Target-Based Sentiment Analysis as a Sequence-Tagging Task*

Zoe Gerolemou and Johannes C. Scholtes

Department of Knowledge Engineering (DKE), Faculty of Science and Engineering, Maastricht University, The Netherlands

Abstract. By focusing on the online-reviews domain, this study aims to provide a complete solution to the sentiment-analysis task consisting off its three constituent components: opinion holder, polarity of the underlying sentiment and target. For the purposes of this research, several challenges and issues related to the nature of the problem are addressed such as class imbalance and the need for meaningful linguistic data-augmentation techniques to increase the size of the training set and make the use of Long Short-Term Memory models (LSTMs) possible. For both of them, new effective approaches are proposed and evaluated. As a means of quantifying class imbalance, the Minority-to-Majority Ratio (M2MR) is introduced. The two sub tasks of target and polarity detection are tackled using machine-learning means. To support the training process, a new data set, which combined sentences from two different review-based corpora, was constructed. In our research, the best-performing LSTM-based models make use of the context-sensitive BERT embeddings and yield F1-Scores of 0.9263 and 0.8911 over all possible classes for the polarity and target components respectively.

Keywords: Target-Based Sentiment Analysis · Class Imbalance · Data Augmentation

1 Introduction

The aim of this study is to introduce a complete solution to the sentiment-mining problem used for social media analysis, by abstracting it as a sequence-tagging task. The ultimate goal will be to label each individual word in a sequence, based on whether it belongs to any of the sentiment components. For example, given the sentence "*I, really, enjoyed watching 'Bohemian Rhapsody' last night.*" as input, the proposed system will have to detect "I" as the *opinion holder*, "Bohemian Rhapsody" as the *target* and determine that the *sentiment polarity* is "Positive". These three elements together form the so-called "sentiment triple".

In addition to the primary goals of this research, some elementary issues will be addressed. Firstly, an attempt will be made to overcome the problem of data set shortage by following a process of target-oriented annotation on a data set comprised of reviews about products and services. This new data

[*] Supported by ZyLAB Technologies B.V., Amsterdam, the Netherlands

set was named *ZyLABSent* and can be made available upon request. Secondly, the issue of class imbalance and its effects on this particular context will be thoroughly investigated and a number of effective solutions for mitigating it will be presented and assessed. In order to be able to evaluate them, a quantitative measure of class balance will be introduced with different adaptations for use in the sentence- and data set-level. Novel approaches of linguistic augmentation functions will be proposed that can be employed for not only the construction of a more balanced data set, but also for increasing the number of useful training samples. Finally, **BERT** and **flair**, two types of contextual embeddings will be used as part of this research.

2 Related Work

Yang and Cardie [1] attempt to perform fine-grained opinion extraction by identifying the opinion entities (holder and target), the opinion expression as well as the relationships between them. Aiming to differentiate their approach from previous ones, they make use of a joint-inference model instead of a pipeline one. Their model is capable of capturing the internal dependencies between the sub tasks of their solution. The features they extract, reflect the local properties of the possible opinion expressions as well as their syntactic and semantic characteristics. The results of the experimental process show that this joint model is significantly more powerful than the traditional pipelines as well as other models that focus solely on a single aspect of the opinion-extraction process. Their model reaches F1-Scores of 0.6163 and 0.5704 (based on the ground-truth-versus-prediction overlap metric) for the holder and target components, respectively.

A recent study on target-based sentiment analysis is presented by Li et al. [2]. Their work aims to solve the complete sentiment-analysis task by proposing a model which treats the problem in an end-to-end manner using a unified tagging scheme. The implemented system is composed of two stacked Recurrent Neural Networks (RNNs). It is worth noting that they use Long Short-Term Memory models (LSTMs) to benefit from any underlying connections between words in a sequence. The bottom layer attempts the auxiliary task of predicting the target's boundaries. Its output acts as a guideline for the upper layer which performs the primary task of estimating the appropriate labels for target-based sentiment analysis.

In addition to the above, Li et al.'s architecture is enriched by three custom-made components: Boundary Guidance (**BG**), Sentiment Consistency (**SC**) and Opinion-Enhanced (**OE**) Target-Word Detection. **BG** utilises information about the target boundaries by modeling all the constraint transitions from target boundaries to target polarities explicitly, in the form of a transition matrix which is used for determining their proportions based on the underlying confidence of the target boundary tagger later on. **SC** makes use of a gate mechanism which combines the features extracted from the current word with its predecessor ones in the sequence, so that it is less likely for tokens belonging to the same target to end up being allocated with different sentiments. Finally, **OE** attempts to improve the quality of the boundary prediction for the opinion targets, by looking for words that indicate the expression of an opinion in a certain window around each token. Subsequently, it trains an elementary classifier which distinguishes between target and non-target words based on distant supervision.

