A Simple and Effective biLSTM Approach to Aspect-Based Sentiment Analysis in Social Media Customer Feedback

Simon Clematide
Institute of Computational Linguistics
University of Zurich, Switzerland
simon.clematide@cl.uzh.ch

Abstract

This paper describes a system for aspect-based sentiment analysis (ABSA) using a straight-forward supervised sequence labeling approach. Specifically, we apply a bidirectional, recurrent long short-term memory (biLSTM) architecture with a multilayer perceptron on top that predicts the labels token by token. We deal with the issue of rare words by dynamically switching between character-level and token-level representations depending on an occurrence threshold. A simple encoding of the aspects and their sentiments, a careful preprocessing of the data, and a generous ensemble of 24 single models beats the published state-of-the-art results for the GermEval 2017 ABSA data set for aspect-based sentiment analysis on the document level (joint prediction of aspect and sentiment in task C). For task D, the opinion target expression (OPE) detection task, our approach improves the current state-of-the-art even by 2.7-14.3 percentage points.

1 Introduction

Text mining on user-generated social media content is an important application domain in natural language processing. *Aspect-based sentiment analysis* (ABSA) is an information extraction task that is generally defined as follows (Liu, 2012, 12): Given a document, recognize all opinions expressed in it. Formally, an *opinion* is a quintuple (e,a,s,h,t) that includes a sentiment s (negative, positive, neutral judgment) about an aspect a of an entity e as expressed at time t by an opinion holder h.

Dedicated ABSA shared tasks started in SemEval 2014 on English product and restaurant reviews (Pontiki et al., 2014). The GermEval Shared Task 2017 on "Aspect-based Sentiment in Social Media Customer Feedback"[1] (Wojatzki et al., 2017) provides the first publicly available ABSA data set of substantial size for German.

Task C of GermEval 2017 instantiates the general ABSA information extraction task as follows: Given a social media document, recognize all aspects and their respective sentiments regarding the entity "Deutsche Bahn" (German railway company). The information about the time and opinion holder are given by the document's metadata and are not part of the task. Task D, the opinion target expression (OPE) detection, additionally requires the identification of mentions in the text that express a certain aspect.

2 Material

The *official data sets* (version 1.4) contain 19,432 training and 2,369 development documents. There is a synchronic test set SYN with 2,566 documents (drawn from May 2015 to June 2016 as the training set) and a diachronic test set DIA with 1842 documents (drawn from November 2016 to January 2017). Each document was annotated separately by two annotators and differences were adjudicated by a supervisor (Wojatzki et al., 2017).

The distribution of *sentiment polarity* on document level is highly imbalanced: 17,758 neutral, 6.911 negative, 1,540 positive. The high number of neutral documents is probably due to the content keyword-based crawling using subword matchings, e.g. allowing hits as "zugig" ('drafty') for the search term "zug" ('train'). An SVM-based text classification pre-filtering step applied after randomly down-sampling the crawled documents did probably exclude a substantial amount of irrelevant documents (83% of all documents are relevant in the final data sets), but obviously kept many neutral ones w.r.t the target entity. The prevalence of

[1] Datasets and guidelines are available from https://sites.google.com/view/germeval2017-absa

Proceedings of the 14th Conference on Natural Language Processing (KONVENS 2018)
Vienna, Austria – September 19-21, 2018

```
<Document ...><Opinions>
<Opinion category="Allgemein#Haupt" from="0" to="0" target="NULL" polarity="negative"/>
<Opinion category="Sonstige_Unregelmässigkeiten#Haupt" from="5" to="20"
target="Weichen Störung" polarity="negative"/></Opinions>
<relevance>true</relevance><sentiment>negative</sentiment>
<text>Juhu Weichen Störung! Ich liebe die Bahn ... Nicht -.-</text></Document>

juhu/O weichen/Sonstige_Unregelmässigkeiten:negative störung/Sonstige_Unregelmässigkeiten:negative !/O
ich/O liebe/O die/O bahn/O .../O nicht/O -.-/O __D__/Allgemein:negative
```

Figure 1: Original XML format (above) and input format for the sequence tagger (below)

negative compared to positive feedback is expected for such an application.

