Abstract

This paper discusses the fourth year of the "Sentiment Analysis in Twitter Task". SemEval-2016 Task 4 comprises five subtasks, three of which represent a significant departure from previous editions. The first two subtasks are reruns from prior years and ask to predict the overall sentiment, and the sentiment towards a topic in a tweet. The three new subtasks focus on two variants of the basic “sentiment classification in Twitter” task. The first variant adopts a five-point scale, which confers an ordinal character to the classification task. The second variant focuses on the correct estimation of the prevalence of each class of interest, a task which has been called quantification in the supervised learning literature. The task continues to be very popular, attracting a total of 43 teams.

1 Introduction

Sentiment classification is the task of detecting whether a textual item (e.g., a product review, a blog post, an editorial, etc.) expresses a POSITIVE or a NEGATIVE opinion in general or about a given entity, e.g., a product, a person, a political party, or a policy. Sentiment classification has become a ubiquitous enabling technology in the Twittersphere. Classifying tweets according to sentiment has many applications in political science, social sciences, market research, and many others (Martínez-Cámara et al., 2014 [Mejova et al., 2015]).

As a testament to the prominence of research on sentiment analysis in Twitter, the tweet sentiment classification (TSC) task has attracted the highest number of participants in the last three SemEval campaigns (Nakov et al., 2013; Rosenthal et al., 2014; Rosenthal et al., 2015; Nakov et al., 2016b).

Previous editions of the SemEval task involved binary (POSITIVE vs. NEGATIVE) or single-label multi-class classification (SLMC) when a NEUTRAL class is added (POSITIVE vs. NEGATIVE vs. NEUTRAL). SemEval-2016 Task 4 represents a significant departure from these previous editions. Although two of the subtasks (Subtasks A and B) are reincarnations of previous editions (SLMC classification for Subtask A, binary classification for Subtask B), SemEval-2016 Task 4 introduces two completely new problems, taken individually (Subtasks C and D) and in combination (Subtask E):

1.1 Ordinal Classification

We replace the two- or three-point scale with a five-point scale \{HIGHLYPOSITIVE, POSITIVE, NEUTRAL, NEGATIVE, HIGHLYNEGATIVE\}, which is now ubiquitous in the corporate world where human ratings are involved: e.g., Amazon, TripAdvisor, and Yelp, all use a five-point scale for rating sentiment towards products, hotels, and restaurants.

Moving from a categorical two/three-point scale to an ordered five-point scale means, in machine learning terms, moving from binary to ordinal classification (a.k.a. ordinal regression).

\footnote{Fabrizio Sebastiani is currently on leave from Consiglio Nazionale delle Ricerche, Italy.}
1.2 Quantification

We replace classification with quantification, i.e., supervised class prevalence estimation. With regard to Twitter, hardly anyone is interested in whether a specific person has a positive or a negative view of the topic. Rather, applications look at estimating the prevalence of positive and negative tweets about a given topic. Most (if not all) tweet sentiment classification studies conducted within political science (Borge-Holthoefer et al., 2015; Kaya et al., 2013; Marchetti-Bowick and Chambers, 2012), economics (Bollen et al., 2011; O'Connor et al., 2010), social science (Dodds et al., 2011), and market research (Burton and Soboleva, 2011; Qureshi et al., 2013), use Twitter with an interest in aggregate data and not in individual classifications.

Estimating prevalences (more generally, estimating the distribution of the classes in a set of unlabelled items) by leveraging training data is called quantification in data mining and related fields. Previous work has argued that quantification is not a mere byproduct of classification, since (a) a good classifier is not necessarily a good quantifier, and vice versa, see, e.g., (Forman, 2008); (b) quantification requires evaluation measures different from classification. Quantification-specific learning approaches have been proposed over the years; Sections 2 and 5 of (Esuli and Sebastiani, 2015) contain several pointers to such literature.

Note that, in Subtasks B to E, tweets come labelled with the topic they are about and participants need not classify whether a tweet is about a given topic. A topic can be anything that people express opinions about; for example, a product (e.g., iPhone6), a political candidate (e.g., Hillary Clinton), a policy (e.g., Obamacare), an event (e.g., the Pope’s visit to Palestine), etc.