To evaluate the performance of their model, Li et al. merge the training sets from the data sets provided by the SemEval ABSA challenge [3], which belong to the "Laptop" and "Restaurants" domain with tweets collected and used by Mitchell et al. [4]. GLoVe embeddings [5] are extracted and fine-tuned during training to be employed as features. Then, they compare their proposed architecture with several baselines introduced in previous studies and report that it is capable of achieving state-of-the-art performance. Their full model yields F1-Scores of 0.5790, 0.6980 and 0.4801 over three data sets and exceeds the corresponding results from the previous best-performing configuration (i.e. 0.5619, 0.6638 and 0.4735, respectively).

3 Target-Based Sentiment Analysis

3.1 Word Embeddings for Sentiment Analysis

The advances in the implementation of contextual word embeddings has proved to be of particular importance for tasks in which context really matters such as sentiment analysis. In sentiment analysis, words may be associated with a different meaning based on the context they appear in. Also, not only can there be sudden changes of meaning and sentiment due to different word arrangements, but word-based and syntactical negations can also cause additional levels of ambiguity.

Thus, a natural approach was to employ context-sensitive word embeddings as features in order to be able to capture intrinsic properties and internal relationships between the words that consist the data set. Where Word2Vec [6] and GloVe [5] had issues dealing with the abovementioned requirements for sentiment analysis, more context-sensitive representations such as BERT [7] and flair [8] can better deal with the subtleties required, thereby adding value to the sentiment-analysis task.

3.2 LSTM-Based Models for Sequence Tagging

According to the work by Huang et al. [9], the current-state-of-the-art architecture for sequence-tagging tasks leverages Bidirectional LSTMs followed by a Conditional Random Fields (CRF) layer. This model combines both the strong abilities of LSTMs to capture dependencies in the dimension of time and the aptitude of CRFs to identify possible transitions in sequences.

Akbik et al. [8] who developed the **flair** framework[1] have also enriched their library with the ability to perform sequence-tagging tasks using this very architecture by employing different word embeddings as features. This framework was used for the purposes of the following experiments.

[1] http://github.com/zalandoresearch/flair/

In our initial experiments, different class imbalance handling techniques, various combinations of word embeddings as well as some tweaks on the configuration of the model architecture (i.e. the inclusion of the CRF layer or not) were applied and compared in order to determine the optimal configuration for the purposes of this task. We will discuss this in more detail in the results section.

4 Data set for Sentiment Target and Polarity Detection

The main focus of this research is on the creation of a data set that was sufficiently large to allow for the use of deep-learning models. The only publicly-available[2] data set that includes target-oriented annotations is the "Restaurant" domain of the *SemEval-2016 Task 5: "Aspect-Based Sentiment Analysis"* data set [3] (thus, referred to as *SemEval-2016-Task-5*). Unfortunately, it was not possible to solely rely on it, due to a variety of reasons. Firstly, the data set consists of sentences that originate only from the "Restaurant" domain. Therefore, they contain words from a limited vocabulary related to this specific context, affecting the ability to generalise over opinionated sentences from different domains. Secondly, due to its relatively small size, it could not be used effectively for training deep-learning models.

A research team from Johns Hopkins University collected Amazon Reviews from several product categories (four domains in the first release[3] and twenty-five in the second one[4]): the *Multi-Domain Sentiment Data set*. The reviews are accompanied by their star rating (from one up to five stars). Reviews with less than three stars are labelled as "Negative" whilst the ones that contain more than three are marked as "Positive". Three stars indicate a neutral review.

Unfortunately, the *Multi-Domain* data set is characterised by two important limitations: (i) the scope of sentiment annotations is provided in a per-review and not in a per-sentence manner. So, when considering each sentence individually, it is highly possible to come across sentences whose polarity disagrees with the overall review polarity. For example, in a negative review, there can be a sentence which expresses a positive opinion, in contrast with the rest. This may become even more acute with the use of cynical language. Thus, it is not safe to deduce that the polarity of each sentence matches the global polarity of the review it belongs to. (ii) The data set states the product for which a review is about, but the same does not hold for each individual sentence. It is plausible that a sentence may include an opinion related to a different entity than the rest of the review.

