Bringing replication and reproduction together with generalisability in NLP: Three reproduction studies for Target Dependent Sentiment Analysis

Andrew Moore and Paul Rayson
School of Computing and Communications, Lancaster University, Lancaster, UK
initial.surname@lancaster.ac.uk

Abstract

Lack of repeatability and generalisability are two significant threats to continuing scientific development in Natural Language Processing. Language models and learning methods are so complex that scientific conference papers no longer contain enough space for the technical depth required for replication or reproduction. Taking Target Dependent Sentiment Analysis as a case study, we show how recent work in the field has not consistently released code, or described settings for learning methods in enough detail, and lacks comparability and generalisability in train, test or validation data. To investigate generalisability and to enable state of the art comparative evaluations, we carry out the first reproduction studies of three groups of complementary methods and perform the first large-scale mass evaluation on six different English datasets. Reflecting on our experiences, we recommend that future replication or reproduction experiments should always consider a variety of datasets alongside documenting and releasing their methods and published code in order to minimise the barriers to both repeatability and generalisability. We have released our code with a model zoo on GitHub with Jupyter Notebooks to aid understanding and full documentation, and we recommend that others do the same with their papers at submission time through an anonymised GitHub account.

1 Introduction

Repeatable (replicable and/or reproducible[1]) experimentation is a core tenet of the scientific endeavour. In Natural Language Processing (NLP) research as in other areas, this requires three crucial components: (a) published methods described in sufficient detail (b) a working code base and (c) open dataset(s) to permit training, testing and validation to be reproduced and generalised. In the cognate sub-discipline of corpus linguistics, releasing textual datasets has been a defining feature of the community for many years, enabling multiple comparative experiments to be conducted on a stable basis since the core underlying corpora are community resources. In NLP, with methods becoming increasingly complex with the use of machine learning and deep learning approaches, it is often difficult to describe all settings and configurations in enough detail without releasing code. The work described in this paper emerged from recent efforts at our research centre to reimplement other's work across a number of topics (e.g. text reuse, identity resolution and sentiment analysis) where previously published methods were not easily repeatable because of missing or broken code or dependencies, and/or where methods were not sufficiently well described to enable reproduction. We focus on one sub-area of sentiment analysis to illustrate the extent of these problems, along with our initial recommendations and contributions to address the issues.

The area of Target Dependent Sentiment Analysis (TDSA) and NLP in general has been growing rapidly in the last few years due to new neural network methods that require no feature engineering. However it is difficult to keep track of the state of the art as new models are tested on different datasets, thus preventing true comparative evaluations. This is best shown by table 1 where many approaches

[1]We follow the definitions in Antske Fokkens' guest blog post "replication (obtaining the same results using the same experiment) as well as reproduction (reach the same conclusion through different means)" from http://coling2018.org/slowly-growing-offspring-zigglebottom-anno-2017-guest-post/

are evaluated on the SemEval dataset (Pontiki et al., 2014) but not all. Datasets can vary by domain (e.g. product), type (social media, review), or medium (written or spoken), and to date there has been no comparative evaluation of methods from these multiple classes. Our primary and secondary contributions therefore, are to carry out the first study that reports results across all three different dataset classes, and to release a open source code framework implementing three complementary groups of TDSA methods.

In terms of reproducibility via code release, recent TDSA papers have generally been very good with regards to publishing code alongside their papers (Mitchell et al., 2013; Zhang et al., 2015; Zhang et al., 2016; Liu and Zhang, 2017; Marrese-Taylor et al., 2017; Wang et al., 2017) but other papers have not released code (Wang et al., 2016; Tay et al., 2017). In some cases, the code was initially made available, then removed, and is now back online (Tang et al., 2016a). Unfortunately, in some cases even when code has been published, different results have been obtained relative to the original paper. This can be seen when Chen et al. (2017) used the code and embeddings in Tang et al. (2016b) they observe different results. Similarly, when others (Tay et al., 2017; Chen et al., 2017) attempt to replicate the experiments of Tang et al. (2016a) they also produce different results to the original authors. Our observations within this one sub-field motivates the need to investigate further and understand how such problems can be avoided in the future. In some cases, when code has been released, it is difficult to use which could explain why the results were not reproduced. Of course, we would not expect researchers to produce industrial strength code, or provide continuing free ongoing support for multiple years after publication, but the situation is clearly problematic for the development of the new field in general.

In this paper, we therefore reproduce three papers chosen as they employ widely differing methods: Neural Pooling (NP) (Vo and Zhang, 2015), NP with dependency parsing (Wang et al., 2017), and RNN (Tang et al., 2016a), as well as having been applied largely to different datasets. At the end of the paper, we reflect on bringing together elements of repeatability and generalisability which we find are crucial to NLP and data science based disciplines more widely to enable others to make use of the science created.

