# Tick Parasitism Classification from Noisy Medical Records

**James O' Neill**[1*], **Danushka Bollegala**[1], **Alan D. Radford**[2] and **PJ Noble**[3]

[1]Department of Computer Science, University of Liverpool
[2]Department of Infection Biology, University of Liverpool
[3]Small Animal Department, University of Liverpool
{danushka.bollegala, alanrad, rtnortle}@liverpool.co.uk

## Abstract

Much of the health information in the medical domain comes in the form of clinical narratives. The rich semantic information contained in these notes can be modeled to make inferences that assist the decision making process for medical practitioners, which is particularly important under time and resource constraints. However, the creation of such assistive tools is made difficult given the ubiquity of misspellings, unsegmented words and morphologically complex or rare medical terms. This reduces the coverage of vocabulary terms present in commonly used pretrained distributed word representations that are passed as input to parametric models that makes such predictions. This paper presents an ensemble architecture that combines in-domain and general word embeddings to overcome these challenges, showing best performance on a binary classification task when compared to various other baselines. We demonstrate our approach in the context of the veterinary domain for the task of identifying tick parasitism from small animals. The best model shows 84.29% test accuracy, showing some improvement over models, which only use pretrained embeddings that are not specifically trained for the medical sub-domain of interest.

## 1 Introduction

Clinical narratives contain important and useful information about the health of a subject. Medical practitioners often have to spend a considerable amount of time reading these notes to make informed decisions, which can be quite laborious. Parametric models can be used to extract information that can assist medical experts in decision-making while reducing this burden. However, spelling mistakes, complex medical terms and rare terms are ubiquitous in such clinical narratives [Roberts *et al.*, 2018].

Tick parasitism (TP) is commonly seen in veterinary patients. Given that ticks can transmit a variety of diseases including important zoonotic disease (e.g. Lyme's disease), tools to screen clinical records for reporting of tick parasitism would be valuable for surveillance of tick activity and subsequent disease and in developing clinical decision support for clinicians.

The aim of this work is to automate annotation of clinical notes from small animal practice for the presence of TP. We are motivated by the fact that using veterinary notes allows us to keep the privacy of the small animals intact while improving the quality of assistive diagnosis, and in turn, medication. This is something that is not easily achieved outside of veterinary practices.

We are able to take advantage of small animal clinical records collected through the Small Animal Veterinary Surveillance Network (SAVSNET). Narratives in the SAVSNET corpus are currently screened using simple regular-expressions to identify mentions of the word 'tick'. These mentions may refer to a tick present on a pet or simply to discussion of tick prevention and need to be manually annotated accordingly.

We propose a dynamic ensemble neural network that learns to classify TP from imprecise clinical narratives. Our approach incorporates fine-tuned in-domain word embeddings and domain-agnostic pretrained embeddings to improve classification performance. A parameter is used to learn a weighted combination of each embedding in the overall classification.

**Contributions** Our contributions are as follows:

1. A novel application of text classification for noisy clinical narratives, specifically in the veterinary domain.
2. To the best of our knowledge, the first known attempt to predict TP on animals from textual descriptions.
3. An ensemble approach that combines domain agnostic and domain specific representations ($n$-gram character, subword and word vectors) to recurrent neural network architecture and strong baselines of both non-neural network and neural network classifiers.

## 2 Background

### 2.1 Tick-borne disease

Tick-borne diseases (TBD) are caused by a variety of pathogens (bacteria, viruses, rickettsia and protozoa) which are transmitted through tick-bites. Identifying and preventing TBD from spreading is difficult given that ticks have a

*james.o-neill@liverpool.ac.uk

wide geographic range and are highly adaptive to changing environments allowing this range to increase. Globally, significant TBD include tularaemia and rocky mountain spotted fever. In a recent study using electronic health records from cats and dogs, TP was seen most commonly in the south-central region of England with a peak activity in summer and a smaller peak in cats in Autumn [Tulloch *et al.*, 2017].

