Measuring the Novelty of Natural Language Text using the Conjunctive Clauses of a Tsetlin Machine Text Classifier

Bimal Bhattarai[a], Ole-Christoffer Granmo[b] and Lei Jiao[c]

Department of Information and Communication Technology, University of Agder, Grimstad, Norway

Keywords: Novelty Detection, Deep Learning, Interpretable, Tsetlin Machine.

Abstract: Most supervised text classification approaches assume a closed world, counting on all classes being present in the data at training time. This assumption can lead to unpredictable behaviour during operation, whenever novel, previously unseen, classes appear. Although deep learning-based methods have recently been used for novelty detection, they are challenging to interpret due to their black-box nature. This paper addresses *interpretable* open-world text classification, where the trained classifier must deal with novel classes during operation. To this end, we extend the recently introduced Tsetlin machine (TM) with a novelty scoring mechanism. The mechanism uses the conjunctive clauses of the TM to measure to what degree a text matches the classes covered by the training data. We demonstrate that the clauses provide a succinct interpretable description of known topics, and that our scoring mechanism makes it possible to discern novel topics from the known ones. Empirically, our TM-based approach outperforms seven other novelty detection schemes on three out of five datasets, and performs second and third best on the remaining, with the added benefit of an interpretable propositional logic-based representation.

SCITEPRESS
SCIENCE AND TECHNOLOGY PUBLICATIONS

1 INTRODUCTION

In recent years, deep learning-based techniques have achieved superior performance on many text classification tasks. Most of the classifiers use supervised learning, assuming a closed-world environment (Scheirer et al., 2013). That is, the classes presented in the test data (or during operation) are also assumed to be presented in the training data. However, when facing an open-world environment, new classes may appear after training (Bendale and Boult, 2016). In such cases, assuming a closed world can lead to unpredictable behaviour. For example, a chatbot interacting with a human user will regularly face new user intents that it has not been trained to recognize. A chatbot for banking services may, for instance, have been trained to recognize the intent of applying for a loan. However, it will provide meaningless responses if it fails to recognize that asking for a lower interest rate is new and different user intent. The problem with neural network-based supervised classifiers that use the typical softmax layer is that they erroneously force novel input into one of the previously seen classes by normalizing the class output scores to produce a distribution that sums to 1.0. Instead, a robust classifier should be able to flag input as a novel, rejecting to label it according to the presently known classes. Recently, many important application areas make use of novelty detection such as medical applications, fraud detection (Veeramreddy et al., 2011), sensor networks (Zhang et al., 2010), and text analysis (Basu et al., 2004). For a further study of these classes of techniques, the reader is referred to (Pimentel et al., 2014).

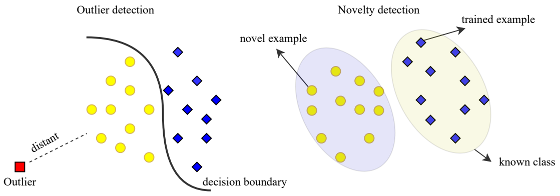

Figure 1: Visualization of outlier detection and novelty detection.

Figure 1 illustrates the problem of novelty detection, i.e., recognizing when the data fed to a classifier is novel and somehow differs from the data that was available during training. In brief, after training on

[a] https://orcid.org/0000-0002-7339-3621
[b] https://orcid.org/0000-0002-7287-030X
[c] https://orcid.org/0000-0002-7115-6489

Bhattarai, B., Granmo, O. and Jiao, L.
Measuring the Novelty of Natural Language Text using the Conjunctive Clauses of a Tsetlin Machine Text Classifier.
DOI: 10.5220/0010382204100417
In *Proceedings of the 13th International Conference on Agents and Artificial Intelligence (ICAART 2021) - Volume 2*, pages 410-417
ISBN: 978-989-758-484-8
Copyright © 2021 by SCITEPRESS – Science and Technology Publications, Lda. All rights reserved

data with known classes (blue data points to the right in Figure 1), the classifier should be able to detect novel data arising from new, and previously unseen classes (yellow data points to the right in the figure). We will refer to the known data points as *positive examples* and the novel data points as *negative examples*. Note that the problem of novelty detection is closely related to so-called outlier detection (Chandola et al., 2009; Pincus, 1995). However, as exemplified to the left in Figure 1, the latter problem involves flagging data points that are part of an already known class, yet deviating from the other data points, e.g., due to measurement errors or anomalies (red point).