Despite the fact that the aforementioned data sets were not suitable as standalone solutions on their own, their union could be useful for training an effective model. In particular, *SemEval-2016-Task-5* and *Multi-Domain* are pretty similar since they are both comprised of reviews. On the other hand, they originate from different domains, which means that they could introduce some variety to

[2] http://alt.qcri.org/semeval2016/task5/

[3] http://cs.jhu.edu/~mdredze/datasets/sentiment/index2.html

[4] http://cs.jhu.edu/~mdredze/datasets/sentiment/

the models during training. Unlike any social media postings which are written in a more quick and informal way, reviews tend to resemble the formal way of writing to a higher degree due to the fact that their authors aim to associate their writing with more credibility.

Therefore, based on the above factors, a new data set (thus, referred to as *ZyLAB-Targeted-Sentiment-Reviews* or *ZyLABSent*) was compiled, which consists of subsets from these two existing data sets. In order to overcome the restrictions affecting the *Multi-Domain* one, a manual annotation process was initiated on a set of 4000 sentences extracted from the *Multi-Domain* corpus. For the given sentences, the annotators were asked to detect the entities that appear to be targets of sentiment; either positive or negative. Additionally, they were informed on the polarity of the review from which a sentence originated, as an indication of the overall context. It was stressed that the "entities to-be-detected" should appear as tokens in the sentence. If a sentence was associated with a target entity of a specific sentiment that could be inferred from the context, but did not appear *explicitly* in the sentence, the corresponding section should have been marked by the "[Unknown]" token. Also, in case a sentence did not include an element of a specific polarity, this section should have been filled in with the "[None]" token. The Kappa measure for inter-judge agreement between the two annotators was equal to **0.7245**.

5 Addressing Class Imbalance

5.1 Minority-to-Majority Ratio (M2MR)

Approaching the task as a sequence-tagging problem and representing the input accordingly resulted to some interesting effects related to the distribution of the different class labels. Originating from the specifics of the problem domain itself, the fact that the focus of the different sub tasks is only on certain sentence tokens (i.e. words that represent the target entities) means that the vast majority of tokens are expected to be classified as "Non-Token" or "Neutral". A serious consequence of this, from a machine-learning point-of-view, is class imbalance which is evident by the abundance of these two classes over the rest. This could possibly result to models that are biased towards predicting these majority classes offering them an unfair advantage.

However, before proceeding with following any approaches for tackling class imbalance, it was important to assess its severity on this specific case and estimate any direct impacts that it could have on the performance of the prediction processes. The first step was to ensure whether the data set was indeed imbalanced and to which degree, whilst the latter could confirm whether the possible class imbalance would hinder the classification procedures significantly and determine whether it was worth investing time handling it.

Since, by the time this research was conducted, there was no standard way to measure class imbalance for sequence-tagging tasks in the literature, a new metric was defined aiming to quantify this phenomenon:

Let s be a sentence originating from a text corpus c. c contains m sentences. The index of each sentence in the corpus is denoted by i. Each of these sentences consists of word tokens associated with one out of l labels corresponding to the possible output classes of a sequence-tagging task. maj_s is the set of sentence tokens which are associated with a label belonging to the majority class and min_s is the union of tokens assigned with labels originating from the other $l-1$ classes that together consist the minority classes. We introduce the **Minority-to-Majority Ratio (M2MR)** measure of a sentence ($M2MR_s$), which is defined as the ratio of the size of the latter over the size of the former (1). To construct a global metric that quantifies the overall balance of a text corpus, the $M2MR_c$ is also defined which contains an additional multiplicative parameter α corresponding to the inverse of m (2).

$$M2MR_s = \begin{cases} \frac{|min_s|}{|maj_s|}, maj_s >= min_s > 0 \\ 0, \qquad otherwise \end{cases} \tag{1}$$

$$M2MR_c = \alpha \times \sum_{i=0}^{m-1} M2MR_{c_i} = \frac{1}{m} \times \sum_{i=0}^{m-1} M2MR_{c_i} \tag{2}$$

By definition, the size of the majority class should be greater than or equal to the size of the union set of the minority classes. The equal case corresponds to the scenario of a balanced sentence that would yield an $M2MR_s$ of one. In the case when min_s is equal to zero, the $M2MR_s$ becomes zero immediately due to the fact that the complete absence of minority-class tokens from a sentence signifies a state of full imbalance. If maj_s is zero, it means that min_s is also zero which is only possible when an empty sentence with no tokens is considered. In this scenario the $M2MR_s$ becomes zero, too.

The use of the α parameter in the corpus-wide version guarantees that a perfectly-balanced data set (i.e. a data set that consists solely balanced sentences) would receive an $M2MR_c$ score of one which is the maximum value this formula can take. Therefore, the optimisation objective when trying to construct a balanced data set can be thought of as the attempt to maximise the score of $M2MR_s$ for each individual sentence or the $M2MR_c$ for the entire data set.

