the novel class.
2. Finally, the contextual score for the word pair (s_1, s_2) in class *Novel* can be defined as:

$$Score_{context}^{\mathcal{N}}(s_1, s_2) = \frac{p_{s_1,s_2}^{\mathcal{N}}}{p_{s_1}^{\mathcal{N}} \times p_{s_2}^{\mathcal{N}}}. \tag{8}$$

   Above, $p_{s_2}^{\mathcal{N}}$ and $p_{s_1}^{\mathcal{N}}$ are the individual frequencies of each word across the novel clauses, from the previous subsection.

Notice how the above score increases with lower individual frequencies and higher joint frequency, measuring

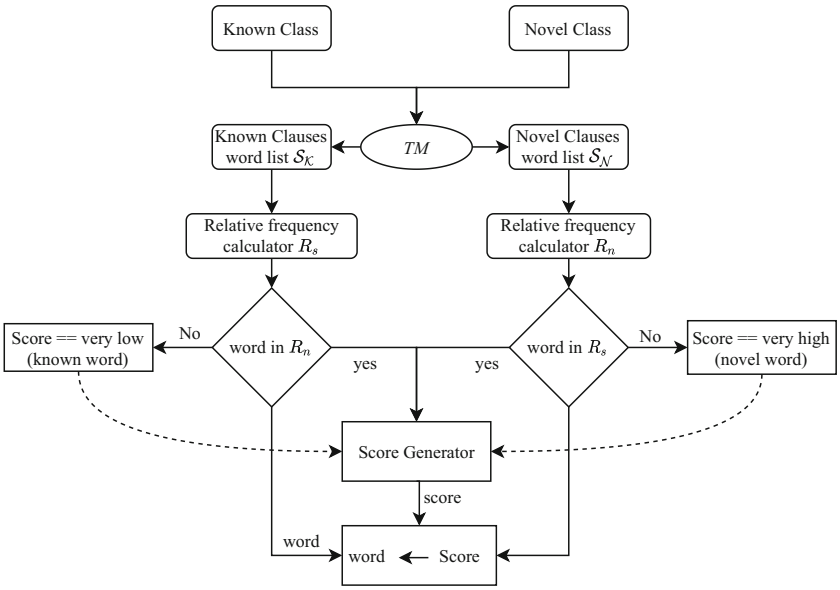


Fig. 3 Novelty scoring calculation for each word

dependence over the clauses. In the same way, we can calculate dependence over the clauses for the known class as well.

4.3 Case study

We now demonstrate our novelty description approach, steb-by-step, using two example sentences from the sports domain. For illustration purposes, we consider the class Cricket to be *Known* and the class Rugby to be *Novel*.

- **Class** : Cricket (Known)
  Text: England won the cricket match by hitting six in the last ball.
  Words: *"England"*, *"won"*, *"cricket"*, *"match"*, *"hit"*, *"six"*, *"ball"*.
- **Class**: Rugby (Novel)
  Text: England won the rugby match despite using old ball.
  Words: *"England"*, *"won"*, *"rugby"*, *"match"*, *"despite"*, *"old"*, *"ball"*.

We first create the set of 10 unique words $W = \{$ *"England"*, *"won"*, *"cricket"*, *"match"*, *"hit"*, *"six"*, *"ball"*, *"rugby"*, *"despite"*, *"old"*$\}$ from the words in the two sentences, each with a unique index o. From this set, we produce the input feature vector for the TM, $X = [x_1, x_2, \ldots, x_{10}]$. Each propositional input x_o in X refers to a particular word. Jointly, the propositional inputs are used to represent an input text. If a word $w_o \in W$ is present in the document, the corresponding propositional input x_o is set to 1, otherwise, it is set to 0.

After TM training, we obtain a set of clauses, as exemplified in Table 1. The clauses $(C_1^+)_{\mathcal{K}}$, $(C_2^+)_{\mathcal{K}}$, $(C_1^-)_{\mathcal{N}}$, $(C_2^-)_{\mathcal{N}}$ vote for class *Known*, while $(C_1^-)_{\mathcal{K}}$, $(C_2^-)_{\mathcal{K}}$, $(C_1^+)_{\mathcal{N}}$, $(C_2^+)_{\mathcal{N}}$ vote for class *Novel*. These clauses are then used to produce two bag-of-words, $\mathcal{B}^K$ and $\mathcal{B}^N$. All the plain words in $(C_1^+)_{\mathcal{K}}$, $(C_2^+)_{\mathcal{K}}$, $(C_1^-)_{\mathcal{N}}$, $(C_2^-)_{\mathcal{N}}$ are placed in $\mathcal{B}^K$, while all the negated words are placed in $\mathcal{B}^N$. Since none of the words are negated in the clauses, we now have $\mathcal{B}^K =$ (*"England"*, *"cricket"*, *"match"*, *"hit"*, *"six"*, *"cricket"*, *"six"*, *"cricket"*, *"won"*, *"six"*, *"ball"*, *"cricket"*, *"hit"*, *"six"*). Correspondingly, all the plain words in $(C_1^-)_{\mathcal{K}}$, $(C_2^-)_{\mathcal{K}}$, $(C_1^+)_{\mathcal{N}}$, $(C_2^+)_{\mathcal{N}}$ are placed in $\mathcal{B}^N$, while all the negated words are placed in $\mathcal{B}^K$.

Within each bag-of-words, each word occurs with a certain frequency. For instance, the word *"match"* occurs once in $\mathcal{B}^K$ and twice in $\mathcal{B}^N$. Notice that the total number of word occurrences are different for each class – 14 words in class *Known* and 13 words in class *Novel*. Hence, the relative frequency for *"match"* in class *Known* becomes $p^{\mathcal{K}}_{match} = \frac{1}{14} = 0.071$ while for class *Novel* it becomes $p^{\mathcal{N}}_{match} = \frac{2}{13} = 0.154$. Table 2 lists the frequencies of the words per class.

We are now ready to calculate the novelty score for each word in W. Let us consider the word "rugby" from the novel word set and the word "cricket" from the known word set. For "rugby", we first calculate its relative frequency (4). In the bag-of-word $\mathcal{B}_N$ for class *Novel*, "rugby" occurs *four* times, i.e., $\mathcal{F}^{\mathcal{N}}_{rugby} = 4$. Since we assume that a word has a minimum frequency of 1, we further have $\mathcal{F}^{\mathcal{K}}_{rugby} = 1$, despite "rugby" not appearing in the text from class *Known*.

From Table 2, we observe that the total word frequencies for the known and novel classes are 14 and 13, respectively. Hence, the relative frequencies for "rugby" becomes $p_{rugby}(\mathcal{K}) = 0.307$ for class *Known* and $p_{rugby}(\mathcal{N}) = 0.071$ for class *Novel* (4).

Because the clauses characterize each class *Known* and *Novel*, notice how "rugby" gets the relatively high novelty score $Score_{rugby} = 4.651$. That is, its relative frequency is high in the novel class and low in the known class. Conversely, the word "cricket" is repeated *four* times in $\mathcal{B}^K$ and *once* in $\mathcal{B}^N$. Its relative frequencies thus becomes $p_{cricket}(\mathcal{K}) = 0.28$ for class *Known* and $p_{cricket}(\mathcal{N}) = 0.076$ for class *Novel*. Accordingly, the novelty score becomes $Score_{cricket} = 0.271$, which is a low score denoting a strong inclination of the word towards the known class.

Overall, Table 2 shows how the words characterizing class *Known* get a relatively low novelty score, while those characterizing class *Novel* obtain high scores.

5 Results and discussions

In this section, we evaluate our proposed novelty description approach on two publicly available datasets: *BBC Sports* and *Twenty Newsgroups*. The performance of the TM framework for novelty detection was previously investigated

Table 1 Clauses with conjunctive word patterns for known and novel class

Known Clauses	Novel Clauses
$(C_1^+)_{\mathcal{K}} =$ *"England"* $\wedge$ *"cricket"* $\wedge$ *"match"* $\wedge$ *"hit"* $\wedge$ *"six"*	$(C_1^+)_{\mathcal{N}} =$ *"England"* $\wedge$ *"won"* $\wedge$ *"rugby"* $\wedge$ *"old"*
$(C_1^-)_{\mathcal{K}} =$ *"won"* $\wedge$ *"rugby"* $\wedge$ *"ball"*	$(C_1^-)_{\mathcal{N}} =$ *"cricket"* $\wedge$ *"won"* $\wedge$ *"six"* $\wedge$ *"ball"*
$(C_2^+)_{\mathcal{K}} =$ *"cricket"* $\wedge$ *"six"*	$(C_2^+)_{\mathcal{N}} =$ *"rugby"* $\wedge$ *"match"* $\wedge$ *"despite"* $\wedge$ *"old"*
$(C_2^-)_{\mathcal{K}} =$ *"rugby"* $\wedge$ *"match"*	$(C_2^-)_{\mathcal{N}} =$ *"cricket"* $\wedge$ *"hit"* $\wedge$ *"six"*

Table 2 Relative frequency and score for each word

Known				Novel			
Word	Frequency	Relative frequency	Score	Word	Frequency	Relative frequency	Score
England	1	0.071	1.070	England	1	0.076	1.070
Won	1	0.071	2.169	Won	2	0.154	2.169
Cricket	4	0.28	0.271	Rugby	4	0.307	4.651
Match	1	0.071	2.169	Match	2	0.154	2.169
Hit	2	0.142	0.535	Despite	1	0.076	1.15
Six	4	0.28	0.271	Old	2	0.153	2.31
Ball	1	0.071	1.070	Ball	1	0.076	1.070

in [8] and is summarized in Table 3. Notably, as has been found across several datasets, a one-class SVM on the simple mean embeddings established a strong baseline. Here, we further explore our model's effectiveness at producing discriminative novelty scores at the word level using TM clauses. To obtain robust performance and ensure that the results are not influenced by the data, we perform a one-class classification using leave-one-out evaluation on 20 Newsgroup dataset. This paper deals with the post-processing after novelty detection to deal with the novelty scoring at the word level. However, the leave-one-out evaluation is necessary because this study leverage the performance of the TM framework in terms of novelty detection. We employ the ROC AUC to quantify the novelty detection performance by using the ground truth labels during testing. Table 4 shows the performance comparison of our method and the baseline algorithms, including a one-class classifier. In the leave-one-out setup, one of the classes is considered a known class, while the remaining classes are treated as novel. The training is conducted using a known class, whereas testing is carried out on samples from a novel class. The ROC AUC is computed during testing with the assumption that the samples from the known class are labeled as $y = 0$ and from novel class as $y = 1$. Our method outperforms baselines algorithms in five out of six evaluation setups with a significant margin.

In the following, we compare the scoring mechanism of our framework with attention and TF-IDF as a baseline. To ensure a fair comparison, the attention score for each word is calculated as described in Section 5.1.1. For TF-IDF, We calculate TF separately for the known and novel classes. Conversely, IDF is calculated using all the documents from both classes (to suppress common words such as stop words). Unlike attention and TF-IDF, even if a word is present in most documents, our scoring considers both relevance and context. For example, if a word from class *Novel* also is present in class *Known*, our model can nevertheless assign more weight to that word. This happens when a word, while *syntactically* the same in both classes, acquires a novel meaning in the novel class due to its appearance in a novel context. The latter contextual information is captured through those clauses of the novel class that trigger for that word. As such, attention and TF-IDF are not context-aware. Moreover, these methods prove especially beneficial on more extensive datasets, such as 20 Newsgroups and BBC Sports, since they filter out general language contexts that are less discriminative for the characterization of a text corpus, making them a strong baseline for performance comparison.

To provide a comparison, we plot the cumulative frequency distribution (CFD) for the scores of (1) the words only found in the novel dataset, (2) the words only found in the known dataset, and (3) the words shared by both datasets. In brief, the CFD demonstrates that the word scores generated by the baseline are relatively similar for both known and novel classes. Thus, the baseline methods lack the discriminatory power necessary to distinguish between the two categories of words.