Problem Definition: Generally, in multiclass classification, we have a set of example data points $X = (x_i, y_i), x_i \in \mathcal{R}^s$, where $i = (1, 2, 3, \dots, N)$ indexes the positive examples, s is the dimensionality of the data point x_i, and y_i is the label of x_i, assigning it a class. For any data point x_i, also referred to as a feature vector, a classifier function $\hat{y} = \mathcal{F}(x; X)$ is to assign the data point a predicted class $\hat{y}$, after the function has been fitted to the training data X. Additionally, a novelty scoring function $z(x; X)$, also fitted on X, calculates a novelty score, so that a threshold σ can be used to recognize novel input. In other words, the classifier is to return the correct class label while simultaneously being capable of rejecting novel examples.

Under the above circumstances, standard supervised learning would fail, particularly for methods based on building a discriminant function, such as neural networks. As shown to the left of Figure 1, a discriminant function captures the discriminating "boundaries" between the known classes and cannot readily be used to discern novel classes. Still, a traditional way of implementing novelty detection is to threshold the entropy of the class probability distribution (Hendrycks and Gimpel, 2017). Such methods are not actually measuring novelty but closeness to the decision boundary. Such an approach thus leads to undetected novel data points when the data points are located far from the decision boundary. Another, and perhaps more robust, approach is to take advantage of the class likelihood function, which estimates the probability of the data given the class.

Paper Contributions: Unlike traditional methods, in this paper we leverage the conjunctive clauses in propositional logic that a TM builds. The TM clauses represent frequent patterns in the data, and our hypothesis is that these frequent patterns characterize the known classes succinctly and comprehensively. Thus, we establish a novelty scoring mechanism based simply on counting how many clauses match the input. This score can, in turn, be thresholded manually to flag novel input. However, for more robust novelty detection, we train several standard machine learning techniques to find accurate thresholds. The main contributions of our work can thus be summarized as follows:

- We devise the first TM-based approach for novelty detection, leveraging the clauses of the TM architecture.
- We illustrate how the new technique can be used to detect novel topics in text, and compare the technique against widely used approaches on five different datasets.

2 RELATED WORK

The perhaps most common approach to novelty detection is distance-based methods (Hautamaki et al., 2004), which assumes that the known or seen data are clustered together while novel data has a high distance to the clusters. The major drawback of these methods is computational complexity when performing clustering or nearest neighbor search in a large dataset. Early work on novelty detection also includes one-class SVMs (Schölkopf et al., 2001), which are only capable of using the positive training examples to maximize the class margin. This shortcoming is overcome by the Center-Based Similarity (CBS) space learning method (Fei and Liu, 2015), which uses binary classifiers over vector similarities of training examples transformed into the center of the class. To build a classifier for detecting novel class distributions, Chow et al. proposed a confidence score-based method that suggests an optimum input rejection rule (Chow, 1970). The method is relatively accurate, but it does not scale well to high-dimensional datasets.

The novelty detection method OpenMax (Bendale and Boult, 2016) is more recent, and estimates the probability of the input belonging to a novel class. To achieve this, the method employs an extra layer, connected to the penultimate layer of the original network. However, the computational complexity of the method is high, and the underlying inference cannot easily be interpreted for quality assurance. Lately, Yu et al. (Yu et al., 2017) adopted the Adversarial Sample Generation (ASG) framework (Hautamaki et al., 2004) to generate positive and negative examples in an unsupervised manner. Then, based on those examples, they trained an SVM classifier for novelty detection. Furthermore, in computer vision, Scheirer et al. introduced the concept of open space risk to recognize novel image content (Scheirer et al., 2013). They proposed a "1-vs-set machine" that creates a decision space using a binary SVM classifier, with two parallel hyperplanes bounding the non-novel regions. In Section 4, we compare the performance of our new TM-based approach with the most widely used approaches among those mentioned above.