### 2.2 Economic Impact of Tick Parasitism

A recent study from Germany highlighted a potential cost of $>$ 30M Euros resulting from Lyme borreliosis alone [Lohr *et al.*, 2015]. In addition, recent work reviewed the impact of TP for production animals that are required to meet standard health conditions [Giraldo-Ríos and Betancur, 2018], reducing the survival rate of the animal and the production of meat, milk, eggs etc. (not to mention costs incurred for treatment).

### 2.3 Small Animal Ticks in the UK

In this work, we focus on small animals within the United Kingdom (UK). In the past decade, there has been an ongoing effort to collect ticks by the public, veterinary health agencies and practitioners within the UK as part of the Tick Surveillance Scheme and *The Big Tick Project* [Jameson and Medlock, 2011; Abdullah *et al.*, 2016] in an effort to identify various tick species (predominantly from companion animals) and their locality across various regions in the UK. Although these projects demonstrate the viability of nationwide surveillance programs to monitor tick species, we argue that the requirement for active manual participation by contributors and lack of automation along with inconsistent/aperiodic data-collection present barriers to participation and to collation of representative data in the long term. Using an automated system to screen clinical notes for tick parasitism will enhance tick-surveillance. Furthermore this work will provide a model for systems that might be used for clinical note summarisation and, subsequently, for clinical decision-making support.

## 3 Related Research

Most previous work on medical text classification has focused on cleaner text, which are extracted from more formal registers. Although, there has been recent work that has explored classification on health records and notes which we include below.

A key challenge is making use of information rich notes while reducing redundancy contained in the corpus due to the copying of notes which can lead to a degradation in performance. In the context of topic modelling, prior work has applied a variant of Latent Dirichlet Allocation (LDA) to patient record notes [Cohen *et al.*, 2014], which they refer to as Red-LDA. Red-LDA removes this redundancy and improves on topic coherence and qualitative assessments in comparison to standard LDA.

Yi and Beheshti used a hidden Markov model for classifying medical documents that incorporates prior knowledge in the form of medical subject headings.

Iyer *et al.* have performed text mining on clinical text for drug-event recognition from 50 million clinical notes to create a timeline of adverse drug event mentions per patient. This was used to identify drug-drug-event associations for 1,165 drugs and 14 events.

Other methods that do not use text have relied on geospatial data for modelling tick presence [Swart *et al.*, 2014]. The model predicted the presence of ticks within a 1 $km^2$ grid from field data using satellite-based methodology with Bayesian priors chosen over landscape types. Ticks were estimated for 54% land cover, finding a 37% presence from all 677 coordinates sampled.

Lastly, recent work has performed medical text classification with convolutional neural networks (CNN) with Word2Vec as input [Hughes *et al.*, 2017]. This was shown to outperform Logistic Regression that uses Doc2Vec representations or Bag of Words (BOW) based Word2Vec approaches. They use k-means clustering (k=1000) to generate a feature vocabulary that is used to generate a soft assignment BOW histogram for each sentence. These features were then used as input to the Logistic regression model. The BOW with Logistic Regression yielded the best baseline results with 51% test accuracy, which was still 17% percentage points lower than the proposed CNN model, which uses Word2Vec. However, there is no use of recurrent models for preserving the sequential nature of text.

## 4 Methodology

### 4.1 Models

**Model Configurations**

For all the below models we use the Binary Cross Entropy (BCE) loss, with a learning rate $\eta = 0.001$ and Adaptive Momentum (`adam`) for optimization. Dropout is used for regularization on all layers (not including the input) with a dropout rate $p_d = 0.2$. The pretrained embeddings used are fixed throughout training (i.e no gradient udpates). The batch size $|x_s| = 200$ for each model. Given the class imbalance, we choose to weight the losses inversely proportional to the frequency of each class during each mini-batch update. This avoids other alternative approach such as sampling methods [Chawla *et al.*, 2002] with little cost.