Table 3 Performance comparison of TM framework with cluster and outlier-based novelty detection algorithms

Algorithms	20 Newsgroup	BBC sports
LOF	52.51 %	47.97 %
Feature Bagging	67.60 %	54.38 %
HBOS	55.03 %	49.53 %
Isolation Forest	52.01 %	49.35%
Average KNN	76.35 %	55.54 %
K-Means clustering	81.00 %	47.70 %
One-class SVM	83.70 %	83.53 %
SO-GAAL	80.2%	83.50%
MO-GAAL	82.9%	86.68%
COPOD	**84.4%**	86.09%
TM framework	82.50%	**89.47%**

Table 4 ROC AUC (%) of one-class classification with leave-one-out evaluation on 20 Newsgroup

Normal class	ABOD	CBLOF	HBOS	IForest	KNN	LOF	OCSVM	COPOD	TM
comp	0.506	0.618	0.625	0.62	0.622	0.62	0.614	**0.627**	0.55
rec	0.508	0.481	0.476	0.483	0.479	0.48	0.48	0.476	**0.60**
sci	0.50	0.435	0.449	0.454	0.434	0.435	0.433	0.45	**0.53**
misc	0.511	0.533	0.527	0.534	0.54	0.542	0.534	0.532	**0.69**
pol	0.492	0.452	0.436	0.445	0.451	0.451	0.454	0.435	**0.71**
rel	0.494	0.449	0.437	0.456	0.472	0.47	0.457	0.443	**0.63**

5.1 Baseline

5.1.1 Attention mechanism

We utilize the weights from attention's layers input representation $\mathcal{A}$ of the trained model. The importance of each token is calculated based on the attention it receives. For instance, if attention to the token $c \in \mathcal{A}$ is higher than the token $d \in \mathcal{A}$, then c is assumed to be "more significant" than d to the model's output. In our work, the scores are calculated using scaled-dot product attention mechanism [57].

Let us consider an input sequence of length o, $X = (x_1, x_2, \ldots, x_o)$, where x_i represents the i^{th} token whose representation in the attention layer is $h_i \in \mathbb{R}^t$. The attention score for the i^{th} token is as follows:

$$\alpha_i = \frac{h_i \times V}{\beta}, \tag{9}$$

where the parameter β is the scaling factor, and $V \in \mathbb{R}^t$ is the context vector that can be seen as a fixed query requesting the "most important token" from input. Either the word embedding or the encoder's output can denote token representation h_i. The attention weight can be expressed as:

$$a_i = \frac{exp(a_i)}{\sum_{i'} exp(a_{i'})}. \tag{10}$$

Finally, the complete input sequence is denoted as:

$$h = \sum_i (a_i h_i). \tag{11}$$

In our experiment, we retrieve the attention score and weights for each token using (9) and (10) respectively.

We conducted experiments using scaled dot-product attention (DP) and additive attention with varying scaling factors (β). The attention scores in our experiments are generated using a Long short-term memory (LSTM) with DP and an affine transformation layer as the input encoder. We used the Adagrad optimizer [19] for gradient descent and used dropout as regularization to prevent over-fitting. To eliminate the influence of prior knowledge, we learn all parameters from scratch and initialize the pre-trained word embeddings with a uniform distribution and dimension $d = 100$. A softmax function is applied over a linear layer for obtaining the final classification output. The readers are referred to [53] for a detailed theoretical explanation to generate the attention scores.

5.1.2 Term frequency-inverse document frequency (TF-IDF)

A commonly used method to analyze the importance of a word is the term frequency-inverse document frequency (TF-IDF) [45]. TF-IDF weighs each word to statistically measure the significance of the word in a given document. To this end, TF-IDF consists of two factors: normalized term frequency (TF) and inverse document frequency (IDF). TF measures the frequency of the word in the document, whereas IDF measures the uniqueness of the word across documents:

$$TF - IDF_s = \frac{\mathcal{F}_s}{\mathcal{F}} \times \log_2 \frac{|D|}{|D_s| + 1}. \tag{12}$$

Here, $\mathcal{F}_s$ is the frequency of the word s in the target document, $\mathcal{F}$ is the sum of the target document word frequencies, $|D|$ is the total number of documents, and $|D_s|$ is the number of documents containing the word s.

5.1.3 Keyword extraction algorithms

Our method extracts keywords from known and novel classes based on the novelty scores. As a result, we also compare the significant words obtained by our method to those captured by existing keyword extraction (KE) algorithms. To do this, we first separate the text documents from known and novel classes before passing them to the KE algorithms. Additionally, we present the top 10 keywords captured by these algorithms. For the KE baselines mentioned below, we use the *pke* package [10]:

- TopicRank [12]: This is a graph-based KE method that depends on the extraction of the top-ranked topic.
- YAKE [13]: A lightweight statistical approach for KE.
- MultipartiteRank [11]: An unsupervised KE method for encoding topical information in a multipartite graph structure.

– BERT-MMR [26]: A KE method that leverages Bidirectional Encoder Representations from Transformers (BERT) embeddings and Maximal Marginal Relevance (MMR).

5.2 Evaluation measures

We use the accuracy, Receiver Operating Characteristics (ROC) curve, precision, and recall to evaluate the performance of novelty detection using word scores obtained from the proposed method. In general, accuracy is a well-known parameter to measure the effectiveness of novelty detection models, which indicates the percentage of correct prediction by a model in a test set. The accuracy is calculated by:

$$Accuracy = \frac{T_P + T_N}{T_P + T_N + F_P + F_N}, \tag{13}$$

where T_P, T_N, F_P, F_N denotes the samples that are correct novel, correct normal, incorrect novel, and incorrect normal respectively. And P, N denotes the total novel and normal samples. The precision is defined by the percentage of correctly identified novel samples and is give as:

$$Precision = \frac{T_P}{T_P + F_P}. \tag{14}$$

Recall is the percentage of the real novel samples identified and is given as:

$$Recall = \frac{T_P}{T_P + F_N}. \tag{15}$$

In general, the higher the precision and the recall, the better the algorithm. However, the precision and recall are mutually constrained. For example, if only one novel sample is detected, the precision is 100%, while the recall is very low. And if all samples detected are novel, the recall will be 100%, while precision tends to be very low. Therefore, we present precision-recall graph in our evaluation.

The ROC is insensitive to the number of novel samples and is calculated by plotting all potential choices of the T_P rate (the portion of novel data ranked among the total novel data) against the F_P rate (the portion of normal data ranked among the total novel data). The ROC curve can be summarized using a single value defined as the area under ROC curve (AUC). The ROC value ranges between 0 and 1 and is regarded as average of the recall. The perfect detection of all test samples would result in ROC value of 1, whereas the null detection would result in ROC value of 0. In general, the greater the ROC AUC value, the better the algorithm. [27] established that the ROC AUC value corresponds to the probability of a pair $(nov,\ nor)$, where nov is certain true novel samples and nor is certain true normal samples. The ROC AUC can then be defined by:

$$ROC\ AUC = \begin{cases} 1, & \textbf{if } score(nov) > score(nor), \\ 0, & \textbf{if } score(nov) < score(nor), \\ 1/2, & \textbf{if } score(nov) = score(nor). \end{cases}$$

Therefore, the ROC AUC has a direct probabilistic interpretation. The AUC can be also be defined as:

$$AUC = \int_0^1 ROC(T)\, d_T, \tag{16}$$

where T denotes a threshold to control novel samples. The ROC AUC is the most often used evaluation metric for novelty detection that provides a ranking. Therefore, in this paper, we compare the evaluation alongside other methods, so it can give different aspects of the performance. To ensure fairness, effectiveness and reproducibility of the evaluation results, we use scikit-plot library[1] to compute ROC and precision-recall graphs.

5.3 BBC sports dataset

The BBC sports dataset comprises 737 documents from the BBC sport website organized in five sports article categories, collected from 2004 to 2005. The resulting vocabulary encompasses 4 613 terms. For our experiment, we consider the classes "cricket" and "football" to be known and the class "rugby" to be novel, thus creating an unbalanced dataset. For preprocessing, we perform tokenization, stopword removal, and lemmatization. We run the TM for 100 epochs with 10 000 clauses, a voting margin T of 50, and a sensitivity s of 25.0.

We present overall novelty score statistics for the words captured by the clauses in Table 5. The table demonstrates that words in the class *Novel* have distinctively higher average scores than words in the class *Known*. Also, notice that the shared words have the highest mean and standard deviation. As analyzed further below, this is the case because the TM will mainly use those words when forming the decision boundary between the two classes. As a result, the shared words will appear in more clauses as characterizing class features. That is, the clauses will either single out the words in one class or suppress the words in the other class.

To gain further insight into the properties of the novelty score, we plot the CFD for the scores of the novel, known, and shared words in Fig. 4. We further compare these CFDs with the corresponding ones obtained using attention weights in Fig. 5 and TF-IDF in Fig. 10. As can be observed

[1] https://scikit-plot.readthedocs.io/en/stable/Quickstart.html.

Table 5 Overall word statistics for BBC sport dataset

Category	Total word count	Average score	Standard deviation
Known words	6660	0.74	0.23
Novel words	1941	1.3125	3.75
Shared words	3135	11.30	316.93

from the plot, our approach produces more distinctive novelty scores than both attention and TF-IDF. The novel words typically produce high scores, while the known words produce low scores. In particular, as shown in Fig. 4a, 85% of the known words output scores lower than 1.0. On the other hand, as seen in Fig. 4b, only approximately 45% of the words unique for the novel class have scores below 1. The majority of the uniquely novel words produce scores greater than 1.

We plot the TM and attention scores for each token in Fig. 6b and a, respectively. Due to the large span of the TM scores, the y-axis is plotted on a log scale. Nonetheless, we note that the scores are structured in successive layers, with known scores at the bottom, novel scores at the top, and shared scores in the center. We notice that even the attention score demonstrates a small degree of differentiation between known and novel categories. However, the variability of the score is quite low when compared to the score generated by TM as seen in Fig. 7 boxplot. Additionally, the shared word scores produced by the attention mechanism exhibit a high degree of resemblance to known word scores.

Finally, we plot the scores for words that are shared between the known and novel classes in Fig. 4c. As can be observed, the words that are shared produce both high and low scores. To cast further light on this observation, we investigate the words that are shared further in Table 6. We see that the words captured frequently by novel clauses have high scores, whereas the words that are frequent in known clauses have low scores. Additionally, common words (e.g., stopwords), also have low scores. For example, the word "Rugby", which is highly characteristic for class *Novel*, is repeated only 5 times in the clauses representing class *Known*. For the clauses that represent class *Novel*, on the other hand, it is repeated 215 times. In other words, the