3 TSETLIN MACHINE-BASED NOVELTY DETECTION

In this section, we propose our approach to novelty detection based on the TM. First, we explain the architecture of the TM. Then, we show how novelty scores can be obtained from the TM clauses. Finally, we integrate the TM with a rule-based classifier for novel text classification.

3.1 Tsetlin Machine (TM) Architecture

The TM is a recent approach to pattern classification (Granmo, 2018) and regression (Darshana Abeyrathna et al., 2020). It builds on a classic learning mechanism called a Tsetlin automaton (TA), developed by M. L. Tsetlin in the early 1960s (Tsetlin, 1961). In all brevity, multiple teams of TA combine to form the TM. Each team is responsible for capturing a frequent sub-pattern in high precision by composing a conjunctive clause. In-built resource allocation principles guide the teams to distribute themselves across the underlying sup-patterns of the problem. Recently, the TM has performed competitively with the state-of-the-art techniques including deep neural networks, for text classification (Berge et al., 2019), and aspect-based sentiment analysis (Yadav et al., 2021). Further, the convergence of TM has been studied in (Zhang et al., 2020). In what follows, we propose a new scheme that extends the TM with the capability to recognize novel patterns.

Figure 2 describes the building blocks of a TM. As seen, a vanilla TM takes a vector $X = (x_1, \ldots, x_o)$ of binary features as input (Figure 3). We binarize text by using binary features that capture the presence/absence of terms in a vocabulary, akin to a bag of words, as done in (Berge et al., 2019). However, as opposed to a vanilla TM, our scheme does not output the predicted class. Instead, it calculates a novelty score per class.

Together with their negated counterparts, $\bar{x}_k = \neg x_k = 1 - x_k$, the features form a literal set, $L = \{x_1, \ldots, x_o, \bar{x}_1, \ldots, \bar{x}_o\}$. A TM pattern is formulated as a conjunctive clause C_j^+ or C_j^-, where $j = (1, \ldots, m/2)$ denotes an index of a clause, and the superscript describes the polarity of a clause. In more detail, the total number of clauses, m, is divided into two parts, where half of the clauses are assigned a positive polarity and the other half are assigned a negative polarity. Any clause, regardless of the polarity, is formed by ANDing a subset of the literal set. For example, the j^{th} clause with positive polarity, $C_j^+(X)$, can be expressed as:

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

where $L_j^+ \subseteq L$ is the set of literals that are involved in the expression of $C_j^+(X)$ after training. For instance, given clause $C_1^+(X) = x_1 x_2$, it consists of the literals $L_1^+ = \{x_1, x_2\}$ and outputs 1 if $x_1 = x_2 = 1$. The output of a conjunctive clause is determined by evaluating it on the input literals. When a clause outputs 1, this means that it has recognized a pattern in the input. Conversely, the clause outputs 0 when no pattern is recognized. The clause outputs, in turn, are combined into a classification decision through summation and thresholding using the unit step function u:

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

That is, the classification is performed based on a majority vote, with the positive clauses voting for $y = 1$ and the negative for $y = 0$. The classifier $\hat{y} = u(x_1\bar{x}_2 + \bar{x}_1 x_2 - x_1 x_2 - \bar{x}_1\bar{x}_2)$, e.g., captures the XOR-relation.

A clause is composed by a team of TA, each TA deciding to *Exclude* or *Include* a specific literal in the clause. The TA learns which literals to include based on reinforcement: Type I feedback is designed to produce frequent patterns, while Type II feedback increases the discriminating power of the patterns (see (Granmo, 2018) for details).

3.2 Novelty Detection Architecture

For novelty detection, however, we here propose to treat all clause output as positive, disregarding clause polarity. This is because both positive and negative clauses capture patterns in the training data, and thus can be used to detect novel input. We use this sum of absolute clause outputs as a novelty score, which denotes the resemblance of the input to the patterns formed by clauses during training. The resulting modified TM architecture is captured by Figure 3, showing how four different outputs (i.e., two per class) are produced by the TM. These outputs form the basis for novelty detection.