The *aspect categories* are organized into 18 semantically specified classes (for instance, "punctuality and connectivity" or "atmosphere", and 1 general class GENERAL that covers anything about the company which can not be assigned to any other class (GermEval, 2017). Again, the distribution of all 21,772 annotated aspects is highly imbalanced: GENERAL covers 68.5% of all cases, the top 10 semantically defined aspect categories cover 29.2%, the rest only 2.3%.

Not all aspect annotations are bound to specific text mentions in the document. A substantial amount (23%) of aspect annotations apply to the document level only. A document as well as a specific expression in the document can have more than one aspect annotation. Therefore, in general, this dataset poses a multi-label multi-class type of classification problem.

The organizers distribute the data sets in a standoff XML markup format where text mentions with aspect annotations are identified by character offsets (zero offsets indicate document-level annotations, see Fig. 1).

3 Methods

Our approach simplifies the multi-label multi-class problem into a standard sequence labeling task with single-labels. The token labels (aspect and/or polarity) could be encoded by any of the IOB variants (Sang and Veenstra, 1999). However, given that the annotations are sparse anyway (less than 4% of the training tokens have a label) and typically are not adjacent to each other, we stick to a simple IO scheme.

The text segmentation and preprocessing works as follows: (a) A simple tokenizer based on regular expressions splits the texts into a sequence of words (hash tags and @USER mentions are kept intact, URLs are replaced by a special token) and punctuation tokens. (b) The annotations defined on character level are mapped onto the computed token level. In cases where the character offsets do not exactly match token boundaries, e.g. "Strecken" in the word "Streckensanierung" (*rehabilitation of lines*), we label the token that contains the annotation. (e) In order to deal with document level aspects in a uniform manner, we attach a dummy token at the end of each document and assign it the document aspect. (f) If more than one aspect is annotated (happens mostly on the document level), we reduce it to the most frequent aspect according to training data statistics. Figure 1 shows a document in the original XML format and the converted "aspect-tagged" version of it used for training and prediction.

3.1 BiLSTM Architecture

Recurrent Neural Networks (RNNs) (Elman, 1990) are well suited for sequence labeling tasks because they can naturally process variable-length input, and – in principle – they can model unbounded label dependencies. Due to learning problems of vanilla RNNs, more complex recurrent architectures such as long short-term memory cells (LSTM) (Greff et al., 2016) were developed, which build the basic blocks in many state-of-the-art NLP systems recently. Our approach uses two bidirectional LSTMs (biLSTM) (Graves and Schmidhuber, 2005) which are well suited for typical text tagging problems (Huang et al., 2015; Plank et al., 2016). More concretely, we adapted an existing part-of-speech tagger[2] that mixes word-level and character-level representations (Ling et al., 2015). The implementation in DyNet (Neubig et al., 2017), an autograd-based neural framework with dynamic computation graphs, considerably eases the flexible combination of word-level and character-level embeddings based on token frequency.

Formally, we learn a task-specific embedding $\mathbb{R}^{64}$ for each token $w \in V$ in our training set that

[2] https://raw.githubusercontent.com/clab/dynet_tutorial_examples/master/tutorial_bilstm_tagger.py

occurs at least 3 times. From all less frequently occurring words, we learn task-specific character embeddings $\mathbb{R}^{32}$. Characters not seen at least 5 times in training are simply ignored. A sequence of T embedded input items $(\mathbf{x}_1,\ldots,\mathbf{x}_T)$ is mapped onto a sequence of recurrent outputs (hidden state of dimension $\mathbb{R}^{64}$ for words, and $\mathbb{R}^{32}$ for characters):

$$(\mathbf{h}_1,\ldots,\mathbf{h}_T) = \mathit{LSTM}((\mathbf{x}_1,\ldots,\mathbf{x}_T))$$