The rest of the paper is structured as follows. In Section 2, we give a general overview of SemEval-2016 Task 4 and the five subtasks. Section 3 focuses on the datasets, and on the data generation procedure. In Section 4, we describe in detail the evaluation measures for each subtask. Section 5 discusses the results of the evaluation and the techniques and tools that the top-ranked participants used. Section 6 concludes, discussing the lessons learned and some possible ideas for a followup at SemEval-2017.

2 Task Definition

SemEval-2016 Task 4 consists of five subtasks:

1. **Subtask A**: Given a tweet, predict whether it is of positive, negative, or neutral sentiment.

2. **Subtask B**: Given a tweet known to be about a given topic, predict whether it conveys a positive or a negative sentiment towards the topic.

3. **Subtask C**: Given a tweet known to be about a given topic, estimate the sentiment it conveys towards the topic on a five-point scale ranging from HIGHLY NEGATIVE to HIGHLY POSITIVE.

4. **Subtask D**: Given a set of tweets known to be about a given topic, estimate the distribution of the tweets in the POSITIVE and NEGATIVE classes.

5. **Subtask E**: Given a set of tweets known to be about a given topic, estimate the distribution of the tweets across the five classes of a five-point scale, ranging from HIGHLY NEGATIVE to HIGHLY POSITIVE.

Subtask A is a rerun – it was present in all three previous editions of the task. In the 2013-2015 editions, it was known as Subtask B. We ran it again this year because it was the most popular subtask in the three previous task editions. It was the most popular subtask this year as well – see Section 5.

Subtask B is a variant of SemEval-2015 Task 10 Subtask C (Rosenthal et al., 2015; Nakov et al., 2016b), with POSITIVE, NEUTRAL, and NEGATIVE as the classification labels.

Subtask E is similar to SemEval-2015 Task 10 Subtask D, which consisted of the following problem: *Given a set of messages on a given topic from the same period of time, classify the overall sentiment towards the topic in these messages as strongly positive, weakly positive, neutral, weakly negative, or strongly negative.* Note that in SemEval-2015 Task 10 Subtask D, exactly one of the five classes had to be chosen, while in our Subtask E, a distribution across the five classes has to be estimated.

Note that we retired the expression-level subtask A, which was present in SemEval 2013–2015 (Nakov et al., 2013; Rosenthal et al., 2014; Rosenthal et al., 2015; Nakov et al., 2016b).
As per the above discussion, Subtasks B to E are new. Conceptually, they form a $2 \times 2$ matrix, as shown in Table 1, where the rows indicate the goal of the task (classification vs. quantification) and the columns indicate the granularity of the task (two- vs. five-point scale).

| Goal      | Granularity |     |     |
|-----------|-------------|-----|-----|
|           | Two-point   | Five-point |
|           | (binary)    | (ordinal) |
| Classification | Subtask B | Subtask C |     |
| Quantification | Subtask D | Subtask E |     |

Table 1: A $2 \times 2$ matrix summarizing the similarities and the differences between Subtasks B-E.

3 Datasets

In this section, we describe the process of collection and annotation of the training, development and testing tweets for all five subtasks. We dub this dataset the Tweet 2016 dataset in order to distinguish it from datasets generated in previous editions of the task.

3.1 Tweet Collection

We provided the datasets from the previous editions (see Table 2) of this task (Nakov et al., 2013; Rosenthal et al., 2014; Rosenthal et al., 2015; Nakov et al., 2016b) for training and development. In addition we created new training and testing datasets.

The training and development tweets were collected from July to October 2015. The test tweets were collected from October to December 2015. We used the public streaming Twitter API to download the tweets.

We then manually filtered the resulting tweets to obtain a set of 200 meaningful topics with at least 100 tweets each (after filtering out near-duplicates). We excluded topics that were incomprehensible, ambiguous (e.g., Barcelona, which is the name both of a city and of a sports team), or too general (e.g., Paris, which is the name of a big city). We then discarded tweets that were just mentioning the topic but were not really about the topic.