	Datasets					
Methods	1	2	3	4	5	6
Mitchell et al. (2013)			✓			
Kiritchenko et al. (2014)				✓		
Dong et al. (2014)	✓					
Vo and Zhang (2015)	✓	✓	✓			
Zhang et al. (2015)			✓			
Zhang et al. (2016)	✓	✓	✓			
Tang et al. (2016a)	✓			✓		
Tang et al. (2016b)				✓		
Wang et al. (2016)				✓		
Chen et al. (2017)	✓			✓		
Liu and Zhang (2017)	✓	✓	✓			
Wang et al. (2017)	✓				✓	
Marrese-Taylor et al. (2017)				✓		✓
1=Dong et al. (2014), 2=Wilson (2008), 3=Mitchell et al. (2013), 4=Pontiki et al. (2014), 5=Wang et al. (2017), 6=Marrese-Taylor et al. (2017)						

Table 1: Methods and Datasets

2 Related work

Reproducibility and replicability have long been key elements of the scientific method, but have been gaining renewed prominence recently across a number of disciplines with attention being given to a 're-producibility crisis'. For example, in pharmaceutical research, as little as 20-25% of papers were found to be replicable (Prinz et al., 2011). The problem has also been recognised in computer science in general (Collberg and Proebsting, 2016). Reproducibility and replicability have been researched for sometime

in Information Retrieval (IR) since the Grid@CLEF pilot track (Ferro and Harman, 2009). The aim was to create a 'grid of points' where a point defined the performance of a particular IR system using certain pre-processing techniques on a defined dataset. Louridas and Gousios (2012) looked at reproducibility in Software Engineering after trying to replicate another authors results and concluded with a list of requirements for papers to be reproducible: (a) All data related to the paper, (b) All code required to reproduce the paper and (c) Documentation for the code and data. Fokkens et al. (2013) looked at reproducibility in WordNet similarity and Named Entity Recognition finding five key aspects that cause experimental variation and therefore need to be clearly stated: (a) pre-processing, (b) experimental setup, (c) versioning, (d) system output, (e) system variation. In Twitter sentiment analysis, Sygkounas et al. (2016) stated the need for using the same library versions and datasets when replicating work.

Different methods of releasing datasets and code have been suggested. Ferro and Harman (2009) defined a framework (CIRCO) that enforces a pre-processing pipeline where data can be extracted at each stage therefore facilitating a validation step. They stated a mechanism for storing results, dataset and pre-processed data[2]. Louridas and Gousios (2012) suggested the use of a virtual machine alongside papers to bundle the data and code together, while most state the advantages of releasing source code (Fokkens et al., 2013; Potthast et al., 2016; Sygkounas et al., 2016). The act of reproducing or replicating results is not just for validating research but to also show how it can be improved. Ferro and Silvello (2016) followed up their initial research and were able to analyse which pre-processing techniques were important on a French monolingual dataset and how the different techniques affected each other given an IR system. Fokkens et al. (2013) showed how changes in the five key aspects affected results.

The closest related work to our reproducibility study is that of Marrese-Taylor and Matsuo (2017) which they replicate three different syntactic based aspect extraction methods. They found that parameter tuning was very important however using different pre-processing pipelines such as Stanford's CoreNLP did not have a consistent effect on the results. They found that the methods stated in the original papers are not detailed enough to replicate the study as evidenced by their large results differential.

Dashtipour et al. (2016) undertook a replication study in sentiment prediction, however this was at the document level and on different datasets and languages to the originals. In other areas of (aspect-based) sentiment analysis, releasing code for published systems has not been a high priority, e.g. in SemEval 2016 task 5 (Pontiki et al., 2016) only 1 out of 21 papers released their source code. In IR, specific reproducible research tracks have been created[3] and we are pleased to see the same happening at COLING 2018[4].

Turning now to the focus of our investigations, Target Dependent sentiment analysis (TDSA) research (Nasukawa and Yi, 2003) arose as an extension to the coarse grained analysis of document level sentiment analysis (Pang et al., 2002; Turney, 2002). Since its inception, papers have applied different methods such as feature based (Kiritchenko et al., 2014), Recursive Neural Networks (RecNN) (Dong et al., 2014), Recurrent Neural Networks (RNN) (Tang et al., 2016a), attention applied to RNN (Wang et al., 2016; Chen et al., 2017; Tay et al., 2017), Neural Pooling (NP) (Vo and Zhang, 2015; Wang et al., 2017), RNN combined with NP (Zhang et al., 2016), and attention based neural networks (Tang et al., 2016b). Others have tackled TDSA as a joint task with target extraction, thus treating it as a sequence labelling problem. Mitchell et al. (2013) carried out this task using Conditional Random Fields (CRF), and this work was then extended using a neural CRF (Zhang et al., 2015). Both approaches found that combining the two tasks did not improve results compared to treating the two tasks separately, apart from when considering POS and NEG when the joint task performs better. Finally, Marrese-Taylor et al. (2017) created an attention RNN for this task which was evaluated on two very different datasets containing written and spoken (video-based) reviews where the domain adaptation between the two shows some promise. Overall, within the field of sentiment analysis there are other granularities such as sentence level (Socher et al., 2013), topic (Augenstein et al., 2018), and aspect (Wang et al., 2016; Tay et al., 2017). Aspect-level sentiment analysis relates to identifying the sentiment of (potentially multiple) topics in the

[2] http://direct.dei.unipd.it/

[3] http://ecir2016.dei.unipd.it/call_for_papers.html

[4] http://coling2018.org/

same text although this can be seen as a similar task to TDSA. However the clear distinction between aspect and TDSA is that TDSA requires the target to be mentioned in the text itself while aspect-level employs a conceptual category with potentially multiple related instantiations in the text.

