Approaching SMM4H with Merged Models and Multi-task Learning

Tilia Ellendorff, Lenz Furrer, Nicola Colic, Noëmi Aepli, Fabio Rinaldi
Institute of Computational Linguistics, University of Zurich
{name.surname}@uzh.ch

Abstract

We describe our submissions to the 4th edition of the Social Media Mining for Health Applications (SMM4H) shared task. Our team (UZH) participated in two sub-tasks: *Automatic classifications of adverse effects mentions in tweets* (Task 1) and *Generalizable identification of personal health experience mentions* (Task 4). For our submissions, we exploited ensembles based on a pre-trained language representation with a neural transformer architecture (BERT) (Tasks 1 and 4) and a CNN-BiLSTM(-CRF) network within a multi-task learning scenario (Task 1). These systems are placed on top of a carefully crafted pipeline of domain-specific pre-processing steps.

1 Introduction

The Social Media Mining for Health Applications (SMM4H) shared task 2019 (Weissenbacher et al., 2019) focused on classical natural-language-processing (NLP) problems applied to Twitter microposts (tweets). Our team participated in two tasks of binary text classification: tweets are labeled positive if they contain an Adverse Drug Reaction (ADR) in Task 1 or a Personal Health Mention (PHM) in Task 4. Task 1 (automatic classifications of adverse effects mentions in tweets) is a re-run of the ADR task from previous editions of the SMM4H shared task. Task 4 (generalizable identification of personal health experience mentions) was run for the first time. This task consists in deciding if a tweet contains personal health mentions, as opposed to mentions of general awareness of a health issue. Here, the main challenge is to generalize from the health contexts given by the two datasets provided as training data (i.e. flu vaccination and flu infection) to other, possibly very different, health contexts.

2 Data and Pre-processing

The organizers provided all participants with labeled training data which included the text of the tweets (as opposed to the previous years where only tweet ids were provided). Table 1 describes the size of the available datasets.

Data for Task 4 originated from two different flu-related contexts, namely flu infection (Lamb et al., 2013) and flu vaccination (Huang et al., 2017). Each of these two datasets has their own specific scope. Within the infection dataset, positively labeled examples are restricted to reports of own infection (i.e., the author of the tweet is infected) or infection of somebody close to the author, whereas tweets mentioning personal vaccination are labeled as negative. The vaccination dataset labels tweets as positive only if they report that either the author, or a person close to the author, has actually been vaccinated. Tweets about personal infection are labeled as negative within this dataset. Task 4, on the other hand, looks to label all instances as positive which contain a personal health mention (be it infection or vaccination or any other health context) without a specified restricted scope. Therefore, the main challenge of Task 4 is to generalize from the specific health contexts, as provided within the training data, to personal health mentions in general.

For both tasks, we pre-processed all tweets with the following steps:

- Without sentence splitting, the tweets are tokenized using NLTK's Twitter tokenizer.[1]
- User names and numbers are replaced with "@user" and "NUMBER", respectively.
- URLs are truncated to their domain names.
- Hash symbols are stripped from hash tags.

[1] https://www.nltk.org/api/nltk.tokenize.html

Proceedings of the 4th Social Media Mining for Health Applications (#SMM4H) Workshop & Shared Task, pages 58–61
Florence, Italy, August 2, 2019. ©2019 Association for Computational Linguistics

		# tweets			**# unique tweets**		
		neg	**pos**	**total**	**neg**	**pos**	**total**
Task 1	total	23301	2377	25678	22497	2368	24861
Task 4	*Inf*	472	564	1036	460	545	1005
	Vacc	4815	1900	6715	4680	1885	6515
	total	5287	2464	7751	5140	2430	7570

Table 1: Number of tweets provided for each task as training data. Task 4 includes data from the health context of Vaccination (*Vacc*) and Infection (*Inf*). Unique tweets are counted after pre-processing followed by removal of duplicates.

- Camel-cased expressions like “SideEffects” are split into their component words.
- Artifacts of upstream processing like “&amp;” are fixed.
- Frequent colloquial abbreviations (e.g. “w/” for “with”) are resolved.
- Repeated letters (“greaaaaat”) are removed. Specifically, runs of three or more equal letters are replaced with a single occurrence, except for “e”, where two letters are retained (e.g. “freeeeeze” becomes “freeze”). Letter de-repetition was not applied to the BERT-based systems (described below).

The datasets contain a considerable number of duplicates, i.e. tweets with the same or very close content, including retweets. For the cross-validation in Task 1, we ensured that duplicate tweets were not spread across different folds. In Task 4, this was achieved by removing all duplicate tweets from the training set after pre-processing and before training (i.e. for our experiments, numbers of unique tweets in Table 1 apply).