Note that the topics in the training and in the test sets do not overlap, i.e., the test set consists of tweets about topics different from the topics the training and development tweets are about.

3.2 Annotation

The 2016 data consisted of four parts: TRAIN (for training models), DEV (for tuning models), DEVTEST (for development-time evaluation), and TEST (for the official evaluation). The first three datasets were annotated using Amazon’s Mechanical Turk, while the TEST dataset was annotated on CrowdFlower.

We employed the following annotation procedure. As in previous years, we first gathered tweets that express sentiment about popular topics. For this purpose, we extracted named entities from millions of tweets, using a Twitter-tuned named entity recognition system (Ritter et al., 2011). The collected tweets were greatly skewed towards the neutral class. In order to reduce the class imbalance, we removed those that contained no sentiment-bearing words. We used SentiWordNet 3.0 (Baccianella et al., 2010) as a repository of sentiment words. Any word listed in SentiWordNet 3.0 with at least one sense having a positive or a negative sentiment score greater than 0.3 was considered sentiment-bearing.

The training and development tweets were collected from July to October 2015. The test tweets were collected from October to December 2015. We used the public streaming Twitter API to download the tweets.

We then manually filtered the resulting tweets to obtain a set of 200 meaningful topics with at least 100 tweets each (after filtering out near-duplicates). We excluded topics that were incomprehensible, ambiguous (e.g., Barcelona, which is the name both of a city and of a sports team), or too general (e.g., Paris, which is the name of a big city). We then discarded tweets that were just mentioning the topic but were not really about the topic.

Note that the topics in the training and in the test sets do not overlap, i.e., the test set consists of tweets about topics different from the topics the training and development tweets are about.

Table 2: Statistics about data from the 2013-2015 editions of the SemEval task on Sentiment Analysis in Twitter, which could be used for training and development for SemEval-2016 Task 4.

| Dataset            | POSITIVE | NEGATIVE | NEUTRAL | Total |
|--------------------|----------|----------|---------|-------|
| Twitter2013-train  | 3,662    | 1,466    | 4,600   | 9,728 |
| Twitter2013-dev    | 575      | 340      | 739     | 1,654 |
| Twitter2013-test   | 1,572    | 601      | 1,640   | 3,813 |
| SMS2013-test       | 492      | 394      | 1,207   | 2,093 |
| Twitter2014-test   | 982      | 202      | 669     | 1,853 |
| Twitter2014-sarcasm| 33       | 40       | 13      | 86    |
| LiveJournal2014-test| 427     | 304      | 411     | 1,142 |
| Twitter2015-test   | 1,040    | 365      | 987     | 2,392 |

For Subtask A, we did not allow training on the testing datasets from 2013–2015, as we used them for progress testing.

Filtering based on an existing lexicon does bias the dataset to some degree; however, the text still contains sentiment expressions outside those in the lexicon.

We distributed the datasets to the task participants in a similar way: we only released the annotations and the tweet IDs, and the participants had to download the actual tweets by themselves via the API, for which we provided a script: https://github.com/aritter/twitter_download
**Instructions:** Given a Twitter message and a topic, identify whether the message is highly positive, positive, neutral, negative, or highly negative (a) in general and (b) with respect to the provided topic. If a tweet is sarcastic, please select the checkbox “The tweet is sarcastic”. Please read the examples and the invalid responses before beginning if this is the first time you are working on this HIT.

---

**Sentence:** We’re excited to announce that AC/DC tribute band Big Jack will take the main stage at Fall Jam on Sunday September 27th!

Overall, the tweet is: ○ Highly Positive ○ Positive ○ Neutral ○ Negative ○ Highly Negative
The sentiment towards the topic AC/DC is: ○ Highly Positive ○ Positive ○ Neutral ○ Negative ○ Highly Negative
○ The tweet is sarcastic.