Tang et al. (2016a) created a Target Dependent LSTM (TDLSTM) which encompassed two LSTMs either side of the target word, then improved the model by concatenating the target vector to the input embeddings to create a Target Connected LSTM (TCLSTM). Adding attention has become very popular recently. Tang et al. (2016b) showed the speed and accuracy improvements of using multiple attention layers only over LSTM based methods, however they found that it could not model complex sentences e.g. negations. Liu and Zhang (2017) showed that adding attention to a Bi-directional LSTM (BLSTM) improves the results as it takes the importance of each word into account with respect to the target. Chen et al. (2017) also combined a BLSTM and attention, however they used multiple attention layers and combined the results using a Gated Recurrent Unit (GRU) which they called Recurrent Attention on Memory (RAM), and they found this method to allow models to better understand more complex sentiment for each comparison. Vo and Zhang (2015) used neural pooling features e.g. max, min, etc of the word embeddings of the left and right context of the target word, the target itself, and the whole Tweet. They inputted the features into a linear SVM, and showed the importance of using the left and right context for the first time. They found in their study that using a combination of Word2Vec embeddings and sentiment embeddings (Tang et al., 2014) performed best alongside using sentiment lexicons to filter the embedding space. Other studies have adopted more linguistic approaches. Wang et al. (2017) extended the work of Vo and Zhang (2015) by using the dependency linked words from the target. Dong et al. (2014) used the dependency tree to create a Recursive Neural Network (RecNN) inspired by Socher et al. (2013) but compared to Socher et al. (2013) they also utilised the dependency tags to create an Adaptive RecNN (ARecNN).

Critically, the methods reported above have not been applied to the same datasets, therefore a true comparative evaluation between the different methods is somewhat difficult. This has serious implications for generalisability of methods. We correct that limitation in our study. There are two papers taking a similar approach to our work in terms of generalisability although they do not combine them with the reproduction issues that we highlight. First, Chen et al. (2017) compared results across SemEval's laptop and restaurant reviews in English (Pontiki et al., 2014), a Twitter dataset (Dong et al., 2014) and their own Chinese news comments dataset. They did perform a comparison across different languages, domains, corpora types, and different methods; SVM with features (Kiritchenko et al., 2014), Rec-NN (Dong et al., 2014), TDLSTM (Tang et al., 2016a), Memory Neural Network (MNet) (Tang et al., 2016b) and their own attention method. However, the Chinese dataset was not released, and the methods were not compared across all datasets. By contrast, we compare all methods across all datasets, using techniques that are not just from the Recurrent Neural Network (RNN) family. A second paper, by Barnes et al. (2017) compares seven approaches to (document and sentence level) sentiment analysis on six benchmark datasets, but does not systematically explore reproduction issues as we do in our paper.

3 Datasets used in our experiments

We are evaluating our models over six different English datasets deliberately chosen to represent a range of domains, types and mediums. As highlighted above, previous papers tend to only carry out evaluations on one or two datasets which limits the generalisability of their results. In this paper, we do not consider the quality or inter-annotator agreement levels of these datasets but it has been noted that some datasets may have issues here. For example, Pavlopoulos and Androutsopoulos (2014) point out that the Hu and Liu (2004) dataset does not state their inter-annotator agreement scores nor do they have aspect terms that express neutral opinion.

We only use a subset of the English datasets available. For two reasons. First, the time it takes to write parsers and run the models. Second, we only used datasets that contain three distinct sentiments (Wilson (2008) only has two). From the datasets we have used, we have only had issue with parsing Wang et al. (2017) where the annotations for the first set of the data contains the target span but the second set does not. Thus making it impossible to use the second set of annotation and forcing us to

only use a subset of the dataset. An as example of this: "Got rid of bureaucrats 'and we put that money, into 9000 more doctors and nurses'... to turn the doctors into bureaucrats#BattleForNumber10" in that Tweet 'bureaucrats' was annotated as negative but it does not state if it was the first or second instance of 'bureaucrats' since it does not use target spans. As we can see from table 2, generally the social media datasets (Twitter and YouTube) contain more targets per sentence with the exception of Dong et al. (2014) and Mitchell et al. (2013). The only dataset that has a small difference between the number of unique sentiments per sentence is the Wang et al. (2017) election dataset.

Lastly we create training and test splits for the YouTuBean (Marrese-Taylor et al., 2017) and Mitchell (Mitchell et al., 2013) datasets as they were both evaluated originally using cross validation. These splits are reproducible using the code that we are open sourcing.