**Convolutional Neural Network**

We test CNNs for text classification, which too have been used in the medical domain (as aforementioned [Hughes *et al.*, 2017]), first proposed by Kim et al. (2014). The CNN model uses 100 $2d$ filters each for kernels of size $(2, 300)$ and $(3, 300)$ for character n-gram embeddings (GloVe), subword embeddings (FastText)[1] and word embeddings (Word2Vec [2]), all of which are $d_w = 300$.

Our motivation for using FastText is that subword embeddings are first learned to create word embeddings and therefore mitigates the problem of misspellings, while they are also used to deal with out-of-vocabulary terms (a new word is likely to share some subwords with the words already in the vocabulary). ReLU activations are used with $1d$ max pooling after each layer followed by a concatenation of the last layer features.

[1] pretrained-fasttext:https://fasttext.cc/docs/en/crawl-vectors.html

[2] pretrained-skipgram: https://code.google.com/archive/p/word2vec/

**Gated Recurrent Network**
As a second baseline approach, we test recurrent architectures with memory networks to preserve any non-local dependencies between terms, which we would expect to further improve performance. The Gated Recurrent (GRU) model uses 2-hidden layers where the last output layer $(1, 300)$ is passed to a dense layer. The weights are initialized using Xavier normalisation [Glorot and Bengio, 2010] ($\mu$=0,$\sigma$=0.01) and $tanh$ activation units are used.

**Ensembled Feature Approach**
In the above two aforementioned models, the challenge of poorly typed notes is addressed using $n$-gram character vectors, sub-word vectors and word vectors. In the ensemble approach shown in Figure 1, we combine the latter two by concatenating both final hidden layer encodings (red) and pass it to a dense layer (green) before making the final prediction $\hat{y}$. This allows for interaction terms among both sentence encodings created by sub-word and word vectors. For regularization, we also use dropout in this dense layer with a rate $p_d = 0.5$, while other layers are kept at $p_d = 0.2$ as previously mentioned.

We also evaluate this approach when combining in-domain word embeddings trained on the clinical narratives and pretrained embeddings. This allows us to systematically combine the benefits of both vector representation by simply adding a dense layer that acts a weighted combination of both sentence embeddings to produce the final encoding. In a similar fashion we carry this ensemble method out for the previously mentioned 2-hidden layer Convolutional Neural Network.

Below we summarize the steps in Equation 1 where $E$ is an embedding matrix, $\tilde{E}$ is a fine-tunable $E$ and $\tilde{E}_S, \tilde{E}_W$ are both subword and word pretrained embeddings repsectively, which are not updated during training. The input tokens $\mathbf{w} \in \mathbb{R}^n$ are passed to the embedding matrix $E \in \mathbb{R}^{n \times d}$ which are then transformed with parameters $W \in \mathbb{R}^{d \times m}$ and $b, h, \Theta \in \mathbb{R}^{m \times 1}$.
Equation 1 shows how $p \in [0, 1]$ controls the tradeoff between the tunable task-specific embeddings and static pretrained subword and word embeddings, acting as a weighted average between both input representations. Here $\oplus$ signifies a concatenation. This is followed by a linear layer with a tanh activation unit, which results in $z$ that we use as input to our model. Note, that in this configuration, we perform ensembling at the input with very few additional parameters.

$$\begin{aligned} \mathbf{E} &= p\tilde{\mathbf{E}}(w) \oplus \big(1-p\big)\big(\bar{\mathbf{E}}_{\mathbf{S}}(w) \oplus \bar{\mathbf{E}}_{\mathbf{W}}(w)\big) \\ \mathbf{z} &= \tanh\big(\langle \mathbf{E}, W\rangle + b\big) \end{aligned} \tag{1}$$