$M2MR_s/M2MR_c \to 0$	Highly Imbalanced Sentence/Corpus
$M2MR_s/M2MR_c \to 1$	Highly Balanced Sentence/Corpus

5.2 Ideally-Balanced Data set (Data Configuration 1 or DC1)

A sequence-tagging data set consists of several sentences containing a varying amount of tokens (i.e. words) each. These tokens, being annotated for a particular purpose, are highly likely to be associated with a class that significantly outnumbers the rest. However, using the $M2MR$ metric, it is possible to select the slice of this sentence which offers the most balanced configuration using the available tokens. Attempting to balance each sentence individually can help mitigating the overall class-imbalance issue which characterises the entire data set.

In order to find the optimal slice for each sentence, all possible ordered slices with sizes ranging from two up to the length of the entire sentence were generated. Slices of size one were not considered due to the fact that such slice is basically a word and thus it no longer corresponds to a proper sequence. Subsequently, the $M2MR_s$ value of each slice was calculated and, after sorting the slices in descending order based on it, a ranking was constructed. To provide an additional boost to slices which were both balanced, but also retained the largest amount of original tokens possible, the slices were grouped based on their $M2MR_s$ score and internally re-sorted based on their length. The slice that made it to the top of this refined ranking bore the highest $M2MR_s$ of all and it was chosen to represent the sentence in the corpus. This process was repeated for all the sentences resulting to a balanced version of the original data set.

However, this approach could be associated with an important limitation. Since, it aimed to arrange the class distribution in the most optimal way, it could demonstrate a tendency towards building a data set which was ideally balanced, but not realistic anymore. For example, if most of the sentences originally contained a single token that was allocated with a minority-class label then a direct consequence of this would be that most of the sentences would end up being replaced by a slice of size two. Thus, if this data set was used for any training purposes, it might result in a biased model which would demonstrate an aptitude towards labelling unseen sentences of this length, but not on sentences of different sizes.

6 Data Augmentation

6.1 Proportion-Based Probabilistic Sampling (DC2)

To mitigate the abovementioned issue of unrealistic distribution, a generalised version of the ideally-balanced data set construction could be proved efficient. Instead of constantly favouring the slices with the highest $M2MR_s$, these slices were simply associated with higher probabilities of being drawn whilst the selection was kept completely random. We named this Data Configuration 2 (DC2).

The steps up to and including the ranking process that were described in the previous balancing technique were also followed for this approach. Then, the ranking was split into four leagues based on the sorting order and the members of each league were replicated in the data set based on them. More specifically, samples belonging to the top league were copied four times, tripled in the second league, doubled in the third and remained unchanged in the last league.

Finally, a random slice (or a number of slices) was chosen from the list. This approach still benefited the balanced slices by giving them more chances of being selected while it introduced the probabilistic element which would lead to a more "realistic" setting. A new version of the data set would be created that is more likely to be balanced than the original one, but still with higher resemblance to a real data set than the ideally-balanced one.

6.2 Synonym-Based Minority Class Augmentation (DC3)

The second approach was based on the creation of new samples by means of data synthesis. To accomplish this, all sentences were iterated over, and their tokens which were annotated as targets of sentiment (i.e. assigned with labels from the minority classes), were detected. Based on the idea that a synonym is, most of the times, capable of conveying the same meaning as the original word without altering the quality of the overall sentence, new copies of the sentences were fabricated by simply replacing the target tokens with their synonyms. One could argue that due to the use of word embeddings, adding additional training data based on synonyms would not add sufficient additional information for the machine-learning process. For this reason, we did not only use strict synonyms, but we also experimented with other methods to generate linguistically and semantically valid training samples.

To be able to identify all the possible synonyms for the different target words, the WordNet lexical database for English [5] was utilised. The target tokens were looked up in the aforementioned resource for synonyms. However, there was some variety on the lemmas returned for a few words. In the English language, words such as "work" can be used in either verb or noun form. In order to tackle this, extracted lemmas that were associated with a different Part-of-Speech (PoS) tag than the original target token were discarded. For their identification, the spaCy PoS-tagging model [6] was employed. The remaining lemmas belonged to synonyms matching the syntactic role of the initial word. As an additional clearance measure, any synonyms which appeared in a different form than the original word (i.e. synonyms in plural while the word itself was in singular and the converse) were disposed to avoid the inclusion of meaningless information in the training data set. Subsequently, the selected synonyms were incorporated on the original sentences and allocated with the same labels as their predecessors, resulting in Data Configuration 3 (DC3).