Dataset	DO	T	Size	M	ATS	Uniq	AVG Len	S1	S2	S3
SemEval 14 L	L	RE	2951	W	1.58	1295	18.57	81.09	17.62	1.29
SemEval 14 R	R	RE	4722	W	1.83	1630	17.25	75.26	22.94	1.80
Mitchel	G	S	3288	W	1.22	2507	18.02	90.48	9.43	0.09
Dong Twitter	G	S	6940	W	1.00	145	17.37	100.00	0.00	0.00
Election Twitter	P	S	11899	W	2.94	2190	21.68	44.50	46.72	8.78
YouTuBean	MP	RE/S	798	SP	2.07	522	22.53	81.45	18.17	0.38

L=Laptop, R=Restaurant, P=Politics, MP=Mobile Phones, G=General, T=Type, RE=Review, S=Social Media, ATS=Average targets per sentence, Uniq=No. unique targets, AVG len=Average sentence length per target, S1=1 distinct sentiment per sentence, S2=2 distinct sentiments per sentence, S3=3 distinct sentiments per sentence, DO=Domain, M=Medium, W=Written, SP=Spoken

Table 2: Dataset Statistics

4 Reproduction studies

In the following subsections, we present the three different methods that we are reproducing and how their results differ from the original analysis. In all of the experiments below, we lower case all text and tokenise using Twokenizer (Gimpel et al., 2011). This was done as the datasets originate from Twitter and this pre-processing method was to some extent stated in Vo and Zhang (2015) and assumed to be used across the others as they do not explicitly state how they pre-process in the papers.

4.1 Reproduction of Vo and Zhang (2015)

Vo and Zhang (2015) created the first NP method for TDSA. It takes the word vectors of the left, right, target word, and full tweet/sentence/text contexts and performs max, min, average, standard deviation, and product pooling over these contexts to create a feature vector as input to the Support Vector Machine (SVM) classifier. This feature vector is in affect an automatic feature extractor. They created four different methods: 1. **Target-ind** uses only the full tweet context, 2. **Target-dep-** uses left, right, and target contexts, 3. **Target-dep** Uses both features of **Target-ind** and **Target-dep-**, and 4. **Target-dep+** Uses the features of **Target-dep** and adds two additional contexts left and right sentiment (LS & RS) contexts where only the words within a specified lexicon are kept and the rest of the words are zero vectors. All of their experiments are performed on Dong et al. (2014) Twitter data set. For each of the experiments below we used the following configurations unless otherwise stated: we performed 5 fold stratified cross validation, features are scaled using Max Min scaling before inputting into the SVM, and used the respective C-Values for the SVM stated in the paper for each of the models.

One major difficulty with the description of the method in the paper and re-implementation is handling the same target multiple appearances issue as originally raised by Wang et al. (2017). As the method requires context with regards to the target word, if there is more than one appearance of the target word then the method does not specify which to use. We therefore took the approach of Wang et al. (2017) and found all of the features for each appearance and performed median pooling over features. This change could explain the subtle differences between the results we report and those of the original paper.

4.1.1 Sentiment Lexicons

Vo and Zhang (2015) used three different sentiment lexicons: MPQA[5] (Wilson et al., 2005), NRC[6] (Mohammad and Turney, 2010), and HL[7] (Hu and Liu, 2004). We found a small difference in word counts between their reported statistics for the MPQA lexicons and those we performed ourselves, as can be seen in the bold numbers in table 3. Originally, we assumed that a word can only occur in one sentiment class within the same lexicon, and this resulted in differing counts for all lexicons. This distinction is not clearly documented in the paper or code. However, our assumption turned out to be incorrect, giving a further illustration of why detailed descriptions and documentation of all decisions is important. We ran the same experiment as Vo and Zhang (2015) to show the effectiveness of sentiment lexicons the results can be seen in table 4. We can clearly see there are some difference not just with the accuracy scores but the rank of the sentiment lexicons. We found just using HL was best and MPQA does help performance compared to the **Target-dep** baseline which differs to Vo and Zhang (2015) findings. Since we found that using just HL performed best, the rest of the results will apply the **Target-dep+** method using HL and using HL & MPQA to show the affect of using the lexicon that both we and Vo and Zhang (2015) found best.

	Word Counts					
	Original		Reproduction			
Lexicons	Positive	Negative	Positive	Positive Lowered	Negative	Negative Lowered
MPQA	2289	4114	**2298**	**2298**	**4148**	**4148**
HL	2003	4780	2003	2003	4780	4780
NRC	2231	3243	2231	2231	3243	3243
MPQA & HL	2706	5069	**2725**	**2725**	**5080**	**5076**
All three	3940	6490	**4016**	**4016**	**6530**	**6526**

Table 3: Sentiment lexicon statistics comparison

	Results (Accuracy %)	
Sentiment Lexicon	Original	Reproduction
Target-dep	65.72	66.81
Target-dep+: NRC	66.05	67.13
Target-dep+: HL	67.24	**68.61**
Target-dep+: MPQA	65.56	66.81
Target-dep+: MPQA & HL	**67.40**	68.37
Target-dep+: All three	67.30	68.23

Table 4: Effectiveness of Sentiment Lexicons

4.1.2 Using different word vectors

The original authors tested their methods using three different word vectors: 1. Word2Vec trained by Vo and Zhang (2015) on 5 million Tweets containing emoticons (W2V), 2. Sentiment Specific Word

[5] http://mpqa.cs.pitt.edu/lexicons/subj_lexicon/

[6] http://saifmohammad.com/WebPages/NRC-Emotion-Lexicon.htm

[7] https://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html#lexicon

Embedding (SSWE) from Tang et al. (2014), and 3. W2V and SSWE combined. Neither of these word embeddings are available from the original authors as Vo and Zhang (2015) never released the embeddings and the link to Tang et al. (2014) embeddings no longer works[8]. However, the embeddings were released through Wang et al. (2017) code base[9] following requesting of the code from Vo and Zhang (2015). Figure 1 shows the results of the different word embeddings across the different methods. The main finding we see is that SSWE by themselves are not as informative as W2V vectors which is different to the findings of Vo and Zhang (2015). However we agree that combining the two vectors is beneficial and that the rank of methods is the same in our observations.