During training at a timestep $t \in T$ we then pass word $w_t$ to obtain $E_t^w$ and subsequently $z_t^w$ which is then passed to the GRU shown in Equation 2. Here $h_{t-1}$ is the output of the GRU hidden state from the previous timestep and $h_T^L$ denotes the output of the last hidden layer $L$ for the hidden state at time $T$.

$$\mathbf{h_t} = \text{GRU}\big(z_t^w, h_{t-1}\big), \quad \hat{y} = \phi(\langle h_T^L, \Theta\rangle\big) \tag{2}$$

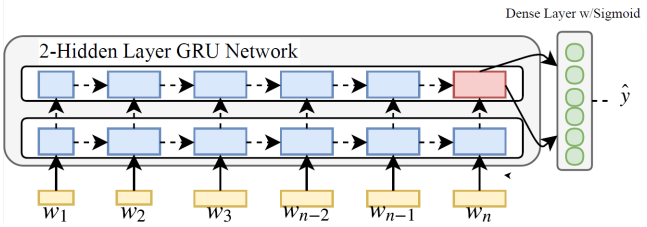


Figure 1: GRU-based Ensemble Architecture (red corresponds to the last hidden state vector that outputs embedding for both in-domain and large pretrained embedding inputs)

In contrast, we also consider passing each embedding separately and instead perform ensembling at the output as shown in Equation 3 (and shown in Figure 1), in which case $\Theta \in \mathbb{R}^{3m}$. In our experiments, we found the latter of these two approaches to outperform the former.

$$\hat{y} = \phi(\langle \tilde{\mathbf{h}}_\mathbf{t} \oplus \bar{\mathbf{h}}_\mathbf{t}^\mathbf{S} \oplus \bar{\mathbf{h}}_\mathbf{t}^\mathbf{W}, \Theta\rangle\big) \tag{3}$$

Binary Cross Entropy (BCE) loss is then used as the objective, as shown in Equation 4 where $N$ is the number of samples in a given mini-batch update.

$$\ell(y, \hat{y}) = -\frac{1}{N}\sum_{i=1}^{N} y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i) \tag{4}$$

## 5 Experimental Data

**Collected Dataset** We demonstrate our method on the task of identifying TP in clinical records from animals, which to our knowledge is a novel application of text-based machine learning for this problem. The *Small Animal Veterinary Surveillance Network* (SAVSNET) dataset contains approximately 3.5 million records. The health records are submitted to SAVSNET at the end of consultations by a veterinary surgeon or nurse that list why the animal was brought into the veterinary practice[3].

A dataset of narratives containing the word tick (identified using the case-insensitive regex '\\W tick \\W') was identified. This comprised 27075 narratives which had been read and annotated for whether the veterinary surgeon had noted TP (the presence of a tick on the patient in the consulting room). 6,529 records were annotated positive for TP. A further set of 1.2 million randomly selected records with no mention of *tick* (which were, therefore, considered to be negative for TP but were not manually annotated) were also added to the dataset. We use an 80-20 split for training and testing and perform 5-fold cross validation on the training data.

## 6 Results

**Exploratory Analysis** Figure 2 shows the log-frequency for a range of sentence lengths for all clinical narratives. Each narrative can contain anywhere from one relatively long sentence to an entire paragraph. Therefore, we split the sentences

[3]see here for more information: https://www.liverpool.ac.uk/savsnet/

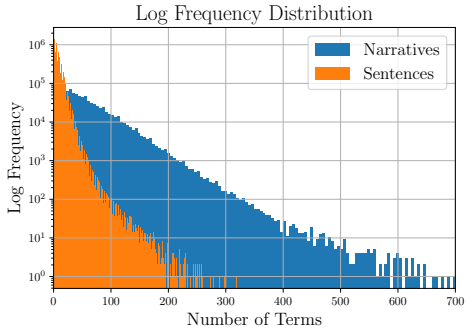


Figure 2: Sentence & Narrative Length Distribution

into separate instances for training during classification. This is because encoding long paragraphs becomes too difficult for the RNN to preserve all the information in a single encoding. Hence, when using an RNN classifier, we average over the encodings of each sentence within a single narrative before passing it to the last fully connected layer.