3 Experiments and System Descriptions

For Task 1, we experimented with two different systems, separately and in combination. The first system (labeled MTL) is a CNN+BiLSTM neural network with multi-task-learning (MTL) capabilities (Caruana, 1997). The multi-task architecture allows tackling multiple tasks (datasets) in a single model, based on the idea that complementary information from different tasks can lead to mutual benefit when they are trained jointly (see e.g. Crichton et al., 2017). The architecture distinguishes shared layers, where parameters are updated for all tasks during training, and task-specific layers with parameters dedicated to a single task. Our MTL architecture is able to handle different types of tasks, such as sequence labeling and document classification, in the same model. In the present configuration, the model was trained on data from Task 1, Task 2, and the CADEC corpus (Karimi et al., 2015), where the latter two served as helper tasks, solving the problem of span detection for ADRs. In the shared part of the model, character embeddings are combined with pre-trained word embeddings (Godin et al., 2015) into a bidirectional Long-Short Term Memory (BiLSTM) layer. In the task-specific layer, the sequence-labeling tasks are modeled with Conditional Random Fields (CRF), whereas the text-level classifier for Task 1 is based on the final state of the BiLSTM layer directly. Additionally, the Task-1 classifier uses a lexicon feature based on a fuzzy-match lookup in the MedDRA vocabulary.[2] We trained 10 different models in a cross-validation setting, using a held-out set to prevent overfitting through early stopping. The predicted labels are based on the mean of the scores of all folds (transformed by softmax).

We based the second system (labeled BERT) for Task 1 on BERT, a pre-trained language representation with a neural transformer architecture (Devlin et al., 2018). Our system merged parameters of 20 models (originating from 10-fold cross validation trained once for four epochs and once with early stopping[3]) into a single model (Utans, 1996; Junczys-Dowmunt et al., 2016). For this, we calculated the weighted sum of parameters across models: we weighted parameters of each model by their performance on the respective testing fold (measured as F-score and transformed by softmax). By applying this method, we first separately merged the systems resulting from training with early stopping and from training for 4 fixed epochs, and subsequently, merged the two resulting systems into a single system. For this last merging step, we gave the system resulting from merging early stopping systems nine times the weight of the other system which resulted from merging systems trained for a fixed number of epochs. For the last run (MTL+BERT), we combined predictions from all 20 BERT systems with the first system and a second MTL configuration which uses different word embeddings (Ellendorff et al., 2018) and omits lexicon features.

For Task 4, our submission consisted of three

[2] `https://www.meddra.org/`

[3] Early stopping was done on 0.2 of the training portion with a patience of 2.

	System	Precision	Recall	F-Score	Accuracy
Task 1	MTL	0.585	0.438	0.501	
	BERT	0.648	**0.567**	**0.605**	
	MTL+BERT	**0.705**	0.420	0.527	
	mean	*0.535*	*0.505*	*0.502*	
Task 4	Merge	0.839	**0.909**	**0.873**	**0.877**
	Average	0.988	0.614	0.757	0.818
	Join	**1.000**	0.515	0.680	0.775
	mean	*0.902*	*0.585*	*0.701*	*0.781*

Table 2: Official scores for our submissions, compared to mean scores of all participating systems (best results in bold).

		Precision	Recall	F-Score	Accuracy
Task 4	HC 1	0.910	0.988	0.947	0.944
	HC 2	0.706	0.774	0.739	0.754
	HC 3	0.750	0.790	0.769	0.839

Table 3: Official scores for Task 4, System 1 (Merged BERT models across contexts) by Health Context/Health Concern (HC).

different types of BERT-based ensemble systems. Our first system (labeled Merge) is similar to the second system (BERT) of Task 1. We trained two systems using 10-fold cross validation: one for infection and one for vaccination. Subsequently, we first merged the resulting systems across folds[4] and, in a second step, we merged the two resulting systems into one single system, giving nine times the weight to the system resulting from training on the infection dataset. This run has ranked first among all systems participating in the task. The second run (labeled Average) is again trained on both datasets separately using 10-fold cross validation, resulting in 20 independent systems. Labels are determined by averaging label probabilities returned by all 20 systems. Finally, the third run (Join) is trained on both datasets jointly but giving twice as much weight to all data points from the infection dataset, again using 10-fold cross validation, and probabilities were averaged across these 10 systems.

For both tasks, our BERT classifiers are based on the PyTorch implementation of BERT[5] and fine-tune the pre-trained model provided by Google research as *BERT-Base, Uncased*[6]. Where not mentioned otherwise, all systems were trained with the BertAdam optimizer for four epochs with a batch size of 30 (Task 1) or 5 (Task 4), a learning rate of 5×10^{-5} and linear warmup schedule with a fixed number of 9050 training steps.