It is worth noting that the outsourced lemmas are not always strict synonyms of the original words. WordNet also looks up for any hypernyms, hyponyms and older forms of a word. These words albeit quite similar, they do not always have an identical meaning. For example, for the word "film", WordNet returns both "movie" and "picture". However, the use of the latter will result to a sentence with a completely new meaning which can be interpreted in a very different way. This was expected to introduce new content and supply the trained classifiers with more variability.

[5] http://wordnet.princeton.edu/

[6] https://spacy.io/usage/linguistic-features#pos-tagging

6.3 Antonym-Based Sentiment-Bearing Word Augmentation

In a similar manner, it was also possible to fabricate new sentences by inverting the meaning of the existing ones. The main difference with the abovementioned synonym-based approach was that the tokens of interest were not the target tokens themselves, but any adjectives or adverbs that were related to them. Their selection was based on the rationale that these are the ones which actually bear the sentiment towards the target. They were identified by employing dependency parsing to resolve any relationships between the target and other sentence tokens characterised by these two PoS tags. The Dependency Parser [7] which is part of spaCy processing pipeline was utilised for this purpose. Two different ways of using antonym-based augmentation were investigated. These approaches will be explained in the next subsections.

Not-Based Inversion (DC4) The first and simplest approach involved the inversion of meaning with the use of the adverb "not". In order to change the polarity of sentiment, it is usually sufficient to flip the polarity of the aforementioned sentiment-bearing words with the use of "not". Equivalently, for the ones which were already preceded by "not", this special negation token was removed to cause the change of sentiment. Due to the aptitude of context-sensitive embeddings in appreciating the surroundings of words, the scope of negation would be appropriately interpreted and handled during the encoding.

Lexical-Antonyms Replacement (DC5) For the more complicated method of inverting the sentiment of these words with their lexical antonyms, an approach quite similar to the synonym-based augmentation was followed. Again, the WordNet lexical database was utilised to look up these words for all their possible lemmas. After the appropriate check for matching PoS tags between the original word and its antonyms, the existing sentences were replicated with the chosen antonyms replacing the sentiment-bearing words.

In a similar manner as its synonym counterpart, the augmentation of the data set with antonyms and the subsequent inversion of polarity introduced new examples of possible sentimental relationships. In this case, in particular, a novel category of training data was created since the different antonyms exploit previously unseen instances of text that express the opposite polarity than their source sentence.

An example that illustrates the results from the application of the different augmentation processes on an actual sentence is provided below:

- **Original Sentence:** "This film is terrible."
- **Synonym-Based:** "This movie is terrible.", "This motion picture is terrible.", "This picture is terrible."
- **Antonym-Not-Based:** "This movie is not terrible."
- **Antonym-Lexical-Based:** "This movie is wonderful."

[7] https://spacy.io/api/dependencyparser

7 Opinion Holder Extraction

Extraction of the opinion holder was not the main topic of this research project. However, in order to provide a complete solution for the sentiment-extraction problem, we also decided to address this aspect of the problem. In contrast to the detection of the sentiment target and polarity, the identification of the opinion holder can be thought as a fairly simple process in the context of online reviews. After an exploratory analysis of the data set samples, it was possible to deduce a few distinct types of opinion-holder occurrences in the sentences. Therefore, instead of following a supervised approach as with the other sentiment constituents, a rule-based heuristic procedure was followed with the expectation of receiving results of high quality.

The first type of opinion-holder occurrence, concerned the scenario when a sentence lacks a direct reference to the opinion holder. For example, in the sentence "*This film was simply amazing.*", the entity that expresses this positive opinion is not mentioned explicitly anywhere. Therefore, it is safe to infer that the opinion holder here is/are the author(s) of this sentence. In our model, the generic "(Author)" token implies either a single or multiple authors.

In other cases, the opinion holders may be explicitly referenced. These situations can be generally broken down into two categories: **first-person** thoughts or experiences and **other-people** views or perceptions. An instance of the first class is a sentence in which an opinion or experience is described by the speakers themselves. Here, the identity of the speaker is evident by the appearance of first-person pronouns (e.g. *I*, *We*). The second class includes all sentences in which opinions or perceptions of other people are quoted by their author. Therefore, the opinion holders (or *"subjects"*) in this scenario are these third-party entities.