The overall novelty detection architecture is shown in Figure 4. Each TM (one per class) produces two

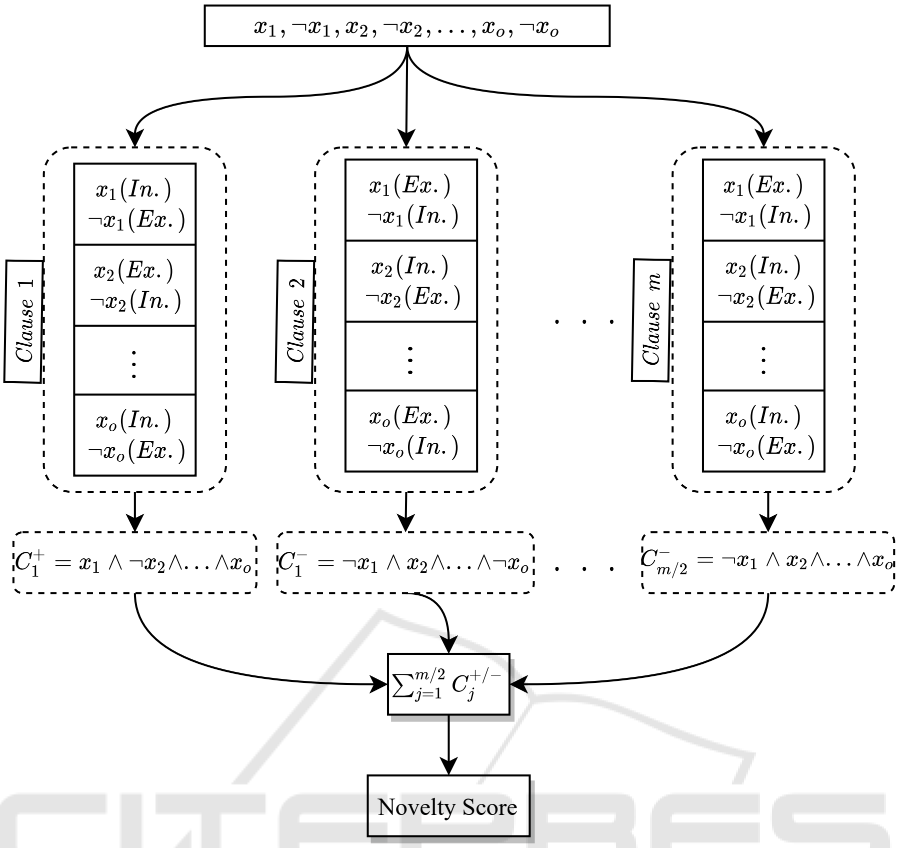
$x_1, \neg x_1, x_2, \neg x_2, \ldots, x_o, \neg x_o$
Clause 1
Clause 2
Clause m
$x_1(In.)$ $\neg x_1(Ex.)$
$x_2(Ex.)$ $\neg x_2(In.)$
$x_o(In.)$ $\neg x_o(Ex.)$
$x_1(Ex.)$ $\neg x_1(In.)$
$x_2(In.)$ $\neg x_2(Ex.)$
$x_o(Ex.)$ $\neg x_o(In.)$
$x_1(Ex.)$ $\neg x_1(In.)$
$x_2(In.)$ $\neg x_2(Ex.)$
$x_o(In.)$ $\neg x_o(Ex.)$
$C_1^+ = x_1 \wedge \neg x_2 \wedge \ldots \wedge x_o$
$C_1^- = \neg x_1 \wedge x_2 \wedge \ldots \wedge \neg x_o$
$C_{m/2}^- = \neg x_1 \wedge x_2 \wedge \ldots \wedge x_o$
$\sum_{j=1}^{m/2} C_j^{+/-}$
Novelty Score

Figure 2: Optimization of clauses in TM for generating novelty score.