**Non-ANN Classification Results** Table 1 shows the results of non-neural network based models that include ensemble-based (Random Forest and Gradient Boosting), large margin methods (Support Vector Machines) and kernel-based methods (Gaussian Processes). All models use a combination of tf-idf scores and unigram frequencies. We find that, in general, most of these models perform similarly, with Support Vector Machines with a Radial Basis Function slightly outperforming the alternative models. These methods are fast and require little memory as these features are essentially counts (unigram) and normalizations thereof (tf-idf).

**Neural Network Classification Results** Table 2 shows the classification results when using pretrained embeddings. Since classes are imbalanced, 72% accuracy is achieved if the model only predicts the absence of TP. For this reason it should be pointed out that although the performance seems relatively accurate, it is a particularly challenging to mitigate false negatives.

The first section are the model results of CNN models with pretrained GloVe n-gram character vectors, FastText subword vectors and `skipgram` word embeddings trained on GoogleNews. The second section are the same input but instead using GRU networks. In the third section "T" denotes vectors trained on clinical narratives. Lastly, the ensemble models use a combination of both pretrained `fasttext` vectors and

| | Train (10-Fold CV) | | | Test | | |
|---|---|---|---|---|---|---|
| Models | Acc. | AUC | F1 | Acc. | AUC | F1 |
| SVM (RBF) | 87.14 | 0.89 | 0.87 | 84.03 | 0.82 | 0.84 |
| SVM (Linear) | 85.91 | 0.84 | 0.79 | 81.69 | 0.82 | 0.86 |
| Random Forest | 81.62 | 0.83 | 0.80 | 80.49 | 0.83 | 0.80 |
| Gradient Boosting | 85.40 | 0.83 | 0.85 | 84.34 | 0.85 | 0.85 |
| Gaussian Process | 86.92 | 0.81 | 0.83 | 82.28 | 0.81 | 0.82 |

Table 1: TP (Non-Neural Network) Classification Results

| | Train | | | Test | | |
|---|---|---|---|---|---|---|
| Models | Acc. | AUC | F1 | Acc. | AUC | F1 |
| Char-CNN | 78.13 | 0.78 | 77.29 | 70.27 | 0.69 | 68.90 |
| SubWord-CNN | 85.84 | 0.89 | 84.38 | 82.61 | 0.81 | 81.15 |
| Word-CNN | 84.49 | 0.86 | 83.78 | 80.44 | 0.79 | 80.13 |
| Char-GRU | 79.13 | 0.79 | 77.37 | 74.02 | 0.73 | 74.92 |
| SubWord-GRU | 87.24 | 0.90 | 87.29 | 83.47 | 0.84 | 82.97 |
| Word-RNN | 84.20 | 0.83 | 84.88 | 79.68 | 0.77 | 79.02 |
| T-Char-GRU | 81.60 | 0.80 | 80.18 | 74.47 | 0.74 | 73.89 |
| T-SubWord-GRU | 84.78 | 0.83 | 82.69 | 76.11 | 0.75 | 75.45 |
| T-Word-RNN | 86.98 | 0.89 | 86.01 | 76.46 | 0.75 | 76.28 |
| Ensemble-CNN | 86.11 | 0.89 | 85.34 | 83.07 | 0.83 | 82.73 |
| Ensemble-GRU | 88.63 | 0.91 | 88.51 | 84.29 | 0.82 | 85.20 |

Table 2: TP Neural Network Classification Results

`fasttext` vectors trained on the clinical narratives, as discussed in the previous section. We find best results are obtained using the GRU ensemble based on the overall test performance (shaded).