In order to handle the aforementioned scenarios, a set of rules was defined which, when satisfied, they classified each sentence to the most suitable opinion-holder category and treated it appropriately in order to extract the entity of interest. Before this, any personal pronouns were resolved to their subjective form (e.g. **my** → **I**). Starting off with the most general scenario of considering the "(Author)" as the opinion holder, dependency parsing was applied to detect whether the opinion holder corresponded to any of the more specific categories. This was accomplished by investigating the structure of the sentence and exploiting any noun phrase - thinking/perceiving verb dependencies which were expected to indicate the expression of an opinion by the first constituent of these syntactic relationships. Here, the detected subject was pronounced as the opinion holder. Appropriate refinements to these rules were made when the entities of interest contained multiple tokens or when they expressed a conjugation of many entities. Lastly, as a final refinement step, co-reference resolution was applied to uncover the existence of any co-referring clusters. The Neuralcoref module [8] of spaCy NLP engine was used for this which performs co-reference resolution by employing a scoring model based on a neural network proposed by Clark and Manning [10]. In case the detected subject belonged to any of these

[8] https://spacy.io/universe/project/neuralcoref

clusters, it was replaced by its head. For clarity, the entire process is presented below.

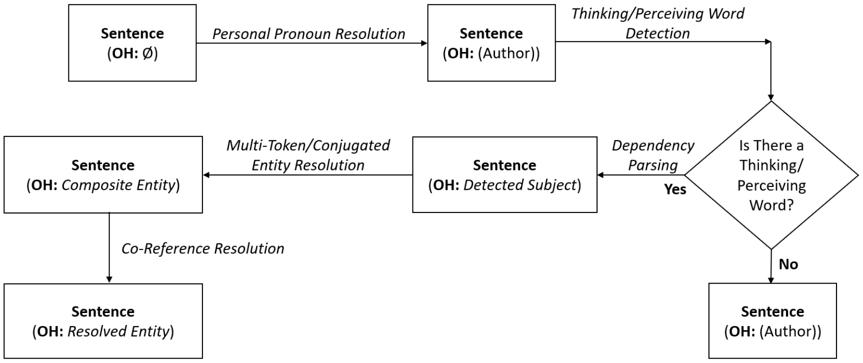

Fig. 1. A flowchart depicting the workflow for opinion holder extraction.

8 Results

8.1 Performance Evaluation

Due to the classification nature of the different sub tasks, F1-Score was used as the evaluation metric. This was also the metric used by most of the previous studies. However, it was hard to directly compare them with this experimental iteration due to the fact that they tackled the sentiment-analysis problem in different ways (and not as a sequence-tagging problem) and made use of different annotated data. Thus, for the overall performance of each approach, Micro-Averaged F1-Scores ($F1_\mu$) were calculated because, by definition, this measure takes class imbalance into consideration and reflects it accordingly on the results. Additionally, this metric was used as the loss function for training all the LSTM-based models.

It is important to note that, although $F1_\mu$ considers the class distribution, it can overshadow the poor performance of smaller classes completely, if the majority class performs significantly well. Of course, this is unfair to the minority classes since their low individual scores would be masked by a high overall metric and thus they would end up being overlooked during the optimisation of a model.

Therefore, a more fair, Weighted F1-Score ($F1_w$) was additionally considered for the two machine-learning-based components which assigned weights to every class. These weights were inversely proportional to their frequency in the data set. This means that the smaller the number of instances from a given class appearing in the data set was, the larger the weight that was associated with it would be.

8.2 Experiment 1 - Data Augmentation Approaches

In order to compare the effectiveness of the different augmentative approaches described earlier, each of them was applied on 80% of the *ZyLABSent* dataset (Data Configuration 0 - **DC0**), which was subsequently used for training the two models. The additional 20% was set apart for testing and was not augmented to avoid any side effects from the augmentation on the training results. Apart from the individual approaches described in the previous sections, two additional Data Configurations (DC) were applied as well: Combined Antonym Augmentation (**DC6**) and Combined Synonym and Antonym Augmentation (**DC7**).

In addition to $F1_\mu$ and $F1_w$, the $M2MR_c$ score of the training corpus was also calculated to uncover possible correlations between the data set balance and the classification performance. Along with them, the total number of data set sentences per experiment (**NoS**), after the application of each approach was also counted.

As it can be seen on Table 1, the ideally-balanced scenario resulted in the worst performance. This can be explained by the fact that this configuration is not realistic, as it was explained earlier. The classifier was trained to label balanced sentences, but this optimal distribution did not characterise the test split which was still highly imbalanced ($M2MR_c = 0.1541$).