	System	Precision	Recall	F-Score
Task 1	Single	**0.765**	0.385	0.512
	Majority vote	0.688	0.462	0.552
	Merge unweighted	0.623	0.617	0.619
	Merge weighted	0.625	**0.621**	**0.623**

Table 4: Scores for post-submission runs for Task 1 (all BERT classifiers trained with early stopping). Single: single system trained on the whole training data; Majority vote: majority voting ensemble; Merge unweighted: unweighted parameter merging; Merge weighted: weighted parameter merging.

4 Results and Discussion

Table 2 shows official results on the test set. The official unlabeled test sets for Tasks 1 and 4 comprise 4575 and 285 tweets, respectively. Apart from an overall evaluation, systems submitted for Task 4 were also evaluated with respect to three different health contexts (also: health concerns), which were still undisclosed by the time we wrote this system description. For our best performing system (Merge), results for each health context can be found in Table 3.

In Task 1, the BERT-based model clearly outperformed our competing MTL-based approach. After the submission deadline, we used the evaluation interface to obtain test set evaluation scores for a BERT system, which for Task 1 only includes the systems trained with early stopping (i.e. we excluded the system which was trained for 4 fixed epochs). This still gave us a considerable improvement. Besides merging the 10 models into one, we also experimented with voting ensembles but found that merging models in fact gave us the best performance, with the weighted version still achieving a slight improvement compared to the unweighted version. Results for Task 1 post-submission runs can be found in Table 4.

Our results for both tasks show that merging models gives us a large improvement compared to traditional ensembling techniques (such as majority voting). Furthermore, merging parameters from several models into a single model means that only a single model is needed at prediction time. This brings a considerable advantage in terms of memory and computation time when predicting labels.

[4] For Task 4 we did not weight systems by their performance on the test fold, as we did for Task 1.

[5] https://github.com/huggingface/pytorch-pretrained-BERT

[6] https://github.com/google-research/bert

References

Rich Caruana. 1997. Multitask learning. *Machine Learning*, 28(1):41–75.

Gamal Crichton, Sampo Pyysalo, Billy Chiu, and Anna Korhonen. 2017. A neural network multi-task learning approach to biomedical named entity recognition. *BMC Bioinformatics*, 18(1):368.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Tilia Ellendorff, Joseph Cornelius, Heath Gordon, Nicola Colic, and Fabio Rinaldi. 2018. UZH@SMM4H: System descriptions. In *Proceedings of the 2018 EMNLP Workshop SMM4H: The 3rd Social Media Mining for Health Applications Workshop and Shared Task*, pages 56–60, Brussels, Belgium.

Fréderic Godin, Baptist Vandersmissen, Wesley De Neve, and Rik Van de Walle. 2015. Multimedia Lab @ ACL W-NUT NER shared task: Named entity recognition for Twitter microposts using distributed word representations. In *Proceedings of the Workshop on Noisy User-generated Text*, pages 146–153, Beijing, China.

Xiaolei Huang, Michael Smith, Michael Paul, Dmytro Ryzhkov, Sandra Quinn, David Broniatowski, and Mark Dredze. 2017. Examining patterns of influenza vaccination in social media. In *AAAI Joint Workshop on Health Intelligence (W3PHIAI)*.

Marcin Junczys-Dowmunt, Tomasz Dwojak, and Rico Sennrich. 2016. The AMU-UEDIN submission to the WMT16 news translation task: Attention-based NMT models as feature functions in phrase-based SMT. In *Proceedings of the First Conference on Machine Translation*, pages 319–325, Berlin, Germany.

Sarvnaz Karimi, Alejandro Metke-Jimenez, Madonna Kemp, and Chen Wang. 2015. CADEC: A corpus of adverse drug event annotations. *Journal of Biomedical Informatics*, 55:73–81.

Alex Lamb, Michael J. Paul, and Mark Dredze. 2013. Separating fact from fear: Tracking flu infections on twitter. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 789–795, Atlanta, Georgia. Association for Computational Linguistics.

Joachim Utans. 1996. Weight averaging for neural networks and local resampling schemes. In *Proc. AAAI-96 Workshop on Integrating Multiple Learned Models*, pages 133–138. AAAI Press.

Davy Weissenbacher, Abeed Sarker, Arjun Magge, Ashlynn Daughton, Karen O'Connor, Michael Paul, and Graciela Gonzalez-Hernandez. 2019. Overview of the fourth Social Media Mining for Health (SMM4H) shared task at ACL 2019. In *Proceedings of the 2019 ACL Workshop SMM4H: The 4th Social Media Mining for Health Applications Workshop & Shared Task*.