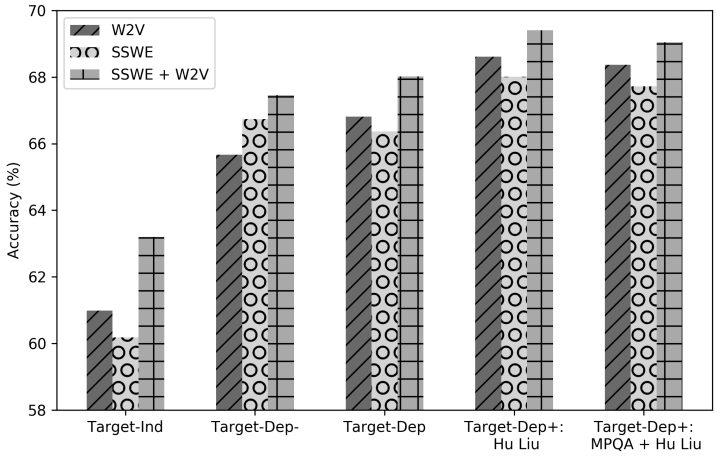

Figure 1: Effectiveness of word embedding

4.1.3 Scaling and Final Model comparison

We test all of the methods on the test data set of Dong et al. (2014) and show the difference between the original and reproduced models in figure 2. Finally, we show the effect of scaling using Max Min and not scaling the data.

As stated before, we have been using Max Min scaling on the NP features, however Vo and Zhang (2015) did not mention scaling in their paper. The library they were using, LibLinear (Fan et al., 2008), suggests in its practical guide (Hsu et al., 2003) to scale each feature to [0, 1] but this was not re-iterated by Vo and Zhang (2015). We are using scikit-learn's (Pedregosa et al., 2011) LinearSVC which is a wrapper of LibLinear, hence making it appropriate to use here. As can be seen in figure 2, not scaling can affect the results by around one-third.

4.2 Reproduction of Wang et al. (2017)

Wang et al. (2017) extended the NP work of Vo and Zhang (2015) and instead of using the full tweet/sentence/text contexts they used the full dependency graph of the target word. Thus, they created three different methods: 1. **TDParse-** uses only the full dependency graph context, 2. **TDParse** the feature of **TDParse-** and the left and right contexts, and 3. **TDParse+** the features of **TDParse** and LS and RS contexts. The experiments are performed on the Dong et al. (2014) and Wang et al. (2017) Twitter datasets where we train and test on the previously specified train and test splits. We also scale our features using Max Min scaling before inputting into the SVM. We used all three sentiment lexicons as in the original paper, and we found the C-Value by performing 5 fold stratified cross validation on the training datasets. The results of these experiments can be seen in figure 3[10]. As found with the results of Vo and Zhang (2015) replication, scaling is very important but is typically overlooked when reporting.

[8] `http://ir.hit.edu.cn/~dytang/`

[9] `https://github.com/bluemonk482/tdparse`

[10] For the Election Twitter dataset TDParse+ result were never reported in the original paper.