---

**Figure 1:** The instructions provided to the Mechanical Turk annotators, followed by a screenshot.

**Annotation with Amazon’s Mechanical Turk.** A Human Intelligence Task (HIT) consisted of providing all required annotations for a given tweet message. In order to qualify to work on our HITs, a Mechanical Turk annotator (a.k.a. “Turker”) had to have an approval rate greater than 95% and to have completed at least 50 approved HITs. Each HIT was carried out by five Turkers and consisted of five tweets to be annotated. A Turker had to indicate the overall polarity of the tweet message (on a five-point scale) as well as the overall polarity of the message towards the given target topic (again, on a five-point scale). The annotation instructions along with an example are shown in Figure 1. We made available to the Turkers several additional examples, which are shown in Table 3.

We rejected HITs with the following problems:

- one or more responses do not have the overall sentiment marked;
- one or more responses do not have the sentiment towards the topic marked;
- one or more responses appear to be randomly selected.

**Annotation with CrowdFlower.** We annotated the TEST data using CrowdFlower, as it allows better quality control of the annotations across a number of dimensions. Most importantly, it allows us to find and exclude unreliable annotators based on hidden tests, which we created starting with the highest-confidence and highest-agreement annotations from Mechanical Turk. We added some more tests manually. Otherwise, we setup the annotation task giving exactly the same instructions and examples as in Mechanical Turk.

**Consolidation of annotations.** In previous years, we used majority voting to select the true label (and discarded cases where a majority had not emerged, which amounted to about 50% of the tweets). As this year we have a five-point scale, where the expected agreement is lower, we used a two-step procedure. If three out of the five annotators agreed on a label, we accepted the label. Otherwise, we first mapped the categorical labels to the integer values $-2, -1, 0, 1, 2$. Then we calculated the average, and finally we mapped that average to the closest integer value. In order to counter-balance the tendency of the average to stay away from $-2$ and $2$, and also to prefer $0$, we did not use rounding at $±0.5$ and $±1.5$, but at $±0.4$ and $±1.4$ instead.

To give the reader an idea about the degree of agreement, we will look at the TEST dataset as an example. It included 20,632 tweets. For 2,760, all five annotators assigned the same value, and for another 9,944 there was a majority value. For the remaining 7,928 cases, we had to perform averaging as described above.

The consolidated statistics from the five annotators on a three-point scale for Subtask A are shown in Table 4. Note that, for consistency, we annotated the data for Subtask A on a five-point scale, which we then converted to a three-point scale.

The topic annotations on a two-point scale for Subtasks B and D are shown in Table 5 while those on a five-point scale for Subtasks C and E are in Table 6. Note that, as for Subtask A, the two-point scale annotation counts for Subtasks B and D derive from summing the POSITIVES with the HIGHLYPOSITIVES, and the NEGATIVES with the HIGHLYNEGATIVES from Table 6, moreover, this time we also remove the NEUTRALS.
Why would you still wear shorts when it’s this cold?! I love how Britain see’s a bit of sun and they’re like ‘OOOH LET’S STRIP!'

Saturday without Leeds United is like Sunday dinner it doesn’t feel normal at all (Ryan)

Who are you tomorrow? Will you make me smile or just bring me sorrow? #HottieOfTheWeek Demi Lovato

Overall Sentiment | Topic Sentiment
--- | ---
POSITIVE | Britain: NEGATIVE
NEGATIVE | Leeds United: HIGHLY POSITIVE
NEUTRAL | Demi Lovato: POSITIVE

Table 3: List of example tweets and annotations that were provided to the annotators.

| Tweet | Overall Sentiment | Topic Sentiment |
|---|---|---|
| Why would you still wear shorts when it’s this cold?! I love how Britain see’s a bit of sun and they’re like ‘OOOH LET’S STRIP!' | POSITIVE | Britain: NEGATIVE |
| Saturday without Leeds United is like Sunday dinner it doesn’t feel normal at all (Ryan) | NEGATIVE | Leeds United: HIGHLY POSITIVE |
| Who are you tomorrow? Will you make me smile or just bring me sorrow? #HottieOfTheWeek Demi Lovato | NEUTRAL | Demi Lovato: POSITIVE |

Table 4: 2016 data statistics (Subtask A).

| | POSITIVE | NEUTRAL | NEGATIVE | Total |
|---|---|---|---|---|
| TRAIN | 3,094 | 863 | 2,043 | 6,000 |
| DEV | 844 | 765 | 391 | 2,000 |
| DEVTEST | 994 | 681 | 325 | 2,000 |
| TEST | 7,059 | 10,342 | 3,231 | 20,632 |