We use the LSTM variant with coupled input and output gates with peepholes (Greff et al., 2016). The forward pass $F = \mathit{LSTM}((\mathbf{x}_1,\ldots,\mathbf{x}_T))$ and the backward pass $B = \mathit{LSTM}((\mathbf{x}_T,\ldots,\mathbf{x}_1))$ are concatenated elementwise:

$$\begin{aligned}(\mathbf{b}_1,\ldots,\mathbf{b}_T) &= \mathit{BiLSTM}((\mathbf{x}_1,\ldots,\mathbf{x}_T)) \\ &= ([F_1;B_1^{-1}],\ldots,[F_T;B_T^{-1}]),\end{aligned}$$

where B^{-1} denotes the reversed sequence B, and $[\bullet;\bullet]$ denotes vector concatenation.

On the level of words, each BiLSTM word representation $\mathbf{b}_i$ has a dimension of $\mathbb{R}^{2*64}$. A "unidirectional" word represented by a character BiLSTM embedding has a dimension of $\mathbb{R}^{64}$, but each character has a hidden size of 32. A simple way of combining the character-level representation is the vector concatenation of the hidden state from the last character $\mathbf{x}_T$ of the forward pass F with the last hidden state of B (=the first character):

$$\mathit{BiLSTM}_{\mathit{char}}((\mathbf{x}_1,\ldots,\mathbf{x}_T)) = [B_T;F_T]$$

It is worth noting that we insert a special marker at the beginning and end of a word. Therefore, every word actually starts and ends with the same artificial boundary "character".

The contextualized BiLSTM representation $\mathbf{b}_i$ of input word $\mathbf{x}_i$ goes into a multilayer perceptron MLP (only one hidden layer with dimensionality $\mathbb{R}^{64}$ and tanh activation function) with an output layer of the dimensionality of the number of classes $\mathbb{R}^{20}$. The softmax function computes the probability for each class y^k of each input i given all model parameters Θ:

$$P_\Theta(y_i^k) = \mathrm{softmax}^k(\mathit{MLP}(\mathbf{b}_i))$$

The actual prediction is the class with the highest probability.[3]

Training For hard classification, the loss of a predicted label sequence $(\hat{y}_1,\ldots,\hat{y}_T)$ is simply the sum of the negative log likelihood (cross entropy loss) of the true label sequence $(y_1,\ldots,y_T)$:

$$L((y_1,\ldots,y_T),\Theta) = \sum_{i\in\{1..T\}} -\log(P_\Theta(y_i))$$

We train with the ADAM optimizer (Kingma and Ba, 2014) for maximally 60 epochs and apply early stopping with a patience of 5 on the criterion of F score of all real aspect labels (the dominant class "O" is ignored). Training by single instances (no mini-batching) on CPU is efficient with DyNet and takes a couple of minutes for a single model.

Ensembling Neural approaches are typically sensitive to different initializations, and the performance of single models can vary considerably, especially in the presence of sparse feature and label distributions. A simple measure against weaknesses and biases in individual models are ensembles. Our final results are therefore built by a voting scheme from 24 models. On the development set, we determined a well-performing threshold of 33%, meaning, if one third of the models suggest an aspect label ("O" labels excluded) we take it. This boosts recall in the presence of highly imbalanced classes.

4 Experiments and Results

We trained two different systems according to the different evaluation regimes (aspects with or without sentiment labels): Our system A is trained with aspect classes only, system A:S has labels that combine the aspect with the sentiment. For the task C, we take the set of all real aspect predictions of the ensemble (no duplicate aspects are predicted)[4].

Table 1 compares our results on task C with official baselines and the top performing system from the official shared task (F scores as computed by the official evaluation script) and the organizers own system LT-ABSA (Ruppert et al., 2017), which did not participate in the shared task. Both our systems beat all shared task systems on task C. LT-ABSA performs better for the aspect-only evaluation, our systems outperforms LT-ABSA in aspect/sentiment prediction. It is interesting to note that the more fine-grained label set A:S (there are 59 different labels in the training set) has a better performance

[3] Neither beam search nor global CRF-style decoding is applied.