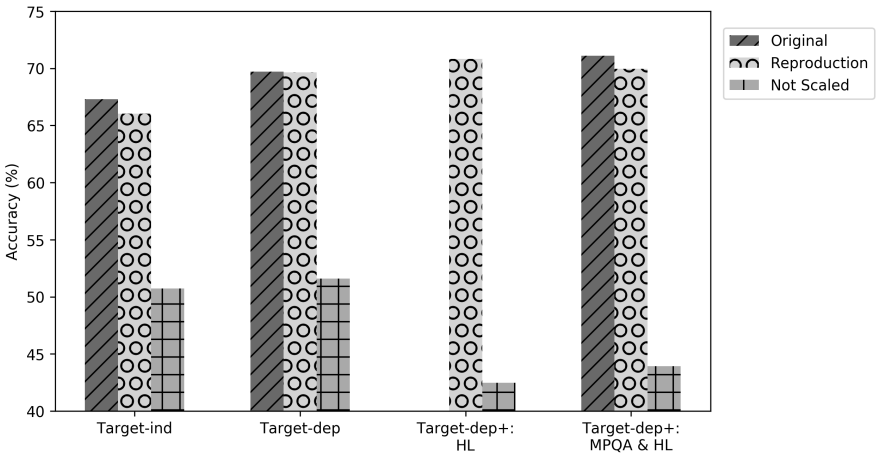

Figure 2: Target Dependent Final Results

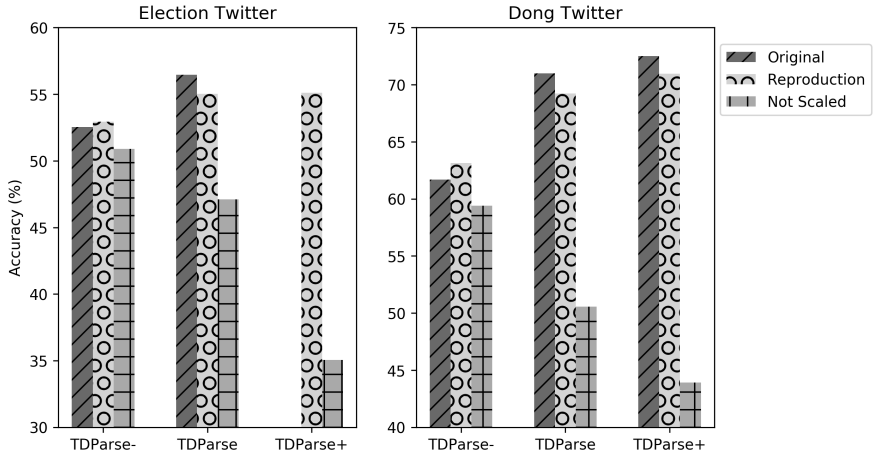

Figure 3: TDParse Final Results

4.3 Reproduction of Tang et al. (2016a)

Tang et al. (2016a) was the first to use LSTMs specifically for TDSA. They created three different models: 1. **LSTM** a standard LSTM that runs over the length of the sentence and takes no target information into account, 2. **TDLSTM** runs two LSTMs, one over the left and the other over the right context of the target word and concatenates the output of the two, and 3. **TCLSTM** same as the **TDLSTM** method but each input word vector is concatenated with vector of the target word. All of the methods outputs are fed into a softmax activation function. The experiments are performed on the Dong et al. (2014) dataset where we train and test on the specified splits. For the LSTMs we initialised the weights using uniform distribution U(0.003, 0.003), used Stochastic Gradient Descent (SGD) a learning rate of 0.01, cross entropy loss, padded and truncated sequence to the length of the maximum sequence in the training dataset as stated in the original paper, and we did not "set the clipping threshold of softmax layer as 200" (Tang et al., 2016a) as we were unsure what this meant. With regards to the number of epochs trained, we used early stopping with a patience of 10 and allowed 300 epochs. Within their experiments they used SSWE (Tang et al., 2014) and Glove Twitter vectors[11] (Pennington et al., 2014).

As the paper being reproduced does not define the number of epochs they trained for, we use early stopping. Thus for early stopping we require to split the training data into train and validation sets to know when to stop. As it has been shown by Reimers and Gurevych (2017) that the random seed statistically significantly changes the results of experiments we ran each model over each word embedding thirty times, using a different seed value but keeping the same stratified train and validation split, and

[11] https://nlp.stanford.edu/projects/glove/

	Macro F1		
Methods	O	R (Max)	R (Mean)
LSTM	64.70	64.34	60.69
TDLSTM	69.00	67.04	65.63
TCLSTM	69.50	67.66	65.23
O=Original, R=Reproduction			

Table 5: LSTM Final Results

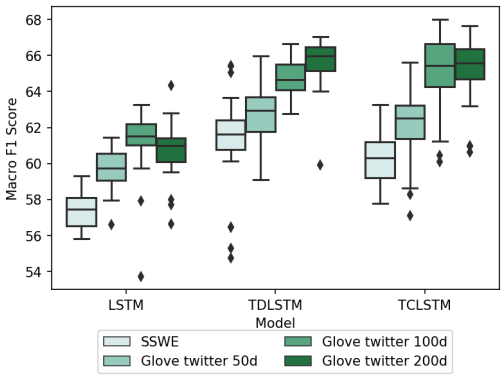

Figure 4: Distribution of the LSTM results

reported the results on the same test data as the original paper. As can be seen in Figure 4, the initial seed value makes a large difference more so for the smaller embeddings. In table 5, we show the difference between our mean and maximum result and the original result for each model using the 200 dimension Glove Twitter vectors. Even though the mean result is quite different from the original the maximum is much closer. Our results generally agree with their results on the ranking of the word vectors and the embeddings.

Overall, we were able to reproduce the results of all three papers. However for the neural network/deep learning approach of Tang et al. (2016a) we agree with Reimers and Gurevych (2017) that reporting multiple runs of the system over different seed values is required as the single performance scores can be misleading, which could explain why previous papers obtained different results to the original for the **TDLSTM** method (Chen et al., 2017; Tay et al., 2017).

5 Mass Evaluation

For all of the methods we pre-processed the text by lower casing and tokenising using Twokenizer (Gimpel et al., 2011), and we used all three sentiment lexicons where applicable. We found the best word vectors from SSWE and the common crawl 42B 300 dimension Glove vectors by five fold stratified cross validation for the NP methods and the highest accuracy on the validation set for the LSTM methods. We chose these word vectors as they have very different sizes (50 and 300), also they have been shown to perform well in different text types; SSWE for social media (Tang et al., 2016a) and Glove for reviews (Chen et al., 2017). To make the experiments quicker and computationally less expensive, we filtered out all words from the word vectors that did not appear in the train and test datasets, and this is equivalent with respect to word coverage as using all words. Finally we only reported results for the LSTM methods with one seed value and not multiple due to time constraints.