When it came to the different data augmentation techniques, they all led to a more balanced state when compared to the original data set. However, higher balance did not directly correlate with better performance. The Proportion-Based approach, albeit the most competent in terms of balancing the data set, did not result to the best classification performance for either models. On the contrary, two approaches that involved the generation of new sentences using antonym and synonym replacement, were proved the most effective. The combination of the two antonym-based techniques was the best-performing one for the "Target" model ($F1_w = 0.6478$) whilst the synonym-based augmentation outperformed the rest ($F1_w = 0.7808$) for the "Polarity" one.

	Target				**Polarity**			
DC#	**NoS**	$M2MR_c$	$F1_\mu$	$F1_w$	**NoS**	$M2MR_c$	$F1_\mu$	$F1_w$
DC0	4000	0.1535	0.9227	0.6395	4000	0.1535	0.9063	0.5219
DC1	4000	1.0000	0.6806	0.4335	4000	1.0000	0.6449	0.5202
DC2	4000	0.4227	0.9001	0.6397	4000	0.4227	0.8792	0.5465
DC3	8565	0.2753	0.8975	0.5762	8527	0.2736	0.8911	**0.7808**
DC4	6361	0.1928	0.9160	0.6065	6338	0.2069	0.8920	0.7681
DC5	5228	0.1852	0.9156	0.5800	5226	0.1950	0.8985	0.5254
DC6	7589	0.2082	0.9204	**0.6478**	7564	0.2235	0.8939	0.5911
DC7	12154	0.2735	0.9004	0.6106	12091	0.2923	0.8894	0.7031

Table 1. The resulting M2MRs and F1-Scores from the experimentation with different data augmentation approaches.

8.3 Experiment 2 - Type of Embeddings

Between the two types of contextual embeddings namely BERT and flair, the former resulted to higher $F1_w$ than the latter for both components (0.6478 and

0.7808, respectively). Additionally, in order to prove the superiority of BERT over context-insensitive word embeddings, one configuration which employed GloVe was also used. Indeed, the BERT-based models outperformed it by more than 6% in terms of $F1_w$ for both models.

	Target		**Polarity**	
Type of Embeddings	$F1_\mu$	$F1_w$	$F1_\mu$	$F1_w$
BERT	0.9204	**0.6478**	0.8911	**0.7808**
flair	0.8938	0.6105	0.8910	0.6214
GloVe	0.9127	0.5805	0.8965	0.6252

Table 2. The resulting F1-Scores from the experimentation with different types of embeddings.

8.4 Experiment 3 - Use of a CRF layer

The next experiment aimed to investigate whether the inclusion of the extra CRF layer on top of the Bidirectional LSTM would result to any additional performance gains for the classification process. Without the use of this layer, the final classification was performed using softmax. Otherwise, the CRF unit undertook the classification process itself. As the experiments demonstrated, the configuration which included a CRF component resulted to an increased performance for the "Target" model, but not for the "Polarity" one.

	Target		**Polarity**	
Is a CRF layer used?	$F1_\mu$	$F1_w$	$F1_\mu$	$F1_w$
No	0.9204	0.6478	0.8911	**0.7808**
Yes	0.9263	**0.6741**	0.8883	0.6573

Table 3. The resulting F1-Scores from the experimentation with the use of a CRF layer.

8.5 Experiment 4 - Opinion Holder Extraction

In order to be able to evaluate the performance of the opinion holder extraction, a set of 100 randomly sampled sentences from *ZyLABSent* was manually annotated with respect to the person who expressed the opinions they included. To get an idea of its performance in a per-sentence manner, the ground-truth values along with the predicted ones were compared one by one. For 96 out of the 100 sentences (i.e. a success rate of **96%**), these two values matched completely.

Furthermore, the performance was evaluated in a per-token basis. A new representation of the sentences was constructed where each word was tagged with either "H" or "NH" depending on whether it was part of the detected opinion-holding-entity string or not. Finally, the labelled sentence was compared with the ground-truth version of it. The per-token results showed F1-Scores of **0.9200** and **0.9970** for the "H" and "NH" classes, respectively, and an $F1_\mu$ of **0.9940**.

An example that illustrates the results from the application of the complete sentiment-extraction process on an actual sentence is provided below:

- **Original Sentence:** “This video has exceeded our expectations.”
- **Labelled Sentence:** {“This”: (T, POS, NH), “video”: (T, POS, NH), “has”: (NT, NEU, NH), “exceeded”: (NT, NEU, NH), “our”: (NT, NEU, H), “expectations”: (NT, NEU, NH)}
- **Sentiment Triple:** (“We”, POS, “this video”)

9 Conclusions

This study aimed to provide a complete solution to the challenging task of sentiment analysis by breaking it down to its three fundamental components: opinion holder, sentiment’s polarity and target. Due to the straightforward character of the opinion holder extraction, this could be tackled using a simple rule-based method. The last two were approached by means of supervised learning. The result from the application of full sentiment analysis on a sentence, was a sentiment triple consisting of the aforementioned elements.