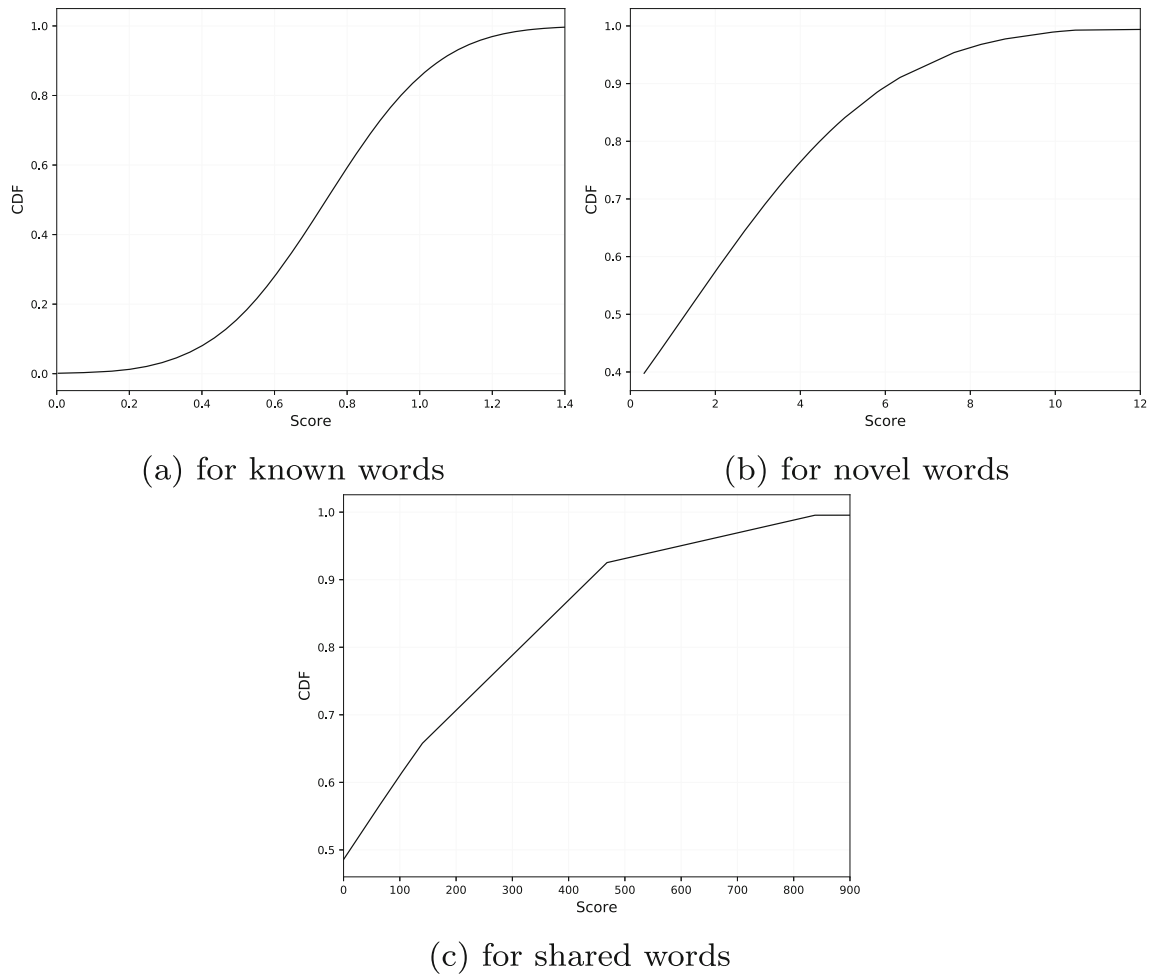


Fig. 4 Cumulative frequency distribution (CFD) graph for word scores in different categories of BBC Sports using TM

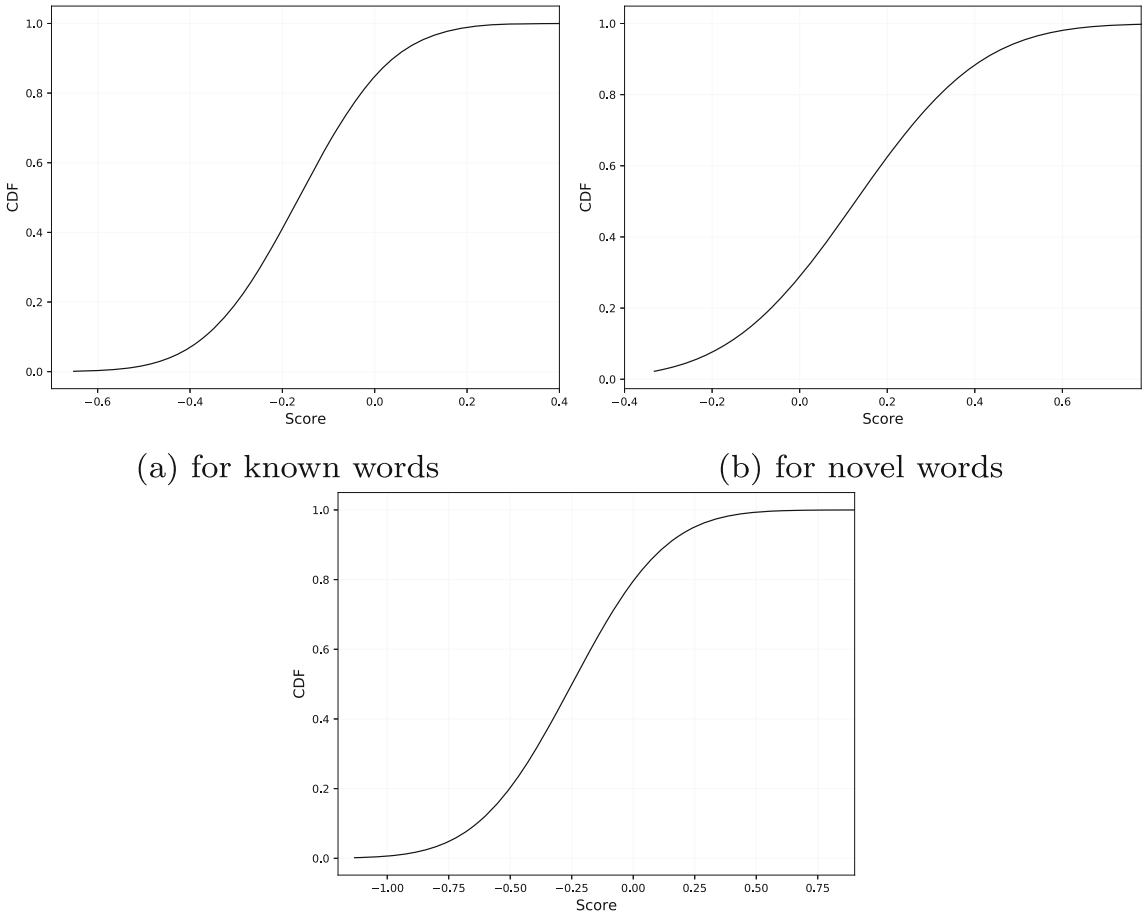


Fig. 5 Cumulative frequency distribution (CFD) graph for word scores in different categories of BBC Sports using attention weights

shared words constitute words that are either characteristic for class *Known* or class *Novel*. This finding also suggests that the scores can be calculated accurately even if the words are present in both categories. We analyze the most frequently used words to obtain an intuition of the overall theme captured by the clauses. We generate such lists by counting the top words according to the highest scores from known and novel classes. Such a list may assist a

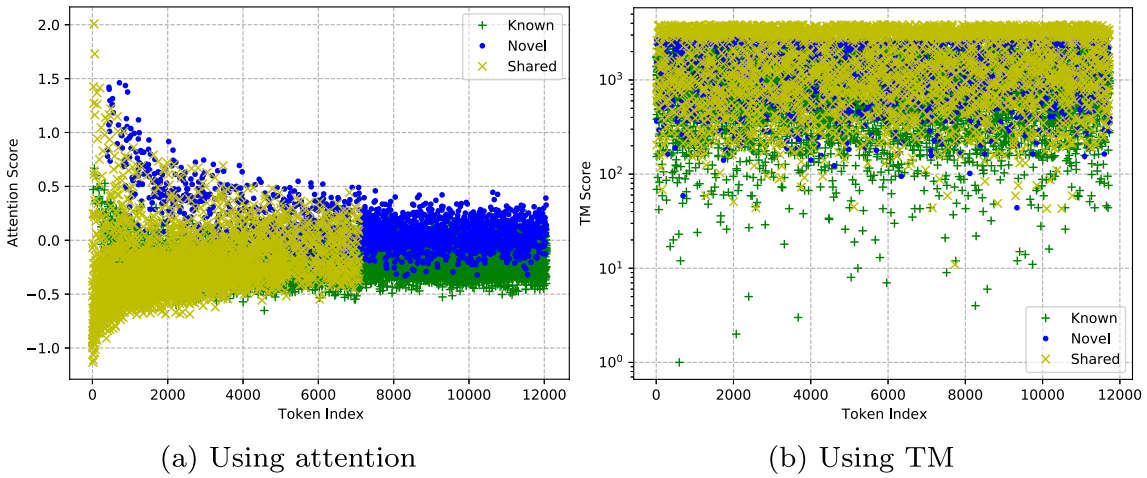
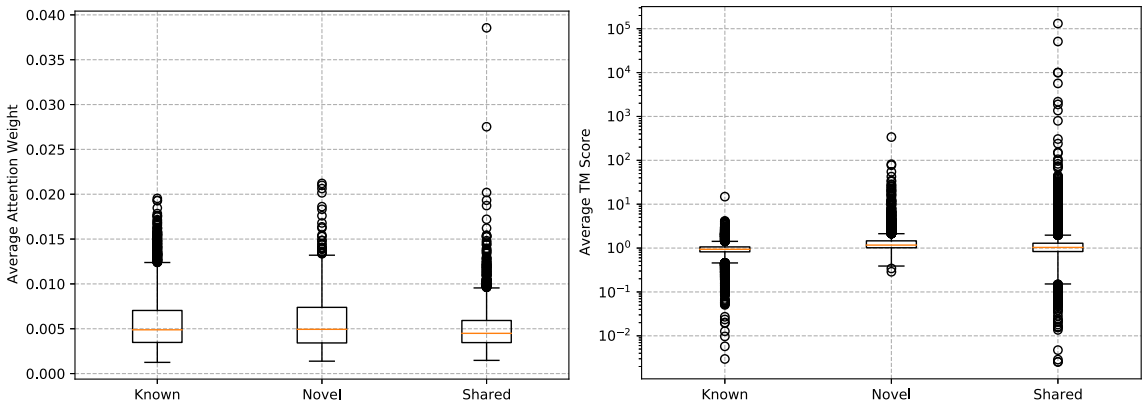


Fig. 6 Visualization of tokens in known, Novel and Shared categories from BBC Sports



(a) Using attention (b) Using TM

Fig. 7 Boxplot of scores in known, Novel and Shared categories from BBC Sports

Table 6 Composition of shared words in BBC Sport

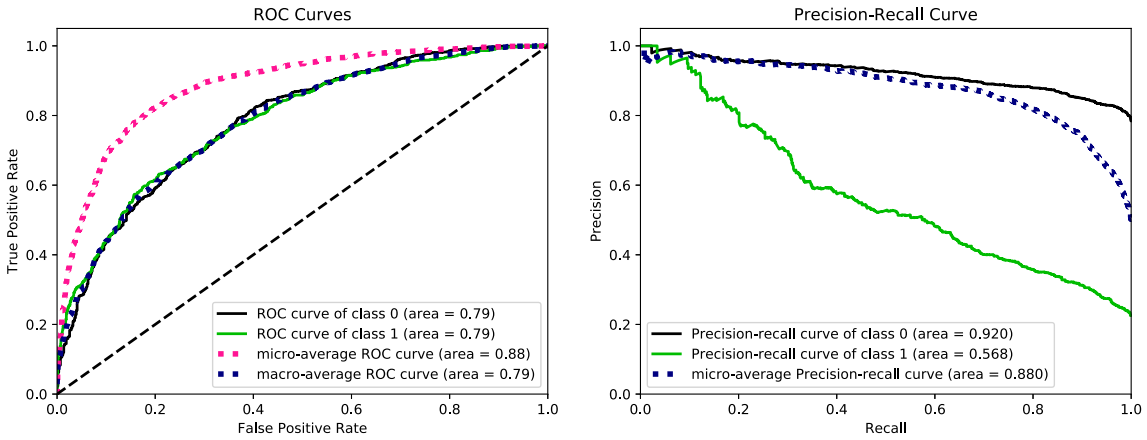
Composition	Total word count	Average score	Standard deviation
Known words	10	0.11	0.070
Novel words	17	1941.13	3919.02
Common words	3051	1.03	0.99

Table 7 Example of top words extracted from KE baselines for the Known class in BBC Sports