The results of the methods using the best found word vectors on the test sets can be seen in table 6. We find that the **TDParse** methods generally perform best but only clearly outperforms the other non-dependency parser methods on the YouTuBean dataset. We hypothesise that this is due to the dataset containing, on average, a deeper constituency tree depth which could be seen as on average more complex sentences. This could be due to it being from the spoken medium compared to the rest of the datasets which are written. Also that using a sentiment lexicon is almost always beneficial, but only by a small amount. Within the LSTM based methods the **TDLSTM** method generally performs the best indicating that the extra target information that the **TCLSTM** method contains is not needed, but we believe this needs further analysis.

We can conclude that the simpler NP models perform well across domain, type and medium and that even without language specific tools and lexicons they are competitive to the more complex LSTM based methods.

Dataset	Target-Dep F1	Target-Dep+ F1	TDParse F1	TDParse+ F1	LSTM F1	TDLSTM F1	TCLSTM F1
Dong Twitter	65.70	65.70	66.00	**68.10**	63.60	66.09	67.14
Election Twitter	45.50	45.90	**46.20**	44.60	38.70	43.60	42.08
Mitchell	40.80	42.90	40.50	50.00	47.17	**51.16**	41.03
SemEval 14 L	60.00	63.70	59.60	**64.50**	47.84	57.91	46.80
SemEval 14 R	56.20	57.70	59.40	**61.00**	46.36	57.68	55.38
YouTuBean	53.10	55.60	**71.70**	68.00	45.93	45.47	38.07
Mean	53.55	55.25	57.23	**59.37**	48.27	53.65	48.42

Table 6: Mass Evaluation Results

6 Discussion and conclusion

The fast developing subfield of TDSA has so far lacked a large-scale comparative mass evaluation of approaches using different models and datasets. In this paper, we address this generalisability limitation and perform the first direct comparison and reproduction of three different approaches for TDSA. While carrying out these reproductions, we have noted and described above, the many emerging issues in previous research related to incomplete descriptions of methods and settings, patchy release of code, and lack of comparative evaluations. This is natural in a developing field, but it is crucial for ongoing development within NLP in general that improved repeatability practices are adopted. The practices adopted in our case studies are to reproduce the methods in open source code, adopt only open data, provide format conversion tools to ingest the different data formats, and describe and document all settings via the code and Jupyter Notebooks (released initially in anonymous form at submission time)[12]. We therefore argue that papers should not consider repeatability (replication or reproduction) or generalisability alone, but these two key tenets of scientific practice should be brought together.

In future work, we aim to extend our reproduction framework further, and extend the comparative evaluation to languages other than English. This will necessitate changes in the framework since we expect that dependency parsers and sentiment lexicons will be unavailable for specific languages. Also we will explore through error analysis in which situations different neural network architectures perform best.

Acknowledgements

This research is funded at Lancaster University by an EPSRC Doctoral Training Grant.

References

Isabelle Augenstein, Sebastian Ruder, and Anders Søgaard. 2018. Multi-task learning of pairwise sequence classification tasks over disparate label spaces. *arXiv preprint arXiv:1802.09913*.

Jeremy Barnes, Roman Klinger, and Sabine Schulte im Walde. 2017. Assessing state-of-the-art sentiment models on state-of-the-art sentiment datasets. In *Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 2–12. Association for Computational Linguistics.

Peng Chen, Zhongqian Sun, Lidong Bing, and Wei Yang. 2017. Recurrent attention network on memory for aspect sentiment analysis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 463–472. Association for Computational Linguistics.

Christian Collberg and Todd A. Proebsting. 2016. Repeatability in computer systems research. *Commun. ACM*, 59(3):62–69, February.

Kia Dashtipour, Soujanya Poria, Amir Hussain, Erik Cambria, Ahmad YA Hawalah, Alexander Gelbukh, and Qiang Zhou. 2016. Multilingual sentiment analysis: state of the art and independent comparison of techniques. *Cognitive Computation*, 8(4):757–771.

[12] `https://github.com/apmoore1/Bella`

Li Dong, Furu Wei, Chuanqi Tan, Duyu Tang, Ming Zhou, and Ke Xu. 2014. Adaptive recursive neural network for target-dependent twitter sentiment classification. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 49–54. Association for Computational Linguistics.

Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. 2008. Liblinear: A library for large linear classification. *Journal of machine learning research*, 9(Aug):1871–1874.

Nicola Ferro and Donna Harman. 2009. Clef 2009: Grid@ clef pilot track overview. In *Workshop of the Cross-Language Evaluation Forum for European Languages*, pages 552–565. Springer.

Nicola Ferro and Gianmaria Silvello. 2016. The clef monolingual grid of points. In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 16–27. Springer.

Antske Fokkens, Marieke Van Erp, Marten Postma, Ted Pedersen, Piek Vossen, and Nuno Freire. 2013. Offspring from reproduction problems: What replication failure teaches us. In *ACL (1)*, pages 1691–1701.

Kevin Gimpel, Nathan Schneider, Brendan O'Connor, Dipanjan Das, Daniel Mills, Jacob Eisenstein, Michael Heilman, Dani Yogatama, Jeffrey Flanigan, and Noah A. Smith. 2011. Part-of-speech tagging for twitter: Annotation, features, and experiments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 42–47. Association for Computational Linguistics.

Chih-Wei Hsu, Chih-Chung Chang, Chih-Jen Lin, et al. 2003. A practical guide to support vector classification.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177. ACM.