[4] The official evaluation script expects duplicates of the same aspect when present in the gold standard, which does not make much sense to us.

Task C	SYN		DIA	
System	A	A:S	A	A:S
Majority bsl.	44.3	31.5	45.6	38.4
Organizers' bsl.	48.1	32.2	49.5	38.9
Mishra	42.1	34.9	46	40.1
Lee (best run)	48.2	35.4	n/a	n/a
LT-ABSA	**53.7**	39.6	**55.6**	42.4
Our A	49.0		53.2	
Our A:S	49.6	**39.8**	53.6	**44.7**

Table 1: F score results of task C (A=aspect, S=sentiment, bsl.=baseline)

Task D (=OTE)	SYN		DIA	
System	exact	overl.	exact	overl.
Organizers' bsl.	17.0	23.7	21.6	27.1
Mishra	22.0	22.1	28.1	28.2
Lee (best run)	20.3	34.8	n/a	n/a
LT-ABSA	22.9	30.6	30.1	36.5
Our	**36.8**	**37.5**	**44.4**	**45.2**

Table 2: F score results of task D (overl.=overlap)

than the seemingly easier label set A (20 different labels).

The imbalanced distribution, that is, a lot of neutral aspects of type GENERAL, results in a strong majority class baseline (Wojatzki et al., 2017). The organizers' baseline (Wojatzki et al., 2017) uses a linear SVM for task C and performs pretty well.

The LT-ABSA is stronger than our system for the subtask of pure aspect classification. However, on the full task of joint prediction of aspect and sentiment our system sets a new state-of-the-art benchmark on this data set.

Table 2 shows the results of the exact and overlap match evaluation strategy for task D, the so-called opinion target expression (OTE) task. Here, our sequence-labeling approach shows an outstanding performance on all datasets compared to the published results: improvement between 2.7 and 14.3 percentage points in F score.

There are several ways to improve our system. The basic tokenization which does not especially well deal with the orthography of user-generated content. The preprocessing could be adapted along the ideas of the top-performing SemEval system by Baziotis et al. (2017) who insert structural tags for dates, URLs, hashtags, emoticons, or ALLCAP text etc.

5 Related Work

Lee et al. (2017) apply an IOB encoding on the token level for the text-bound aspects and use a BiLSTM-CRF architecture with word and character representations similar to ours. For the document-level aspects and polarity predictions, however, they use separate learners. The overall architecture is considerably more complex than ours without an actual benefit. Document level aspects for short texts can be represented easily by our dummy tokens. Lee et al. (2017) make use of external data to compute sentence embeddings.

Mishra et al. (2017) also use BiLSTMs for task C, but they model it as a more complex multi-label multi-class classification task. In contrast to our solution which relies only on task-specific embeddings, they integrate pre-trained word embeddings. For task D, they apply a biLSTM averaged structured perceptron approach which give the best official shared task results on the diachronic test set.

The LT-ABSA system (Ruppert et al., 2017) uses a specifically crawled comparable in-domain corpus and builds (a) pre-trained embeddings from it, (b) TF/IDF features, and (c) distributionally extended sentiment lexicons from it. Note that our systems only relies on the official training data and uses no external resources at all.

6 Conclusion

We presented a simple and yet effective neural system for supervised aspect-based sentiment analysis that dynamically mixes word and character-level representations. The system performs considerably better on joint aspect-sentiment document-level prediction on the GermEval 2018 data set than any other published systems.[5] On task D, the opinion target expression (OPE) detection task, our approach improves the current state-of-the-art of the LT-ABSA system depending on the data set and evaluation regime by 2.7 to 14.3 percentage points. The LT-ABSA system (Ruppert et al., 2017) is still stronger for the aspect-only predictions in task C, however, it uses additional external resources. Given the comfortable size of the training data, task-specific word and character embeddings seem to be sufficient (or even superior) for achieving good performance.