TM		TopicRank		YAKE		MultipartiteRank		BERT-MMR	
Word	Score	Word	Score	Word	Score	Word	Score	Word	Score
manchester	0.004	players	0.0065	said	2.65	players	0.0053	kickoff	0.25
manager	0.040	ball	0.0050	game	5.16	games	0.0050	fifaasian	0.29
arsenal	0.041	team	0.0050	england	6.48	goal	0.0038	espnstar	0.28
united	0.043	goal	0.0043	chelsea	7.01	chelsea	0.0029	fifa	0.24
cricket	0.046	chelsea	0.0038	players	7.08	team	0.0029	juventus	0.23
chelsea	0.049	wickets	0.0030	team	7.44	manchester	0.0026	matchwinner	0.22
oneday	0.051	oneday	0.0029	oneday	8.35	arsenal	0.0024	sportsweek	0.34
striker	0.074	england	0.0027	united	8.39	ball	0.0023	goalkick	0.20
batsman	0.114	arsenal	0.0027	league	8.39	england	0.0022	clubmate	0.12
bowler	0.133	tests	0.0026	arsenal	9.62	matches	0.0021	autobiography	0.13

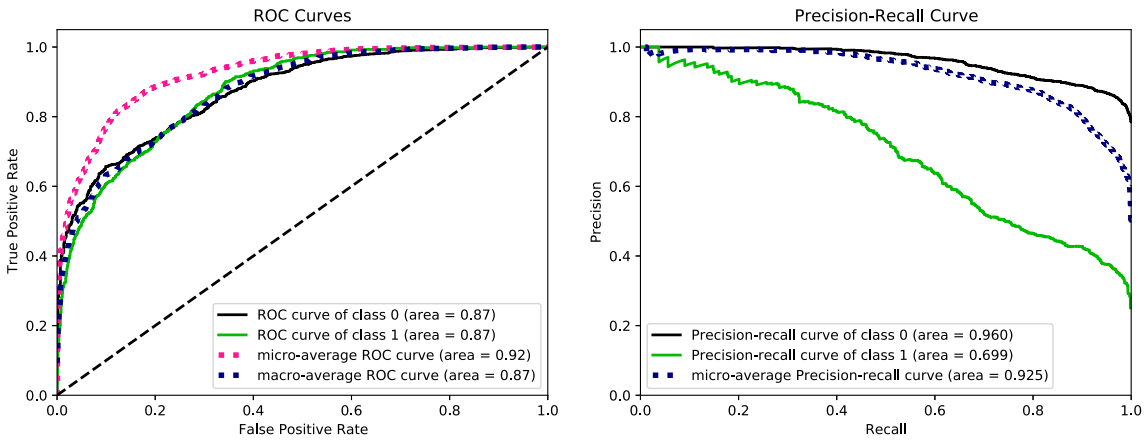
Table 8 Example of top words extracted from KE baselines for the novel class in BBC Sports

TM		TopicRank		YAKE		MultipartiteRank		BERT-MMR	
Word	Score	Word	Score	Word	Score	Word	Score	Word	Score
rugby	11143.58	nations	0.012	england	0.006	nations	0.0089	nflstyle	0.29
nations	10000	game	0.011	rugby	0.101	england	0.0082	july	0.05
ireland	468.18	player	0.009	wales	0.112	game	0.0074	wednesday	0.08
flyhalf	54.79	side	0.007	nations	0.113	wales	0.0071	rugby	0.17
lions	38.18	wales	0.007	ireland	0.113	player	0.0064	wordclass	0.06
scrumhalf	25.09	ireland	0.006	game	0.125	side	0.0048	dropkicking	0.12
flanker	19.61	years	0.006	coach	0.146	team	0.0038	goalkickers	0.21
irish	17.55	team	0.005	side	0.182	france	0.0036	sportsweek	0.28
centre	17.19	ball	0.004	players	0.206	win	0.0033	tournament	0.13
squad	9.28	win	0.003	win	0.220	squad	0.0030	kickoff	0.23



(a) ROC curve (b) precision-recall graph

Fig. 8 ROC curve and precision-recall of known/novel class classification of BBC Sports using word scores obtained from TM



(a) ROC curve (b) precision-recall graph

Fig. 9 ROC curve and precision-recall of known/novel class classification of BBC Sports using attention scores

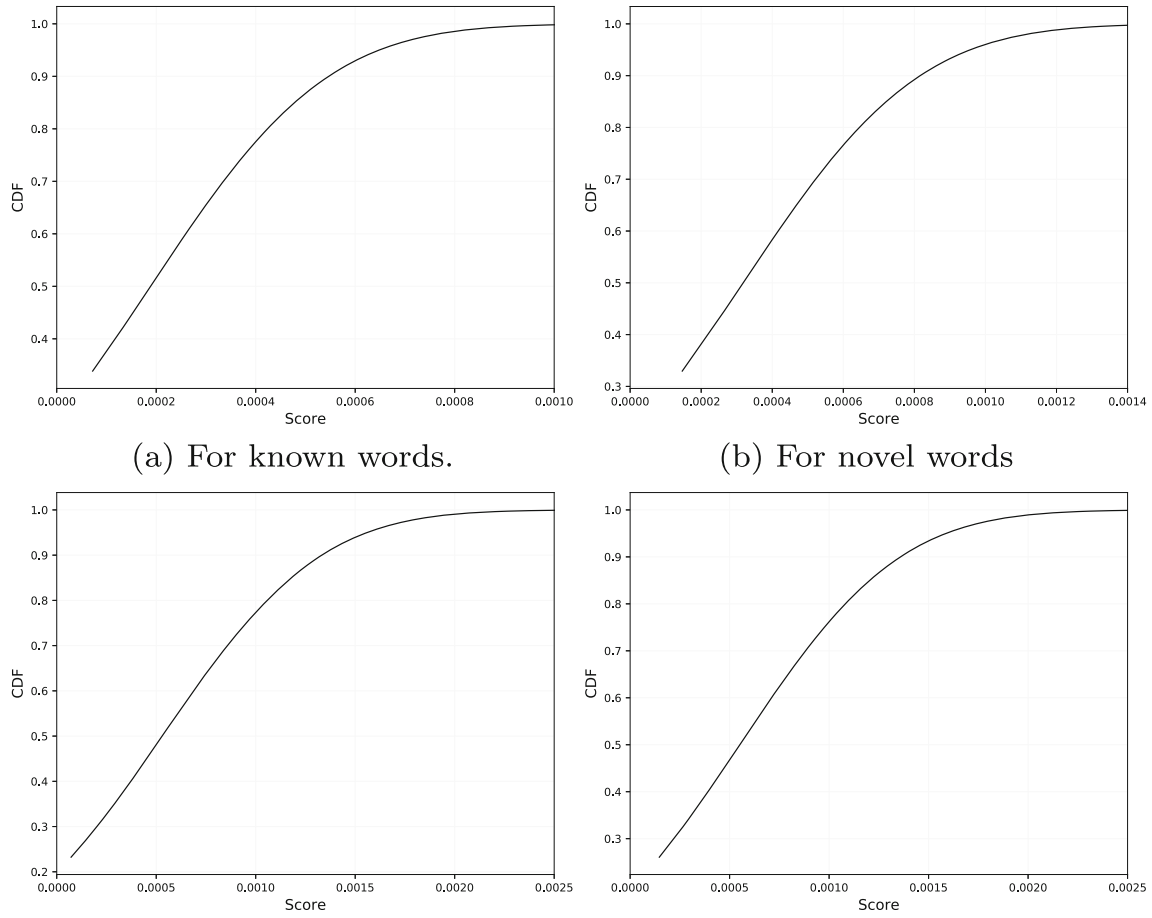


Fig. 10 Cumulative frequency distribution (CFD) graph for TF-IDF scores in different categories of BBC Sports

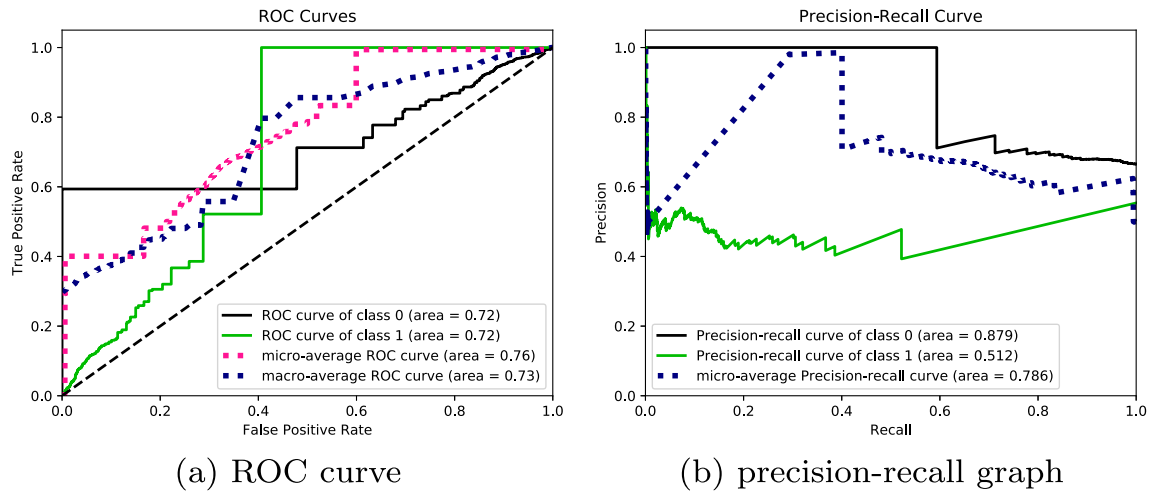


Fig. 11 ROC curve and precision-recall of known/novel class classification of BBC Sports using TF-IDF scores

Table 9 Overall word statistics for 20 Newsgroups dataset

Category	Total word count	Average score	Standard deviation
Known words	23133	0.99	0.21
Novel words	6921	1.20	1.04
Shared words	5786	3.04	131.62

Table 10 Composition of shared words in 20 Newsgroups dataset

Composition	Total word count	Average score	Standard deviation
Known words	9	0.14	0.074
Novel words	33	640.75	2378.87
Common words	5697	1.11	0.58

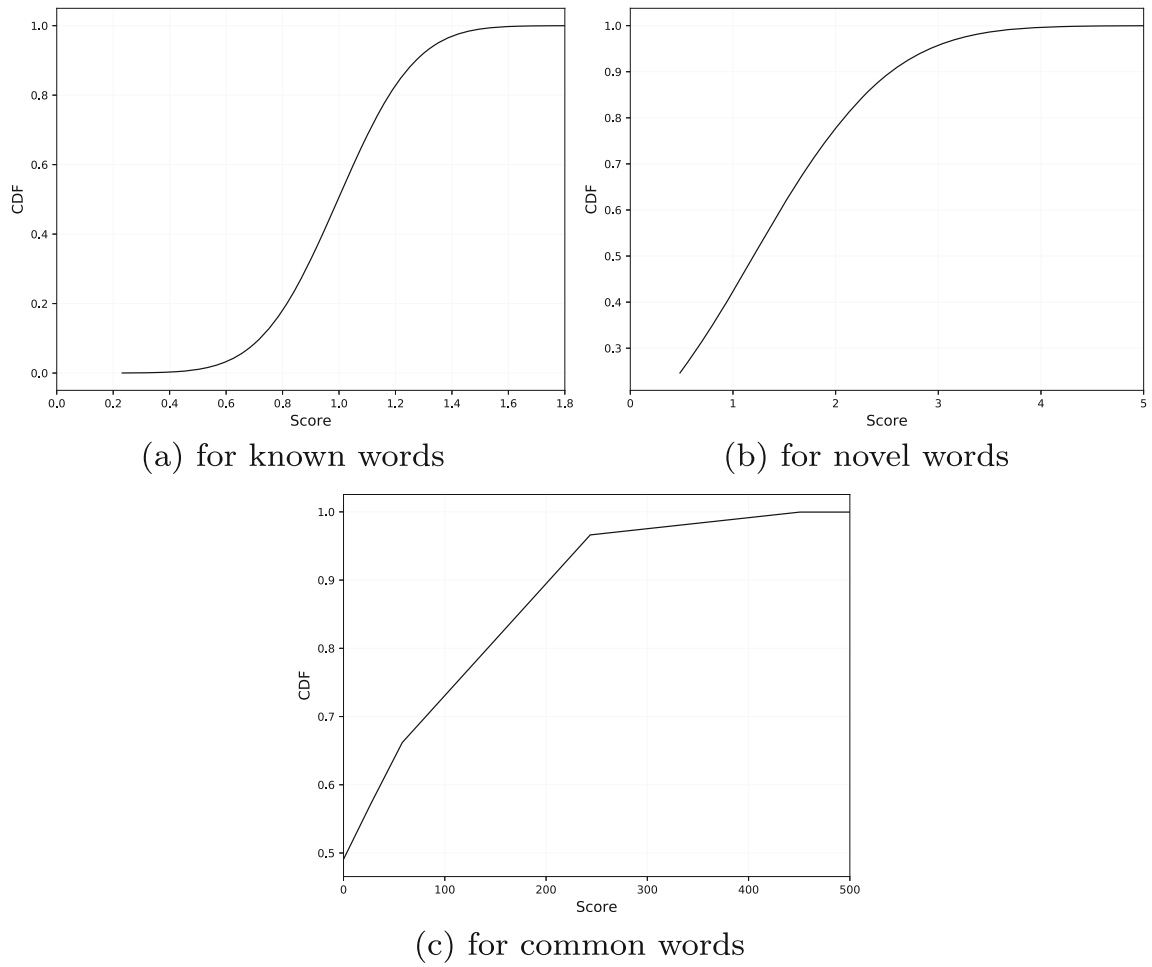


Fig. 12 Cumulative frequency distribution (CFD) graph for word scores in different categories of 20 Newsgroups using TM