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

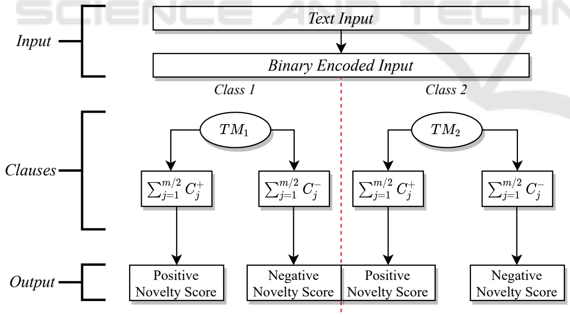

Figure 3: Multiclass Tsetlin Machine (MTM) framework to produce the novelty score for each class.

novelty scores for its respective class, one based on the positive polarity clauses and another based on the negative polarity clauses. The novelty scores are normalized and then given to a classifier, such as decision tree (DT), k-nearest neighbor (KNN), support vector machine (SVM), and logistic regression (LR). The output from these classifiers are "Novel" or "Not Novel", i.e. *0* or *1* in the figure.

For illustration purposes, instead of using a machine learning algorithm to decide upon novelty, one can instead use a simple rule-based classifier, introducing a classification threshold T. Then the novelty score for the input sentence can be compared with T to detect whether a sentence is novel or not. That is, T decides how many clauses must match to qualify the input as non-novel. The classification function $\mathcal{F}(X)$ for a single class can accordingly be given as:

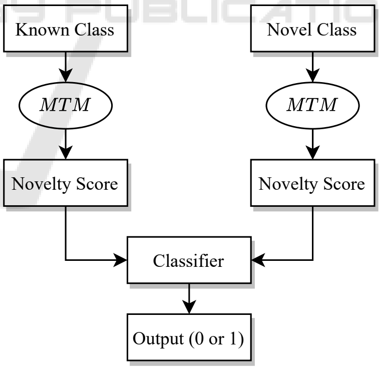

Figure 4: Novelty detection architecture.

$$\mathcal{F}(X) = \begin{cases} 1, & \textbf{if } \sum_{j=1}^{m/2} C_j^+(X) > T, \\ 1, & \textbf{if } \sum_{j=1}^{m/2} C_j^-(X) < -T, \\ 0, & \textbf{otherwise}. \end{cases}$$

4 EXPERIMENTS AND RESULTS

In this section, we present the experimental evaluation of our proposed TM model and compare it with other baseline models across five datasets.

4.1 Datasets

We used the following datasets for evaluation.

- 20-newsgroup Dataset: The dataset contains 20 classes with a total of 18,828 documents. In our experiment, we consider the two classes *"comp.graphics"* and *"talk.politics.guns"* as known classes and the class *"rec.sport.baseball"* as novel. We take 1000 samples from both known and novel classes for novelty classification.
- CMU Movie Summary Corpus: The dataset contains 42,306 movie plot summaries extracted from Wikipedia and metadata extracted from Freebase. In our experiment, we consider the two movie categories *"action"* and *"horror"* as known classes and *"fantasy"* as novel.
- Spooky Author Identification Dataset: The dataset contains 3,000 public domain books from the following horror fiction authors: *Edgar Allan Poe (EAP), HP Lovecraft (HPL)*, and *Mary Shelley (MS)*. We train on written texts from *EAP* and *HPL* while treating texts from *MS* as novel.
- Web of Science Dataset (Kowsari et al., 2017): This dataset contains 5,736 published papers, with eleven categories organized under three main categories. We use two of the main categories as known classes, and the third as a novel class.
- BBC Sports Dataset: This dataset contains 737 documents from the BBC Sport website, organized in five sports article categories and collected from 2004 to 2005. In our work, we use *"cricket"* and *"football"* as the known classes and *"rugby"* as novel.

4.2 A Case Study

To cast light on the interpretability of our scheme, we here use substrings from the 20 Newsgroup dataset, demonstrating novelty detection on a few simple cases. First, we form an indexed vocabulary set V, including all literals from the dataset. The input text is binarized based upon the index of the literals in V. For example, if a word in the text substring has been assigned index 5 in V, the 5^{th} position of the input vector is set to 1. If a word is absent from the substring, its corresponding feature is set to 0. Let us consider substrings from the two known classes and the novel class from the 20 Newsgroup dataset.