## 7 Conclusion & Future Work

We proposed an ensemble-based neural network to overcome the difficulties in inference when dealing with noisy medical data in the form of veterinary clinical notes. Similar baselines also show good performance, particularly when used with subword vectors. Recurrent models in general show improvements over convolutional neural networks. These models can be used to reduce manual labor for medical practitioners by assisting in the decision making process even when misspellings are common.

The challenge of class balancing without a degradation in overall performance is a problem we defer to future work. Specifically, we plan to investigate other strategies to address imbalanced classes in the presence of noisy medical texts using data-augmentation strategies. One such approach involves the use of generative modeling of sentence embeddings to upsample the minority class with the goal of reducing false positives, but more importantly to reduce true negatives.

## 8 Acknowledgements

SAVSNET is based at the University of Liverpool. It is currently funded by the Biotechnology and Biological Sciences Research Council. The SAVSNET team is grateful to the veterinary practices and diagnostic laboratories that provide health data and without whose support this research would not be possible.

## References

[Abdullah *et al.*, 2016] Swaid Abdullah, Chris Helps, Severine Tasker, Hannah Newbury, and Richard Wall. Ticks infesting domestic dogs in the uk: a large-scale surveillance programme. *Parasites & vectors*, 9(1):391, 2016.

[Chawla *et al.*, 2002] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: syn-

thetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002.

[Cohen *et al.*, 2014] Raphael Cohen, Iddo Aviram, Michael Elhadad, and Noémie Elhadad. Redundancy-aware topic modeling for patient record notes. *PloS one*, 9(2):e87555, 2014.

[Giraldo-Ríos and Betancur, 2018] Cristian Giraldo-Ríos and Oscar Betancur. Economic and health impact of the ticks in production animals. In *Ticks and Tick-Borne Pathogens*. IntechOpen, 2018.

[Glorot and Bengio, 2010] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256, 2010.

[Hughes *et al.*, 2017] Mark Hughes, I Li, Spyros Kotoulas, and Toyotaro Suzumura. Medical text classification using convolutional neural networks. *Stud Health Technol Inform*, 235:246–50, 2017.

[Iyer *et al.*, 2013] Srinivasan V Iyer, Rave Harpaz, Paea LePendu, Anna Bauer-Mehren, and Nigam H Shah. Mining clinical text for signals of adverse drug-drug interactions. *Journal of the American Medical Informatics Association*, 21(2):353–362, 2013.

[Jameson and Medlock, 2011] Lisa J Jameson and Jolyon M Medlock. Tick surveillance in great britain. *Vector-Borne and Zoonotic Diseases*, 11(4):403–412, 2011.

[Kim, 2014] Yoon Kim. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*, 2014.

[Lohr *et al.*, 2015] B Lohr, I Müller, M Mai, DE Norris, O Schöffski, and K-P Hunfeld. Epidemiology and cost of hospital care for lyme borreliosis in germany: lessons from a health care utilization database analysis. *Ticks and tick-borne diseases*, 6(1):56–62, 2015.

[Roberts *et al.*, 2018] Kirk Roberts, Yuqi Si, Anshul Gandhi, and Elmer Bernstam. A framenet for cancer information in clinical narratives: Schema and annotation. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*. European Language Resource Association, 2018.

[Swart *et al.*, 2014] Arno Swart, Adolfo Ibañez-Justicia, Jan Buijs, Sip E van Wieren, Tim R Hofmeester, Hein Sprong, and Katsuhisa Takumi. Predicting tick presence by environmental risk mapping. *Frontiers in public health*, 2:238, 2014.

[Tulloch *et al.*, 2017] JSP Tulloch, L McGinley, F Sánchez-Vizcaíno, JM Medlock, and AD Radford. The passive surveillance of ticks using companion animal electronic health records. *Epidemiology & Infection*, 145(10):2020–2029, 2017.

[Yi and Beheshti, 2009] Kwan Yi and Jamshid Beheshti. A hidden markov model-based text classification of medical documents. *Journal of Information Science*, 35(1):67–81, 2009.