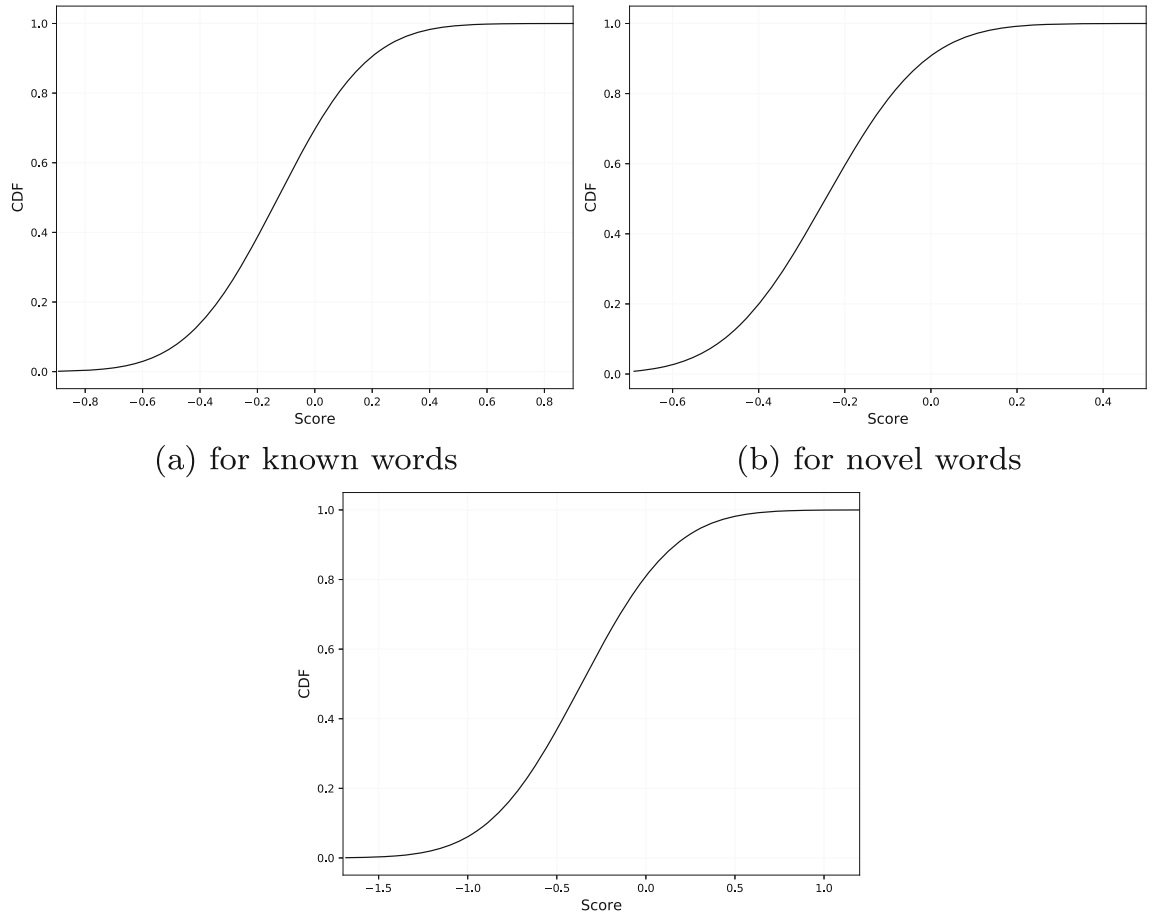


Fig. 13 Cumulative frequency distribution (CFD) graph for word scores in different categories of 20 Newsgroups using attention weights

user in weighting and selecting relevant words in a specific application.

Tables 7 and 8 show an example of a top word list for each class, from which we make the following observations. First, our proposed method assigns low scores to words belonging to known classes while assigning comparatively high scores to words belonging to the novel class. In general, the words that appear in a novel context are

Table 11 Example of top words extracted from KE baselines for the known class in 20 Newsgroups

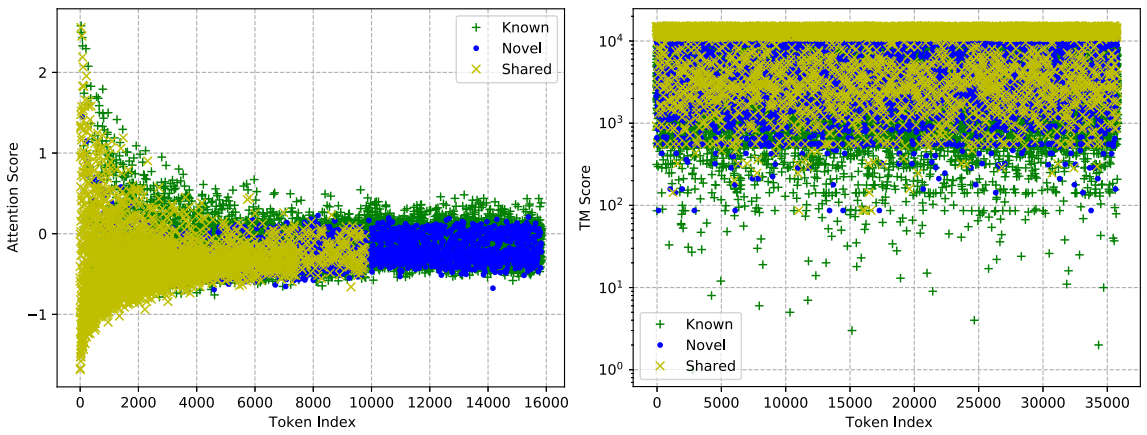
TM		TopicRank		YAKE		MultipartiteRank		BERT-MMR	
Word	Score	Word	Score	Word	Score	Word	Score	Word	Score
program	0.103	people	0.074	image	4.60	people	0.054	lawbook	0.44
gun	0.104	gun	0.041	gun	6.77	guns	0.033	interpolation	0.42
use	0.117	article	0.038	people	7.19	article	0.027	literature	0.41
write	0.145	fire	0.032	file	7.25	government	0.026	reading	0.41
public	0.154	government	0.032	article	7.41	fire	0.022	writing	0.41
file	0.175	image	0.027	like	10.1	time	0.019	translation	0.41
image	0.187	fbi	0.021	jpeg	13.1	day	0.018	lexidata	0.40
email	0.074	weapons	0.019	think	15.6	weapons	0.017	prisoncamp	0.17
police	0.358	problem	0.018	time	16.6	fbi	0.016	comparable	0.40
fbi	0.231	information	0.017	program	17.4	year	0.014	literate	0.40

Table 12 Example of top words extracted from KE baselines for the novel class in 20 Newsgroups

TM		TopicRank		YAKE		MultipartiteRank		BERT-MMR	
Word	Score	Word	Score	Word	Score	Word	Score	Word	Score
team	450.66	game	0.082	writes	1.48	game	0.065	baseball	0.38
baseball	243.62	team	0.074	game	1.55	team	0.048	reporters	0.33
pitcher	62.26	player	0.055	article	2.29	pitch	0.037	salaries	0.32
league	55.88	year	0.053	team	2.30	player	0.037	basketball	0.31
season	40.50	baseball	0.049	last	2.95	baseball	0.034	fittest	-0.07
hit	24.63	pitcher	0.042	baseball	3.41	pitcher	0.026	kickoff	0.15
play	22.68	ball	0.037	player	3.44	ball	0.026	nba	0.10
batting	22.25	runs	0.030	time	3.94	runs	0.022	secretive	0.30
pitching	22.00	season	0.030	hit	4.06	season	0.020	jerseys	0.30
player	21.86	braves	0.030	run	4.66	braves	0.019	catchers	0.30

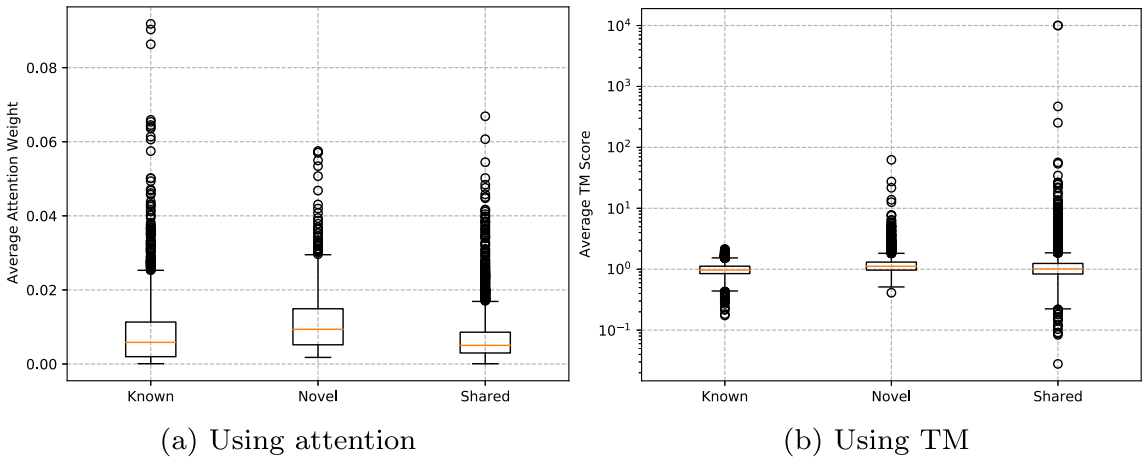
boosted. Second, the words that are most representative for the respective classes are captured frequently by clauses, making them the most repeated ones. Third, the keywords captured by other KE baselines are comparable to those extracted by our method and accurately define the corresponding classes. We observe that TopicRank, YAKE, and MultipartiteRank all yield words with a high degree of similarity to our approach. Additionally, we notice that BERT-MMR exhibits the worst performance. This might be due to the fact that we utilized pre-trained sentence embedding for BERT, and the keywords are extracted from overall documents. Even though the words are not highly relevant to the classes, BERT is capable of producing words relating to the class's general theme. For example, sports-related words are included in both classes.