- **Class:** comp.graphics (known)
  Text: Presentations are solicited on all aspects of Navy-related scientific visualization and virtual reality.
  Literals: *"Presentations"*, *"solicited"*, *"aspects"*, *"Navy"*, *"related"*, *"scientific"*, *"visualization"*, *"virtual"*, *"reality"*.
- **Class:** talk.politics.guns (known)
  Text: Last year the US suffered almost 10,000 wrongful or accidental deaths by handguns alone. In the same year, the UK suffered 35 such deaths.
  Literals: *"Last "*, *"year"*, *"US"*, *"suffered"*, *"wrongful"*, *"accidental"*, *"deaths"*, *"handguns"*, *"UK"*, *"suffered"*.
- **Class:** rec.sport.baseball (Novel)
  Text: The top 4 are the only true contenders in my mind. One of these 4 will definitely win the division unless it snows in Maryland.
  Literals: *"top "*, *"only"*, *"true"*, *"contenders"*, *"mind"*, *"win"*, *"division"*, *"unless"*, *"snows"*, *"Maryland"*.

After training, the two known classes form conjunctive clauses that capture literal patterns reflecting the textual content. For the above example, we get the following clauses:

- C_1^+ = *"Presentations"* ∧ *"aspects"* ∧ *"Navy"* ∧ *"scientific"* ∧ *"virtual"* ∧ *"reality"* ∧ *"year"* ∧ *"US"* ∧ *"mind"* ∧ *"division"*
- C_1^- = ¬(*"suffered"* ∧ *"accidental"* ∧ *"unless"* ∧ *"snows"*)
- C_2^+ = *"last"* ∧ *"year"* ∧ *"US"* ∧ *"wrongful"* ∧ *"deaths"* ∧ *"accidental"* ∧ *"handguns"* ∧ *"Navy"* ∧ *"snows"* ∧ *"Maryland"* ∧ *"divisions"*
- C_2^- = ¬(*"presentations"* ∧ *"solicited"* ∧ *"virtual"* ∧ *"top"* ∧ *"win"*)

Table 1: Novelty score example when known text sentence is passed to the model.

Class	C_1^+	C_1^-	C_2^+	C_2^-
Known	+6	-3	+3	-5
Novel	+2	-1	+1	-2

Here, the clauses from each class captures the frequent patterns from the class. However, it may also contain certain literals from other classes. The positive polarity clauses provide evidence on the presence of a class, while negative polarity clauses provide evidence on the absence of the class. The novelty score for each class is calculated based on the propositional

Table 2: Accuracy of different machine learning classifiers to detect novel class in various dataset.

Dataset	DT	KNN	SVM	LR	NB	MLP
20 Newsgroup	72.5 %	82 %	78.0 %	72.75 %	69.0 %	**82.50 %**
Spooky action author	53.42 %	57.89 %	63.15 %	52.63 %	58.68 %	**63.15 %**
CMU movie	61.05 %	64.73 %	62.10 %	55.00 %	58.94 %	**68.68** %
BBC sports	84.21 %	85.96 %	75.43%	70.17 %	73.68 %	**89.47** %
Web of Science	64.70 %	67.97 %	69.93 %	67.10 %	62.09 %	**70.37** %

formula formed by the clauses (Figure 2). In general, for input from known classes, the novelty scores are higher. This is because the clauses have been trained to vote for or against input from the known class. For example, when we pass a known class to our model, the clauses might produce scores as in Table 1 (for illustration purposes).

The scores are then used as features to prepare a dataset for employing machine learning classifiers to enhance novelty detection. As exemplified in the table, novelty scores for known classes are relatively higher than those of novel classes. This allows the final classifier to robustly recognize novel input, as explored empirically below.

4.3 Empirical Evaluation

We divided the task into two experiments, i.e., 1) Novelty score calculation 2) Novelty/Known Class classification. In the first experiment, we employ the known classes to train the TM. The TM runs for 100 epochs with a hyperparameter setting of 5000 clauses, a threshold T of 25, and a sensitivity s of 15.0. Then, we use the clauses formed by the trained TM model to calculate the novelty scores for both known and novel classes. We adopt an equal number of examples to gather the novelty score from both known and novel classes. In the second experiment, the novelty score generated from the first experiment is forwarded as input to standard machine learning classifiers, such as DT, KNN, SVM, LR, NB, and MLP to classify whether a text is novel or known.