An amalgamation of two publicly-available review-based data sets from various domains was used as the development corpus. 4000 sentences from it were manually annotated by different individuals. The **Kappa** measure of this effort was 0.7450, which shows an almost strong agreement between the annotators.

Different class-balancing approaches were introduced and explained thoroughly in this study and subsequently evaluated via an experimental procedure. In addition to class balancing, these techniques could also be used for augmenting a data set with new samples and introducing additional variability to the models. Antonym-Based and Synonym-Based Augmentation were the best performing approaches that resulted to both a more balanced data set configuration, but also a significant increase in the classification performance.

The incorporation of synthetic data enriched the data set with more useful information. The described experiments have proved that these new training samples acted as an improvement to the already well-performing contextual embeddings. Even though such embeddings are capable of capturing the semantic relationship between words with the same meaning, these new augmentative approaches introduced more variety by constructing similar sentences, but with slightly different or completely opposite meaning. These solutions were accompanied with the definition of several metrics that aimed to quantify and evaluate class imbalance. By using these methods, we were able to triple the size of a labelled set of 4000 sentences. We believe that there are other linguistic transformation functions that will allow us to add even more useful sentences. This will be a subject of future research for us.

Apart from the different data set configurations mentioned earlier, many experiments were conducted involving various types of contextual and context-insensitive embeddings and the inclusion of a CRF decoding layer on top of the existing architecture. As far as the different contextual embeddings are concerned, BERT and flair were used and compared with each other for the very first time (as far as it is known) for sentiment-analysis purposes on the token level since their publication. The best-performing models for the “Target” and “Polarity” subtasks have reached a Micro-Averaged F1-Score of 0.9263 and 0.8911, respectively, and made use of the BERT embeddings. The inclusion of a CRF layer yielded better results only for the "Target" model.

Although this was not the primary focus of our research, to complete our solution for the sentiment-extraction problem, the opinion-holder sub task was tackled by devising a rule-based approach that leveraged syntactical and grammatical patterns which appear in opinionated sentences frequently. The model managed to identify the opinion holder correctly in 96% of the sentences. This high score is expected considering that the detection of the opinion holder is a relatively easy task compared to the other two components and can be mapped to a small number of scenarios and handled appropriately.

Acknowledgements

We would like to thank ZyLAB for their support throughout the course of this research. This included, but was not limited to, help during the data set annotation, provision of computational units for executing the conducted experiments and valuable advice.

References

1. Yang, B., Cardie, C.: Joint Inference for Fine-grained Opinion Extraction. Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics **1**(Long Papers), 1640–1649 (2013)
2. Li, X., Bing, L., Li, P., Lam, W.: A Unified Model for Opinion Target Extraction and Target Sentiment Prediction. CoRR **abs/1811.05082**, 1–9 (2018)
3. Pontiki, M., Galanis, D., Papageorgiou, H., Androutsopoulos, I., Manandhar, S., AL-Smadi, M., Al-Ayyoub, M., Zhao, Y., Qin, B., De Clercq, O., Hoste, V., Apidianaki, M., Tannier, X., Loukachevitch, N., Kotelnikov, E., Bel, N., Jimenez-Zafra, S. M., Eryigit, G.: SemEval-2016 Task 5 Aspect Based Sentiment Analysis. Proceedings of SemEval-2016, 19–30 (2016)
4. Mitchell, M., Aguilar, J., Wilson, T., Van Durme, B.: Open Domain Targeted Sentiment. Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, 1643-1654 (2013)
5. Pennington, J., Socher, R., Manning, C.: Glove: Global Vectors for Word Representation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 1532–1543 (2014)
6. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient Estimation of Word Representations in Vector Space. 1st International Conference on Learning Representations, 1–12 (2013)
7. Devlin, J., Chang, M.-W., Lee, K., Toutanova, K.: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. CoRR, 1–14 (2018)
8. Akbik, A., Blythe, D., Vollgraf, R.: Contextual String Embeddings for Sequence Labeling. COLING 2018, 27th International Conference on Computational Linguistics, 1638–1649 (2018)
9. Huang, Z., Xu, W., Yu, K.: Bidirectional LSTM-CRF Models for Sequence Tagging. CoRR **abs/1508.01991**, 1–10 (2015)
10. Clark, K., Manning, C. D.: Deep Reinforcement Learning for Mention-Ranking Coreference Models. CoRR **abs/1609.08667**, 1–6 (2016)