We now investigate the degree of discrimination power our novelty scoring provides, and therefore uniquely describes novelty at the word level. To this end, we employ logistic regression for classifying novel text based on the word scores obtained from our method. The ROC and precision-recall curves of the experiment are depicted in Fig. 8 for our novelty scoring mechanism. Our method provides the competitive ROC value due to its ability to discriminate novel samples based on their scores. This capability enables our method to acquire a higher true positive T_P rate since it makes separate analysis of both correct novel, i.e., true positive T_P and correct normal, i.e., true negative T_N. Figures 9a, 10, and 11 contains corresponding curves when TF-IDF and attention scores are used instead. We see that the classification performance for our novelty scores is substantially better than what is obtained with TF-IDF. Attention score outperforms our approach for BBC Sports dataset by a small percentage. However, our approach outperforms attention in 20 Newsgroups. This can be attributed to the capability of our approach to deal with a big dataset.



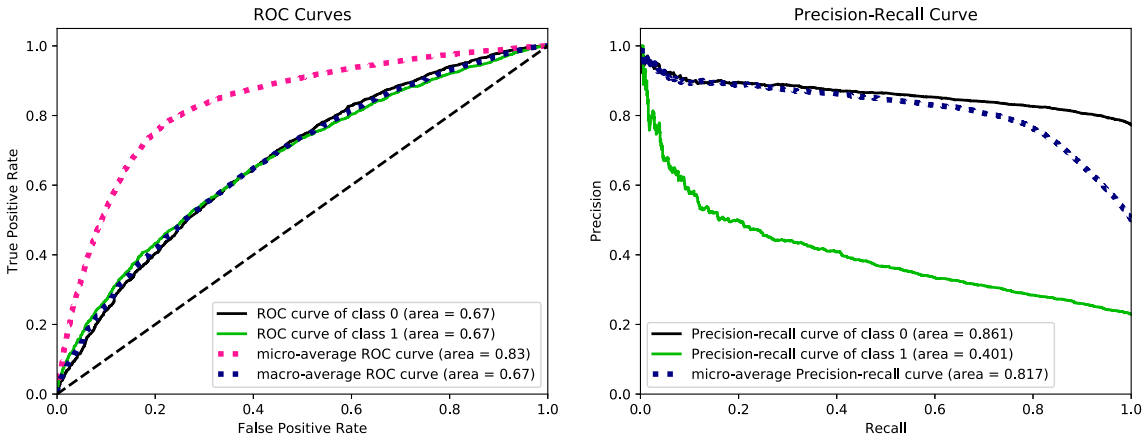
(a) Using attention (b) Using TM

Fig. 14 Visualization of tokens in known, Novel and Shared categories from 20Newsgroups



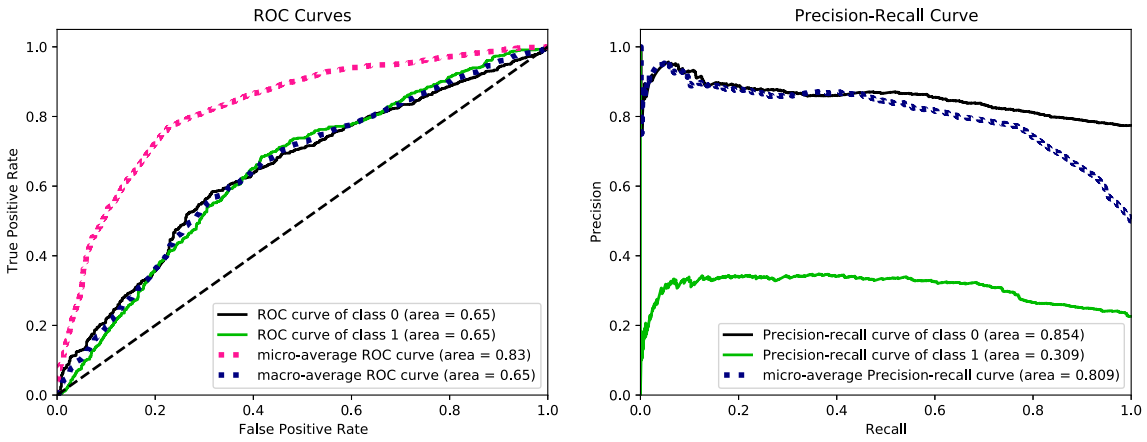
(a) Using attention (b) Using TM

Fig. 15 Boxplot of scores in known, Novel and Shared categories from 20Newsgroups



(a) ROC curve (b) precision-recall graph

Fig. 16 ROC curve and precision-recall of known/novel class classification of 20 Newsgroups using word scores obtained from TM



(a) ROC curve (b) precision-recall graph

Fig. 17 ROC curve and precision-recall of known/novel class classification of 20 Newsgroups using attention scores

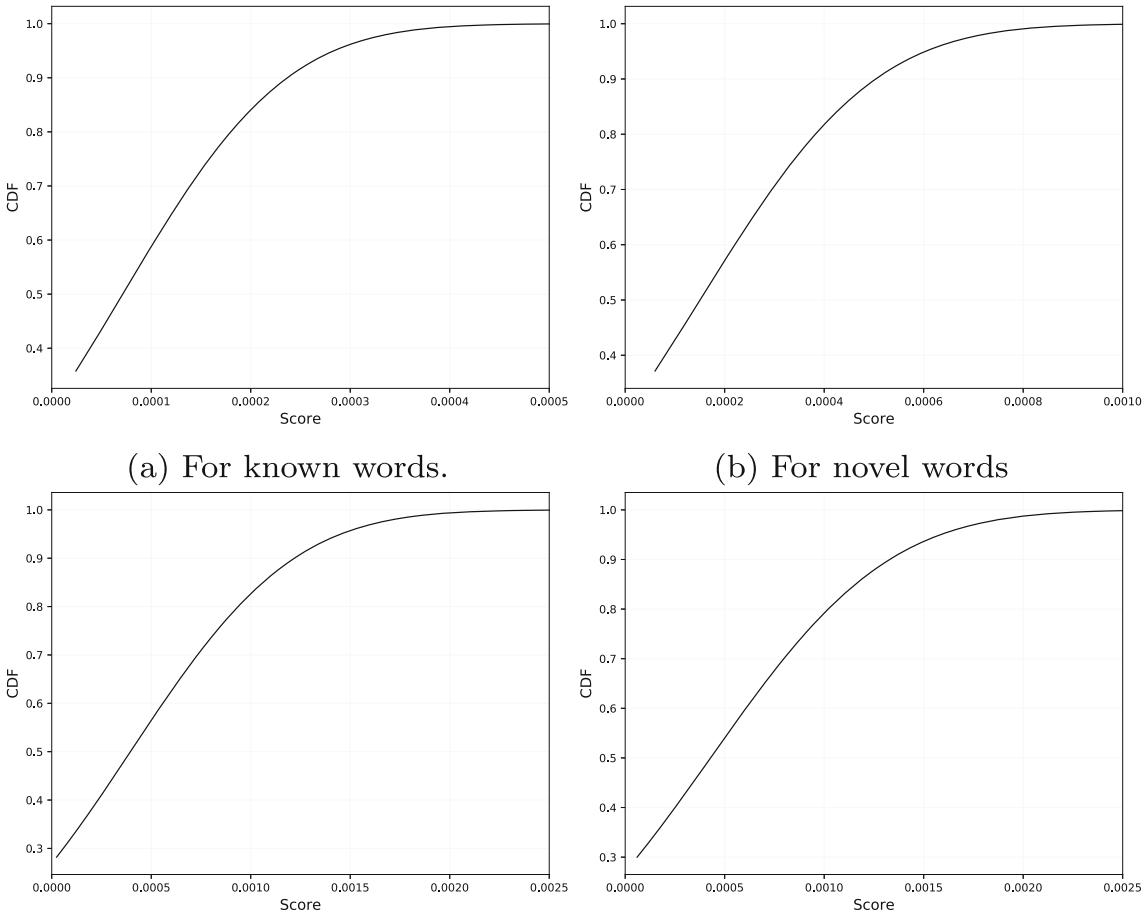


Fig. 18 Cumulative frequency distribution (CFD) graph for TF-IDF scores in different categories of 20 Newsgroups

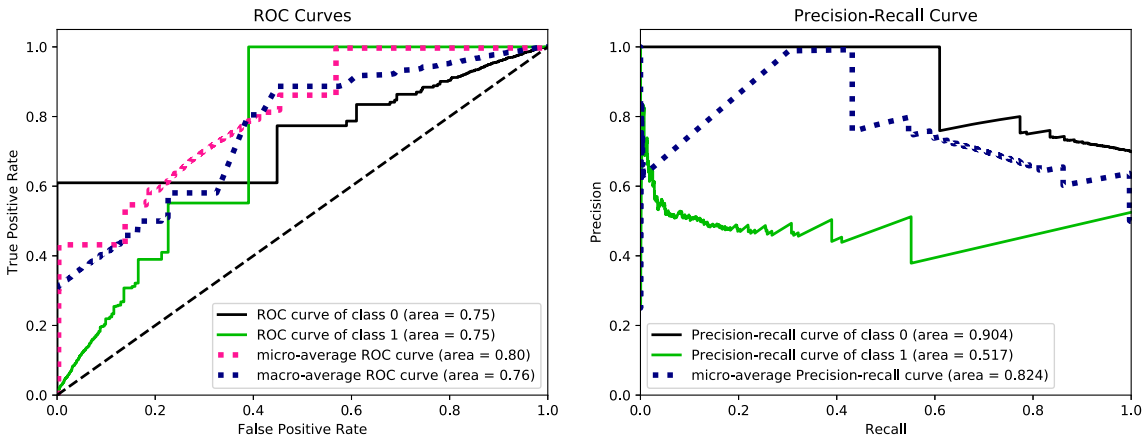


Fig. 19 ROC curve and precision-recall of known/novel class classification of 20 Newsgroups using TF-IDF scores

Table 13 Co-occurrence matrix showing the information gain between words in BBC Sports

	Manchester	Chelsea	Particular	Rugby	Flyhalf
Manchester	14363.688	6.324	4.738	0.33	0.848
Chelsea	6.324	19801.49	6.18	0.466	1.326
Particular	4.738	6.18	30863.006	2.52	4.968
Rugby	0.33	0.466	2.52	486.758	3.952
Flyhalf	0.848	1.326	4.968	3.952	8888.888

Table 15 Co-occurrence matrix showing the similarity between words in BBC Sports using Word2Vec

	Manchester	Chelsea	Particular	Rugby	Flyhalf
Manchester	1	0.782	0.653	0.598	0.718
Chelsea	0.782	1	0.891	0.820	0.829
Particular	0.653	0.891	1	0.821	0.941
Rugby	0.598	0.820	0.821	1	0.706
Flyhalf	0.718	0.829	0.941	0.706	1

5.4 20 newsgroups dataset

The 20 Newsgroups dataset contains a total of 18 828 documents partitioned equally into 20 separate classes. In our experiments, we treat the two classes "comp.graphics" and "talk.politics.guns" as *Known* topics, and then use the class "rec.sport.baseball" to represent a *Novel* topic. Again, we train a TM to produce our clause-based novelty scores. The overall statistics of the resulting word scores are shown in Tables 9 and 10, where we observe similar behavior to that observed with the BBC Sports dataset.

The CFD plot in Fig. 12 presents the score distribution among words per group (known, novel, shared). For known words, in Fig. 12a, we find that 90% of the scores of the words are below around 1.3. In Fig. 12b, however, only 45% of the novel word scores fall below approx. 1.3. From the plots, it is evident that the majority of the novel words have considerably higher scores than the known words. Note that some of the novel word's low scores are attributable to the presence of common words (e.g., stop words) in the novel bag-of-words. Since the common words, as such, do not signify novelty, the TM clauses do not frequently capture them. As a result, they receive relatively low scores despite their appearance among the novel documents.

The CFD plot for attention and TF-IDF both exhibit similar behaviour to that of BBC Sports, as seen in Fig. 13 and 18, respectively. Finally, we again observe that the clauses have used the shared words for discrimination (cf. Table 10), resulting in a mix of low and high novelty scores, as shown in Fig. 12c.