Svetlana Kiritchenko, Xiaodan Zhu, Colin Cherry, and Saif Mohammad. 2014. Nrc-canada-2014: Detecting aspects and sentiment in customer reviews. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 437–442. Association for Computational Linguistics.

Jiangming Liu and Yue Zhang. 2017. Attention modeling for targeted sentiment. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 572–577. Association for Computational Linguistics.

Panos Louridas and Georgios Gousios. 2012. A note on rigour and replicability. *ACM SIGSOFT Software Engineering Notes*, 37(5):1–4.

Edison Marrese-Taylor and Yutaka Matsuo. 2017. Replication issues in syntax-based aspect extraction for opinion mining. In *Proceedings of the Student Research Workshop at the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 23–32. Association for Computational Linguistics.

Edison Marrese-Taylor, Jorge Balazs, and Yutaka Matsuo. 2017. Mining fine-grained opinions on closed captions of youtube videos with an attention-rnn.

Margaret Mitchell, Jacqui Aguilar, Theresa Wilson, and Benjamin Van Durme. 2013. Open domain targeted sentiment. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1643–1654. Association for Computational Linguistics.

Saif Mohammad and Peter Turney. 2010. Emotions evoked by common words and phrases: Using mechanical turk to create an emotion lexicon. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pages 26–34. Association for Computational Linguistics.

Tetsuya Nasukawa and Jeonghee Yi. 2003. Sentiment analysis: Capturing favorability using natural language processing. In *Proceedings of the 2nd international conference on Knowledge capture*, pages 70–77. ACM.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*.

John Pavlopoulos and Ion Androutsopoulos. 2014. Aspect term extraction for sentiment analysis: New datasets, new evaluation measures and an improved unsupervised method. In *Proceedings of the 5th Workshop on Language Analysis for Social Media (LASM)*, pages 44–52. Association for Computational Linguistics.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, pages 27–35. Association for Computational Linguistics.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad AL-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphee De Clercq, Veronique Hoste, Marianna Apidianaki, Xavier Tannier, Natalia Loukachevitch, Evgeniy Kotelnikov, Núria Bel, Salud María Jiménez-Zafra, and Gülşen Eryiğit. 2016. Semeval-2016 task 5: Aspect based sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 19–30. Association for Computational Linguistics.

Martin Potthast, Sarah Braun, Tolga Buz, Fabian Duffhauss, Florian Friedrich, Jörg Marvin Gülzow, Jakob Köhler, Winfried Lötzsch, Fabian Müller, Maike Elisa Müller, et al. 2016. Who wrote the web? revisiting influential author identification research applicable to information retrieval. In *European Conference on Information Retrieval*, pages 393–407. Springer.

Florian Prinz, Thomas Schlange, and Khusru Asadullah. 2011. Believe it or not: how much can we rely on published data on potential drug targets? *Nature Reviews Drug Discovery*, 10:712.

Nils Reimers and Iryna Gurevych. 2017. Reporting score distributions makes a difference: Performance study of lstm-networks for sequence tagging. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 338–348. Association for Computational Linguistics.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642. Association for Computational Linguistics.

Efstratios Sygkounas, Giuseppe Rizzo, and Raphaël Troncy. 2016. A replication study of the top performing systems in semeval twitter sentiment analysis. In *International Semantic Web Conference*, pages 204–219. Springer.

Duyu Tang, Furu Wei, Nan Yang, Ming Zhou, Ting Liu, and Bing Qin. 2014. Learning sentiment-specific word embedding for twitter sentiment classification. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1555–1565. Association for Computational Linguistics.

Duyu Tang, Bing Qin, Xiaocheng Feng, and Ting Liu. 2016a. Effective lstms for target-dependent sentiment classification. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3298–3307. The COLING 2016 Organizing Committee.

Duyu Tang, Bing Qin, and Ting Liu. 2016b. Aspect level sentiment classification with deep memory network. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 214–224.

Yi Tay, Anh Tuan Luu, and Siu Cheung Hui. 2017. Learning to attend via word-aspect associative fusion for aspect-based sentiment analysis. *arXiv preprint arXiv:1712.05403*.

Peter Turney. 2002. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*.

Duy-Tin Vo and Yue Zhang. 2015. Target-dependent twitter sentiment classification with rich automatic features. In *IJCAI*, pages 1347–1353.

Yequan Wang, Minlie Huang, xiaoyan zhu, and Li Zhao. 2016. Attention-based lstm for aspect-level sentiment classification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 606–615. Association for Computational Linguistics.

Bo Wang, Maria Liakata, Arkaitz Zubiaga, and Rob Procter. 2017. Tdparse: Multi-target-specific sentiment recognition on twitter. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 483–493. Association for Computational Linguistics.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*.

Theresa Ann Wilson. 2008. *Fine-grained subjectivity and sentiment analysis: recognizing the intensity, polarity, and attitudes of private states*. University of Pittsburgh.

Meishan Zhang, Yue Zhang, and Duy Tin Vo. 2015. Neural networks for open domain targeted sentiment. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 612–621.

Meishan Zhang, Yue Zhang, and Duy-Tin Vo. 2016. Gated neural networks for targeted sentiment analysis. In *AAAI*, pages 3087–3093.