The experimental results for all datasets are shown in Table 2. As seen, multilayer perceptron (MLP) (hidden layer sizes *100,30* and *ReLU* activation with stochastic gradient descent) is superior for all of the datasets. In our experiments, the 20 Newsgroups and BBC sports datasets yielded better results than the three other data sets, arguably because of the sharp distinctiveness of examples in the known and novel classes. We further believe that the novelty scores are clustered based on known and novel classes; thus, distance-based methods seem effective in classification. We plotted the novelty score of thousand text samples from known and novel classes using our framework, which shows how scores differ significantly for each sample as visualized in Figure 5. Moreover, we can see from the graph that when the texts from the novel and known classes are discriminative, the TM produces distinct novelty scores, thus improving the ability of machine learning classifiers to detect novel texts. In all brevity, we believe that the clauses of a TM capture frequent patterns in the training data, and thus novel samples will reveal themselves by not fitting with a sufficiently large number of clauses. The number of triggered clauses can therefore be utilized to measure novelty score. Also, the clauses formed by trained TM models should only to a small degree trigger on novel classes, producing distinctively low scores.

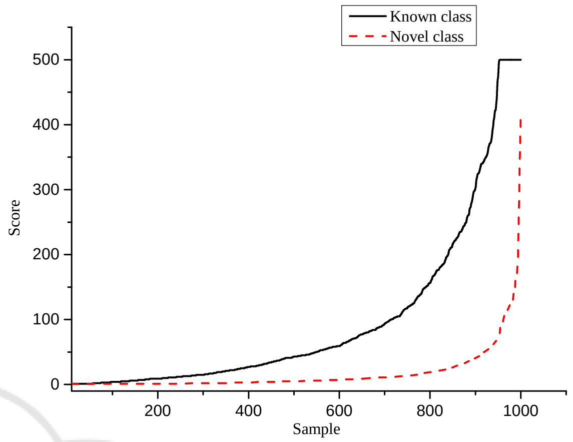

Figure 5: Visualization of differences in novelty score for known and novel classes.

We compared the performance of our TM framework with different clustering and outlier detection algorithms, such as Cluster-based Local Outlier Factor (CBLOF), Feature Bagging (*neighbors, n = 35*), Histogram-base Outlier Detection (HBOS), Isolation Forest, Average KNN, K-Means clustering, and One-class SVM. The evaluation was performed on the same preprocessed datasets for a fair comparison. To make comparison more robust, we preprocessed the data for the baseline algorithms using count vectorizer, term frequency-inverse document frequency (TF-IDF), and Principle component analysis (PCA). Additionally, we utilized the maximum possible outlier fraction (i.e., 0.5) for these methods. The performance comparison is given in Table 3, which shows

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

Algorithms	**20 Newsgroup**	**Spooky action author**	**CMU movie**	**BBC sports**	**WOS**
LOF	52.51 %	50.66 %	48.84 %	47.97 %	55.61 %
Feature Bagging	67.60 %	62.70 %	64.73 %	54.38 %	69.64 %
HBOS	55.03 %	48.55 %	48.57 %	49.53 %	55.09 %
Isolation Forest	52.01 %	48.66%	49.10 %	49.35%	54.70 %
Average KNN	76.35 %	57.76 %	56.21 %	55.54 %	**79.22** %
K-Means clustering	81.00 %	61.30 %	49.20 %	47.70 %	41.31 %
One-class SVM	**83.70 %**	43.56 %	51.94 %	83.53 %	36.32 %
TM framework	82.50 %	**63.15%**	**68.15 %**	**89.47 %**	70.37 %

that our framework surpasses the other algorithms on three of the datasets and performs competitively in the remaining two. However, in datasets like *Web of Science*, where there are many similar words shared between known and novel classes, our method is surprisingly surpassed by the distance-based algorithm (i.e., Average KNN). One-class SVM closely follows the performance of our TM framework, which may be due to its linear structure that prevents overfitting on imbalanced and small datasets.