Table 11 and Table 12 provide examples of the highest-scoring words captured by KE baselines, including TM, for both classes. The visualization of the scores are presented in Figs. 14 and 15. Again, we observe a similar behavior as for the BBC Sports dataset. The ROC and precision-recall curves for our novelty scoring mechanism are illustrated in Figs. 16a, 17a, 18, and 19a include corresponding graphs when TF-IDF and attention scores are used instead. Our method outperforms the ROC value obtained from attention because of its ability to identify more number of correct novel samples, i.e., true positives TP. However, the TF-IDF surprisingly outperforms both of the methods because of its straightforward scoring system and the dataset's moderate size. We can see that our scoring approach outperforms the baselines by a wide margin.

5.5 Contextual scoring

We also implement a context-based scoring approach to investigate how multiple words interact to capture novelty. As detailed in Section 4, we compute the combined novelty score by measuring word co-occurrence in clauses. That is, we intend to demonstrate how context can help uncover novelty when words have multiple meanings. The context-based scoring is critical since the context can transform the word from being novel to known, such as the meaning of the word "apple" in "apple fruit" and "apple phone". For demonstration, we calculate our proposed context-based novelty score for five words (i.e., two known, two novel, and one common word) in both datasets. For the BBC Sports dataset, the pairwise co-occurrence scores

Table 14 Co-occurrence matrix showing the information gain between words in 20 Newsgroup

	Guns	Weapon	Gather	Baseball	Player
Guns	12302.96	17.648	15.754	4.036	4.268
Weapon	17.648	13888.888	12.108	4.66	5.102
Gather	15.754	12.108	14610.272	11.854	15.408
Baseball	4.036	4.66	11.854	4003.824	18.566
Player	4.268	5.102	15.408	18.566	9255.402

Table 16 Co-occurrence matrix showing the similarity between words in 20 Newsgroup using Word2Vec

	Guns	Weapon	Gather	Baseball	Player
Guns	1	0.709	0.726	0.673	0.701
Weapon	0.709	1	0.648	0.454	0.539
Gather	0.726	0.648	1	0.631	0.686
Baseball	0.673	0.454	0.631	1	0.764
Player	0.701	0.539	0.686	0.764	1

are presented in Table 13. We see a significant degree of correspondence between words such as "Manchester" and "Chelsea" from class *Known*. Similarly, there is a high correspondence between words such as "Rugby" and "Flyhalf" from class *Novel*. The common word "Particular", on the other hand, shows similar correspondence with words from both of the classes. Similarly, for the 20 Newsgroups dataset, the co-occurrence scores for five words selected from the known, novel, and common word types are shown in Table 14. The words "Guns" and "Weapon" are from class *Known* and manifest strong co-occurrence. Additionally the words "Baseball" and "Player" from class *Novel* correspond strongly as well. The common word "Gather", on the other hand, co-occurs within both of the classes. These examples demonstrate that the words that are most likely to appear in the same context have a high co-occurrence score. This can be explained by the fact that many clauses capture words that frequently occur together in a similar context.

We compare the contextual scores obtained from our method with the Word2Vec similarity score. To do this, we utilize *Gensim* library to train custom Word2Vec on both datasets. Gensim library enabled us to create word embeddings by training own Word2Vec models on a custom corpus using either CBOW or skip-grams algorithms. Parameter-wise, we used an embedding size of 200 and a window size of 5. We compute the cosine similarity between words by using their word vectors (embeddings). The findings are included in Tables 15 and 16. We notice a significant degree of resemblance between the corresponding words from the known and the novel classes. However, unlike our method, the similarity scores are less distinct, and the common words are not discernible score-wise.

6 Conclusion

In this work, we propose a Tsetlin Machine (TM)-based solution for word-level novelty description. First, we employ the clauses from a trained TM to capture how the most significant words differentiate a group of novel documents apart from a group of known documents. Then, we calculate the score for each word based on the role it plays in the clauses. The analysis of our empirical results for BBC Sports and 20 Newsgroups demonstrate significantly better novelty discrimination power when compared to using attention and TF-IDF. Our empirical results also show that we can capture word relations through a contextual scoring mechanism that measures co-occurrence within TM clauses. By capturing non-linear relationships among words, we can enhance the capability of measuring novelty at the word level. However, training a TM is computationally more expensive than calculating TF-IDF, particularly for large datasets with an extensive vocabulary. We will address computation speed in our future work, employing indexing mechanisms and exploiting feature space sparsity.

Funding Open access funding provided by University of Agder.

Declarations

Conflict of Interests The authors declare that they have no conflict of interest.

References

1. Abeyrathna KD, Granmo O, Jiao L, Goodwin M (2019) The regression tsetlin machine: A tsetlin machine for continuous output problems. In: Oliveira PM, Novais P, Reis LP (eds) Progress in Artificial Intelligence, 19th EPIA Conference on Artificial Intelligence, EPIA 2019, Vila Real, Portugal, September 3-6, 2019, Proceedings, Part II, Springer, Lecture Notes in Computer Science, vol 11805, pp 268–280. https://doi.org/10.1007/978-3-030-30244-3_23
2. Abeyrathna KD, Bhattarai B, Goodwin M, Gorji SR, Granmo O, Jiao L, Saha R, Yadav RK (2021) Massively parallel and asynchronous tsetlin machine architecture supporting almost constant-time scaling. In: Meila M, Zhang T (eds) Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event, PMLR, Proceedings of Machine Learning Research, vol 139, pp 10–20. http://proceedings.mlr.press/v139/abeyrathna21a.html
3. Aggarwal CC (2017) An introduction to outlier analysis. In: Outlier analysis. Springer International Publishing, pp 1–34. https://doi.org/10.1007/978-3-319-47578-3_1
4. Allan J, Papka R, Lavrenko V (1998) On-line new event detection and tracking. In: Croft WB, Moffat A, van Rijsbergen CJ, Wilkinson R, Zobel J (eds) SIGIR '98: Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, August 24-28, 1998. ACM, Australia, pp 37-45. https://doi.org/10.1145/290941.290954
5. Bendale A, Boult TE (2016) Towards open set deep networks. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016, IEEE Computer Society, pp 1563–1572. https://doi.org/10.1109/CVPR.2016.173
6. Bentivogli L, Clark P, Dagan I, Giampiccolo D (2011) The seventh PASCAL recognizing textual entailment challenge. In: Proceedings of the Fourth Text Analysis Conference, TAC 2011, Gaithersburg, Maryland, USA, November 14-15, 2011, NIST. https://tac.nist.gov/publications/2011/additional.papers/RTE7_overview.proceedings.pdf

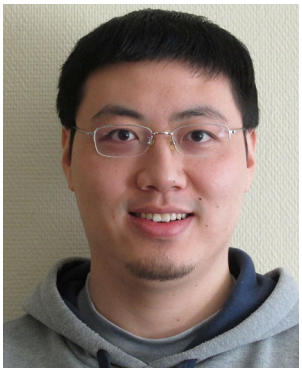
7. Berge GT, Granmo O, Tveit TO, Goodwin M, Jiao L, Matheussen BV (2019) Using the tsetlin machine to learn human-interpretable rules for high-accuracy text categorization with medical applications. IEEE Access 7:115134–115146. https://doi.org/10.1109/ACCESS.2019.2935416
8. Bhattarai B, Granmo O, Jiao L (2021) Measuring the novelty of natural language text using the conjunctive clauses of a tsetlin machine text classifier. In: Rocha AP, Steels L, van den Herik HJ (eds) Proceedings of the 13th International Conference on Agents and Artificial Intelligence, ICAART 2021, Volume 2, Online Streaming, February 4-6, 2021, SCITEPRESS, pp 410–417. https://doi.org/10.5220/0010382204100417
9. Blanchard G, Lee G, Scott C (2010) Semi-supervised novelty detection. J Mach Learn Res 11:2973–3009. http://portal.acm.org/citation.cfm?id=1953028
10. Boudin F (2016) Pke: an open source python-based keyphrase extraction toolkit. In: Watanabe H (ed) COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference System Demonstrations, December 11-16, 2016, Osaka, Japan, ACL, pp 69–73. https://aclanthology.org/C16-2015/
11. Boudin F (2018) Unsupervised keyphrase extraction with multipartite graphs. In: Walker MA, Ji H, Stent A (eds) Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, New Orleans, Louisiana, USA, June 1-6, 2018 Short Papers, vol 2. Association for Computational Linguistics, pp 667–672. https://doi.org/10.18653/v1/n18-2105
12. Bougouin A, Boudin F, Daille B (2013) Topicrank: Graph-based topic ranking for keyphrase extraction. In: Sixth International Joint Conference on Natural Language Processing, IJCNLP 2013, Nagoya, Japan, October 14-18, 2013, Asian Federation of Natural Language Processing / ACL, pp 543–551. https://aclanthology.org/I13-1062/
13. Campos R, Mangaravite V, Pasquali A, Jorge A, Nunes C, Jatowt A (2020) Yake! keyword extraction from single documents using multiple local features. Inf Sci 509:257–289. https://doi.org/10.1016/j.ins.2019.09.013
14. Carbinell J, Goldstein J (2017) The use of mmr, diversity-based reranking for reordering documents and producing summaries. SIGIR Forum 51(2):209–210. https://doi.org/10.1145/3130348.3130369
15. Chandola V, Banerjee A, Kumar V (2012) Anomaly detection for discrete sequences: a survey. IEEE Trans Knowl Data Eng 24(5):823–839. https://doi.org/10.1109/TKDE.2010.235
16. Dasgupta D, Nino L (2000) A comparison of negative and positive selection algorithms in novel pattern detection, Proceedings of the IEEE International Conference on Systems, Man & Cybernetics. In: Cybernetics evolving to systems, humans, organizations, and their complex interactions, sheraton music city hotel, nashville, tennessee, USA. 8-11 October, 2000. IEEE, pp 125–130. https://doi.org/10.1109/ICSMC.2000.884976
17. Dasgupta T, Dey L (2016) Automatic scoring for innovativeness of textual ideas. In: Fortuna B, Grobelnik M, ERH Jr, Witbrock MJ (eds) Knowledge Extraction from Text, Papers from the 2016 AAAI Workshop, Phoenix, Arizona, USA, February 12, 2016, AAAI Press, AAAI Workshops, vol WS-16-10. http://www.aaai.org/ocs/index.php/WS/AAAIW16/paper/view/12663
18. Devlin J, Chang M, Lee K, Toutanova K (2019) BERT: Pre-training of deep bidirectional transformers for language understanding. In: Burstein J, Doran C, Solorio T (eds) Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019 vol 1 (Long and Short Papers). Association for Computational Linguistics, pp 4171–4186. https://doi.org/10.18653/v1/n19-1423
19. Duchi JC, Hazan E, Singer Y (2011) Adaptive subgradient methods for online learning and stochastic optimization. J Mach Learn Res 12:2121–2159. http://dl.acm.org/citation.cfm?id=2021068
20. Fei G, Liu B, Callison-burch C, Su J, Pighin D, Marton Y (2015) Social media text classification under negative covariate shift. In: Márquez L (ed) Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015, The Association for Computational Linguistics, pp 2347–2356. https://doi.org/10.18653/v1/d15-1282
21. Fei G, Liu B (2016) Breaking the closed world assumption in text classification. In: Knight K, Nenkova A, Rambow O (eds) NAACL HLT 2016, The 2016 conference of the north american chapter of the association for computational linguistics: Human language technologies, san diego california, USA, June 12-17, 2016, The Association for Computational Linguistics, pp 506–514. https://doi.org/10.18653/v1/n16-1061
22. Galassi A, Lippi M, Torroni P (2021) Attention in natural language processing. IEEE Trans Neural Networks Learn Syst 32(10):4291–4308. https://doi.org/10.1109/TNNLS.2020.3019893
23. Goodfellow IJ, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville AC, Bengio Y (2014) Generative adversarial nets. In: Ghahramani Z, Welling M, Cortes C, Lawrence ND, Weinberger KQ (eds) Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada, pp 2672–2680. https://proceedings.neurips.cc/paper/2014/hash/5ca3e9b122f61f8f06494c97b1afccf3-Abstract.html
24. Granmo O (2018) The tsetlin machine - A game theoretic bandit driven approach to optimal pattern recognition with propositional logic. arXiv:1804.01508
25. Granmo O, Glimsdal S, Jiao L, Goodwin M, Omlin CW, Berge GT (2019)
26. Grootendorst M (2020) Keybert: Minimal keyword extraction with bert. https://doi.org/10.5281/zenodo.4461265
27. Hanley JA, McNeil BJ (1982) The meaning and use of the area under a receiver operating characteristic (roc) curve. Radiology 143(1):29–36. https://doi.org/10.1148/radiology.143.1.7063747
28. Hautamäki V, Kärkkäinen I, Fränti P (2004) Outlier detection using k-nearest neighbour graph. In: 17th International Conference on Pattern Recognition, ICPR 2004, Cambridge, UK, August 23-26, 2004, IEEE Computer Society, pp 430–433. https://doi.org/10.1109/ICPR.2004.1334558
29. Hawkins DM (1980) Identification of outliers, Monographs on Applied Probability and Statistics. Springer, Berlin. https://doi.org/10.1007/978-94-015-3994-4
30. Hendrycks D, Gimpel K (2017) A baseline for detecting misclassified and out-of-distribution examples in neural networks. In: 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings, OpenReview.net. https://openreview.net/forum?id=Hkg4TI9xl
31. Hill F, Cho K, Korhonen A (2016) Learning distributed representations of sentences from unlabelled data. In: Knight K, Nenkova A, Rambow O (eds) NAACL HLT 2016, The 2016 conference of the north american chapter of the association for computational linguistics: Human language technologies, san diego california, USA, June 12-17, 2016, The Association for Computational Linguistics, pp 1367–1377. https://doi.org/10.18653/v1/n16-1162
32. Hospedales TM, Gong S, Xiang T (2011) Finding rare classes: Adapting generative and discriminative models in active learning.