[5] Our system is available from `https://github.com/simon-clematide/konvens-2018-german-absa`

References

Christos Baziotis, Nikos Pelekis, and Christos Doulkeridis. 2017. DataStories at SemEval-2017 Task 4: Deep LSTM with attention for message-level and topic-based sentiment analysis. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 747–754.

Jeffrey L. Elman. 1990. Finding structure in time. *Cognitive Science*, 14(2):179–211, Mar.

GermEval. 2017. Guidelines sentiment analysis DB v4. electronic `http://ltdata1.informatik.uni-hamburg.de/germeval2017/Guidelines_DB_v4.pdf`.

Alex Graves and Jürgen Schmidhuber. 2005. Framewise phoneme classification with bidirectional lstm networks. In *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, volume 4, pages 2047–2052 vol. 4, July.

Klaus Greff, Rupesh K. Srivastava, Jan Koutník, Bas R. Steunebrink, and Jürgen Schmidhuber. 2016. Lstm: A search space odyssey. *IEEE Transactions on Neural Networks and Learning Systems*, PP(99):1–11.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional LSTM-CRF models for sequence tagging. *CoRR*, abs/1508.01991.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

Ji-Ung Lee, Steffen Eger, Johannes Daxenberger, and Iryna Gurevych. 2017. UKP TU-DA at GermEval 2017: Deep learning for aspect based sentiment detection. In *Proceedings of the GermEval 2017 – Shared Task on Aspect-based Sentiment in Social Media Customer Feedback*.

Wang Ling, Chris Dyer, Alan W Black, Isabel Trancoso, Ramon Fermandez, Silvio Amir, Luis Marujo, and Tiago Luis. 2015. Finding function in form: Compositional character models for open vocabulary word representation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1520–1530.

Bing Liu. 2012. *Sentiment Analysis and Opinion Mining*, volume 5 of *Synthesis Lectures on Human Language Technologies*. Morgan & Claypool Publishers.

Pruthwik Mishra, Vandan Mujadia, and Soujanya Lanka. 2017. GermEval 2017 : Sequence based models for customer feedback analysis. In *Proceedings of the GermEval 2017 – Shared Task on Aspect-based Sentiment in Social Media Customer Feedback*, pages 36–42.

Graham Neubig, Chris Dyer, Yoav Goldberg, Austin Matthews, Waleed Ammar, Antonios Anastasopoulos, Miguel Ballesteros, David Chiang, Daniel Clothiaux, Trevor Cohn, Kevin Duh, Manaal Faruqui, Cynthia Gan, Dan Garrette, Yangfeng Ji, Lingpeng Kong, Adhiguna Kuncoro, Gaurav Kumar, Chaitanya Malaviya, Paul Michel, Yusuke Oda, Matthew Richardson, Naomi Saphra, Swabha Swayamdipta, and Pengcheng Yin. 2017. Dynet: The dynamic neural network toolkit. *arXiv preprint arXiv:1701.03980*.

Barbara Plank, Anders Søgaard, and Yoav Goldberg. 2016. Multilingual part-of-speech tagging with bidirectional long short-term memory models and auxiliary loss. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 412–418.

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.

Eugen Ruppert, Abhishek Kumar, and Chris Biemann. 2017. LT-ABSA: An extensible open-source system for document-level and aspect-based sentiment analysis. In *Proceedings of the GermEval 2017 – Shared Task on Aspect-based Sentiment in Social Media Customer Feedback*, pages 55–60, Berlin, Germany.

Erik F Tjong Kim Sang and Jorn Veenstra. 1999. Representing Text Chunks. In *Proceedings of EACL99*. Bergen, Norway.

Michael Wojatzki, Eugen Ruppert, Sarah Holschneider, Torsten Zesch, and Chris Biemann. 2017. GermEval 2017: Shared Task on Aspect-based Sentiment in Social Media Customer Feedback. In *Proceedings of the GermEval 2017 – Shared Task on Aspect-based Sentiment in Social Media Customer Feedback*, pages 1–12, Berlin, Germany.