5 CONCLUSION

In this paper, we studied the problem of novelty detection in multiclass text classification. We proposed a score-based TM framework for novel class detection. We first used the clauses of the TM to produce a novelty score, distinguishing between known and novel classes. Then, a machine learning classifier is adopted for novelty classification using the novelty scores provided by the TM. The experimental results on various datasets demonstrate the effectiveness of our proposed framework. Our future work includes using a large text corpus with multiple classes for experimentation and studying the properties of the novelty score theoretically.

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

REFERENCES

Basu, S., Bilenko, M., and Mooney, R. J. (2004). A probabilistic framework for semi-supervised clustering. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, page 59–68, New York, NY, USA. Association for Computing Machinery.

Bendale, A. and Boult, T. E. (2016). Towards open set deep networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Berge, G. T., Granmo, O.-C., Tveit, T. O., Goodwin, M., Jiao, L., and Matheussen, B. V. (2019). Using the Tsetlin machine to learn human-interpretable rules for high-accuracy text categorization with medical applications. *IEEE Access*, 7:115134–115146.

Chandola, V., Banerjee, A., and Kumar, V. (2009). Anomaly detection: A survey. *ACM Comput. Surv.*, 41(3).

Chow, C. K. (1970). On optimum recognition error and reject tradeoff. *IEEE Trans. Information Theory*, 16:41–46.

Darshana Abeyrathna, K., Granmo, O.-C., Zhang, X., Jiao, L., and Goodwin, M. (2020). The regression Tsetlin machine: A novel approach to interpretable nonlinear regression. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 378(2164):20190165.

Fei, G. and Liu, B. (2015). Social media text classification under negative covariate shift. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2347–2356, Lisbon, Portugal. Association for Computational Linguistics.

Granmo, O.-C. (2018). The Tsetlin machine - A game theoretic bandit driven approach to optimal pattern recognition with propositional logic. *arXiv preprint arXiv:1804.01508*.

Hautamaki, V., Karkkainen, I., and Franti, P. (2004). Outlier detection using k-nearest neighbour graph. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, volume 3, pages 430–433 Vol.3.

Hendrycks, D. and 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.

Kowsari, K., Brown, D., Heidarysafa, M., Meimandi, K., Gerber, M., and Barnes, L. (2017). Hdltex: Hierarchical deep learning for text classification. *2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 364–371.

Pimentel, M. A. F., Clifton, D. A., Clifton, L., and Tarassenko, L. (2014). Review: A review of novelty detection. *Signal Process.*, 99:215–249.

Pincus, R. (1995). Barnett, v., and lewis t.: Outliers in statistical data. 3rd edition. j. wiley & sons 1994, xvii. 582 pp., £49.95. *Biometrical Journal*, 37(2):256–256.

Scheirer, W. J., de Rezende Rocha, A., Sapkota, A., and Boult, T. E. (2013). Toward open set recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(7):1757–1772.

Schölkopf, B., Platt, J., Shawe-Taylor, J., Smola, A., and Williamson, R. (2001). Estimating support of a high-dimensional distribution. *Neural Computation*, 13:1443–1471.

Tsetlin, M. L. (1961). On the behavior of finite automata in random media. *Avtomatika i Telemekhanika*, 22:1345–1354.

Veeramreddy, J., Prasad, V., and Prasad, K. (2011). A review of anomaly based intrusion detection systems. *International Journal of Computer Applications*, 28:26–35.

Yadav, R. K., Jiao, L., Granmo, O.-C., and Goodwin, M. (2021). Human-level interpretable learning for aspect-based sentiment analysis. In *The Thirty-Fifth AAAI Conference on Artificial Intelligence (AAAI-21)*. AAAI.

Yu, Y., Qu, W.-Y., Li, N., and Guo, Z. (2017). Open category classification by adversarial sample generation. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*. International Joint Conferences on Artificial Intelligence Organization.

Zhang, X., Jiao, L., Granmo, O.-C., and Goodwin, M. (2020). On the convergence of Tsetlin machines for the identity-and not operators. *arXiv preprint arXiv:2007.14268*.

Zhang, Y., Meratnia, N., and Havinga, P. (2010). Outlier detection techniques for wireless sensor networks: A survey. *IEEE Communications Surveys Tutorials*, 12(2):159–170.

SCITEPRESS
SCIENCE AND TECHNOLOGY PUBLICATIONS