In: Huang JZ, Cao L, Srivastava J (eds) Advances in Knowledge Discovery and Data Mining - 15th Pacific-Asia Conference, PAKDD 2011, Shenzhen, China, May 24-27, 2011, Proceedings, Part II, Springer, Lecture Notes in Computer Science, vol 6635, pp 296–308. https://doi.org/10.1007/978-3-642-20847-8_25
33. Jain LP, Scheirer WJ, Boult TE (2014) Multi-class open set recognition using probability of inclusion. In: Fleet DJ, Pajdla T, Schiele B, Tuytelaars T (eds) Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part III, Springer, Lecture Notes in Computer Science, vol 8691, pp 393–409. https://doi.org/10.1007/978-3-319-10578-9_26
34. Jain S, Wallace BC (2019) Attention is not explanation. In: Burstein J, Doran C, Solorio T (eds) Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, vol 1 (Long and Short Papers). Association for Computational Linguistics, pp 3543–3556. https://doi.org/10.18653/v1/n19-1357
35. Jiao L, Zhang X, Granmo O, Abeyrathna KD (2021) On the convergence of tsetlin machines for the XOR operator. arXiv:2101.02547
36. Kiros R, Zhu Y, Salakhutdinov R, Zemel RS, Urtasun R, Torralba A, Fidler S (2015) Skip-thought vectors. In: Cortes C, Lawrence ND, Lee DD, Sugiyama M, Garnett R (eds) Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada, pp 3294–3302. https://proceedings.neurips.cc/paper/2015/hash/f442d33fa06832082290ad8544a8da27-Abstract.html
37. Kliger M, Fleishman S (2018) Novelty detection with GAN. arXiv:1802.10560
38. Kumaran G, Allan J (2004) Text classification and named entities for new event detection. In: Sanderson M, Järvelin K, Allan J, Bruza P (eds) SIGIR 2004: Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Sheffield, UK, July 25-29, 2004, ACM, pp 297–304. https://doi.org/10.1145/1008992.1009044
39. LANN E (1959) Testing statistical hypotheses. Wiley, New York
40. Li Z, Zhao Y, Botta N, Ionescu C, Hu X (2020) COPOD: Copula-based outlier detection. In: Plant C, Wang H, Cuzzocrea A, Zaniolo C, Wu X (eds) 20Th IEEE international conference on data mining, ICDM 2020, sorrento, italy, november 17-20, 2020, IEEE, pp 1118-1123. https://doi.org/10.1109/ICDM50108.2020.00135
41. Liu Y, Li Z, Zhou C, Jiang Y, Sun J, Wang M, He X (2020) Generative adversarial active learning for unsupervised outlier detection. IEEE Trans Knowl Data Eng 32(8):1517–1528. https://doi.org/10.1109/TKDE.2019.2905606
42. Peters ME, Neumann M, Iyyer M, Gardner M, Clark C, Lee K, Zettlemoyer L (2018) Deep contextualized word representations. In: Walker MA, Ji H, Stent A (eds) Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, vol 1 (Long Papers). Association for Computational Linguistics, pp 2227–2237. https://doi.org/10.18653/v1/n18-1202
43. Pimentel MAF, Clifton DA, Clifton LA, Tarassenko L (2014) A review of novelty detection. Signal Process 99:215–249. https://doi.org/10.1016/j.sigpro.2013.12.026
44. Platt JC (1999) Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In: Advances in large margin classifiers
45. Ramos J et al (2003) Using tf-idf to determine word relevance in document queries. In: Proceedings of the first instructional conference on machine learning, pp 29–48
46. Rebuffi S, Kolesnikov A, Sperl G, Lampert CH (2017) icarl: Incremental classifier and representation learning. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017, IEEE Computer Society, pp 5533–5542. https://doi.org/10.1109/CVPR.2017.587
47. Saha R, Granmo O, Goodwin M (2020) Mining interpretable rules for sentiment and semantic relation analysis using tsetlin machines. In: Bramer M, Ellis R (eds) Artificial Intelligence XXXVII - 40th SGAI International Conference on Artificial Intelligence, AI 2020, Cambridge, UK, December 15-17, 2020, Proceedings, Springer, Lecture Notes in Computer Science, vol 12498, pp 67–78. https://doi.org/10.1007/978-3-030-63799-6_5
48. Salazar J, Liang D, Nguyen TQ, Kirchhoff K (2020) Masked language model scoring. In: Jurafsky D, Chai J, Schluter N, Tetreault JR (eds) Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020, Association for Computational Linguistics, pp 2699–2712. https://doi.org/10.18653/v1/2020.acl-main.240
49. Scheirer WJ, de Rezende Rocha A, Sapkota A, Boult TE (2013) Toward open set recognition. IEEE Trans Pattern Anal Mach Intell 35(7):1757–1772. https://doi.org/10.1109/TPAMI.2012.256
50. Schölkopf B, Platt JC, Shawe-taylor J, Smola AJ, Williamson RC (2001) Estimating the support of a high-dimensional distribution. Neural Comput 13(7):1443–1471. https://doi.org/10.1162/089976601750264965
51. Serrano S, Smith NA (2019) Is attention interpretable?. In: Korhonen A, Traum DR, Márquez L (eds) Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, vol 1 (Long Papers). Association for Computational Linguistics, pp 2931-2951. https://doi.org/10.18653/v1/p19-1282
52. Soboroff I, Harman D (2005) Novelty detection: The TREC experience. In: HLT/EMNLP 2005, Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference, 6-8 October 2005, Vancouver, British Columbia, Canada, The Association for Computational Linguistics, pp 105–112. https://aclanthology.org/H05-1014/
53. Sun X, Lu W (2020) Understanding attention for text classification. In: Jurafsky D, Chai J, Schluter N, Tetreault JR (eds) Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020, Association for Computational Linguistics, pp 3418–3428. https://doi.org/10.18653/v1/2020.acl-main.312
54. Tax DMJ, Duin RPW (1998) Outlier detection using classifier instability. In: Amin A, Dori D, Pudil P, Freeman H (eds) Advances in pattern recognition, joint IAPR international workshops SSPR '98 and SPR '98, sydney, NSW, Australia, August 11-13, 1998 (Proceedings, Springer), Lecture Notes in Computer Science, vol 1451, pp 593-601. https://doi.org/10.1007/BFb0033283
55. Tax DMJ, Duin RPW (2004) Support vector data description. Mach Learn 54(1):45–66. https://doi.org/10.1023/B:MACH.0000008084.60811.49
56. Tsai FS, Tang W, Chan KL (2010) Evaluation of novelty metrics for sentence-level novelty mining. Inf Sci 180(12):2359–2374. https://doi.org/10.1016/j.ins.2010.02.020
57. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser L, Polosukhin I (2017) Attention is all you need. In: Guyon I, von Luxburg U, Bengio S, Wallach HM, Fergus R, Vishwanathan SVN, Garnett R (eds) Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA, pp 5998–6008. https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html

58. Wang Y, Huang H, Feng C, Zhou Q, Gu J, Gao X (2016) CSE: Conceptual sentence embeddings based on attention model. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1 (Long Papers). The Association for Computer Linguistics. https://doi.org/10.18653/v1/p16-1048
59. Yadav RK, Jiao L, Granmo O, Goodwin M (2021a) Human-level interpretable learning for aspect-based sentiment analysis. In: In: Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021, AAAI Press, pp 14203–14212. https://ojs.aaai.org/index.php/AAAI/article/view/17671
60. Yadav RK, Jiao L, Granmo OC, Goodwin M (2021b) Enhancing interpretable clauses semantically using pretrained word representation. In: Proceedings of the Fourth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP, Association for Computational Linguistics, Punta Cana, Dominican Republic, pp 265–274 https://doi.org/10.18653/v1/2021.blackboxnlp-1.19. https://aclanthology.org/2021.blackboxnlp-1.19
61. Yu Y, Qu W, Li N, Guo Z (2017) Open category classification by adversarial sample generation. In: Sierra C (ed) Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017, ijcai.org, pp 3357–3363. https://doi.org/10.24963/ijcai.2017/469
62. Zhang X, Jiao L, Granmo O, Goodwin M (5555) On the convergence of tsetlin machines for the identity- and not operators. IEEE Transactions on Pattern Analysis & Machine Intelligence pp 1–1. https://doi.org/10.1109/TPAMI.2021.3085591
63. Zhang Y, Tsai FS (2009) Combining named entities and tags for novel sentence detection. In: Proceedings of the WSDM'09 Workshop on Exploiting Semantic Annotations in Information Retrieval, Association for Computing Machinery, New York, NY, USA, ESAIR '09, p 30–34. https://doi.org/10.1145/1506250. 1506256.

Bimal Bhattarai received the B.Tech. degree in electronics and communication engineering from JNTUA, India, in 2015, and the M.Sc. degree from the Department of Information and Communication Engineering, Chosun University, South Korea. He is currently a PhD Scholar with the Department of Information and Communication Technology, UiA. His research interests include data science, natural language processing, tsetlin machine, indoor positioning and navigation, machine learning, and deep learning applications.

Ole-Christoffer Granmo is the Founding Director of Centre for Artificial Intelligence Research (CAIR), University of Agder, Norway. He obtained his master's degree in 1999 and the PhD degree in 2004, both from the University of Oslo, Norway. Dr. Granmo has authored in excess of 150 refereed papers with 6 best paper awards within machine learning, encompassing learning automata, bandit algorithms, Tsetlin machines, Bayesian reasoning, reinforcement learning, and computational linguistics. He has further coordinated 7+ research projects and graduated 55+ master- and 7 PhD students. Dr. Granmo is also a co-founder of the Norwegian Artificial Intelligence Consortium (NORA). Apart from his academic endeavours, he co-founded the company Anzyz Technologies AS.



Lei Jiao received the B.E. degree in telecommunications engineering from Hunan University, Changsha, China, in 2005, the M.E. degree in communication and information system from Shandong University, Jinan, China, in 2008, and the Ph.D. degree in information and communication technology from the University of Agder (UiA), Norway, in 2012. He is currently working as an Associate Professor with the Department of Information and Communication Technology, UiA. His research interests include reinforcement learning, Tsetlin machine, resource allocation and performance evaluation for communication and energy systems.