Table 5: 2016 data statistics (Subtasks B and D).

| | POSITIVE | NEUTRAL | NEGATIVE | Total |
|---|---|---|---|---|
| TRAIN | 60 | 3,591 | 755 | 4,346 |
| DEV | 20 | 986 | 339 | 1,325 |
| DEVTEST | 20 | 1,153 | 264 | 1,417 |
| TEST | 100 | 8,212 | 2,339 | 10,551 |

Table 6: 2016 data statistics (Subtasks C and E).

| | HIGHLY POSITIVE | POSITIVE | NEUTRAL | NEGATIVE | HIGHLY NEGATIVE | Total |
|---|---|---|---|---|---|---|
| TRAIN | 60 | 437 | 3,154 | 1,654 | 668 | 87 | 6,000 |
| DEV | 20 | 53 | 933 | 675 | 296 | 43 | 2,000 |
| DEVTEST | 20 | 148 | 1,005 | 583 | 233 | 31 | 2,000 |
| TEST | 100 | 382 | 7,830 | 10,081 | 2,201 | 138 | 20,632 |

As we use the same test tweets for all subtasks, the submission of results by participating teams was subdivided in two stages: (i) participants had to submit results for Subtasks A, C, E, and (ii) only after the submission deadline for A, C, E had passed, we distributed to participants the unlabelled test data for Subtasks B and D.

Otherwise, since for Subtasks B and D we filter out the NEUTRALS, we would have leaked information about which the NEUTRALS are, and this information could have been used in Subtasks C and E.

Finally, as the same tweets can be selected for different topics, we ended up with some duplicates; arguably, these are true duplicates for Subtask A only, as for the other subtasks the topics still differ. This includes 25 duplicates in TRAIN, 3 in DEV, 2 in DEVTEST, and 116 in TEST. There is a larger number in TEST, as TEST is about twice as large as TRAIN, DEV, and DEVTEST combined. This is because we wanted a large TEST set with 100 topics and 200 tweets per topic on average for Subtasks C and E.

4 Evaluation Measures

This section discuss the evaluation measures for the five subtasks of our SemEval-2016 Task 4. A document describing the evaluation measures in detail\(^6\) (Nakov et al., 2016a), and a scoring software implementing all the five “official” measures, were made available to the participants via the task website together with the training data\(^7\).

For Subtasks B to E, the datasets are each subdivided into a number of “topics”, and the subtask needs to be carried out independently for each topic. As a result, each of the evaluation measures will be “macroaveraged” across the topics, i.e., we compute the measure individually for each topic, and we then average the results across the topics.

\(^6\)http://alt.qcri.org/semeval2016/task4/
\(^7\)An earlier version of the scoring script contained a bug, to the effect that for Subtask B it was computing $F_1^{PN}$ and not $\rho^{PN}$. This was detected only after the submissions were closed, which means that participants to Subtask B who used the scoring system (and not their own implementation of $\rho^{PN}$) for parameter optimization, may have been penalized in the ranking as a result.
4.1 Subtask A: Message polarity classification

Subtask A is a single-label multi-class (SLMC) classification task. Each tweet must be classified as belonging to exactly one of the following three classes $C=\{\text{POSITIVE, NEUTRAL, NEGATIVE}\}$.

We adopt the same evaluation measure as the 2013-2015 editions of this subtask, $F_1^{PN}$:

$$F_1^{PN} = \frac{\frac{1}{2}(F_P + F_N)}{2}$$  \hspace{1cm} (1)

$F_1^P$ is the $F_1$ score for the POSITIVE class:

$$F_1^P = \frac{2\pi_P \rho_P}{\pi_P + \rho_P}$$  \hspace{1cm} (2)

Here, $\pi_P$ and $\rho_P$ denote precision and recall for the POSITIVE class, respectively:

$$\pi_P = \frac{PP}{PP + PU + PN}$$  \hspace{1cm} (3)

$$\rho_P = \frac{PP}{PP + UP + NP}$$  \hspace{1cm} (4)

where $PP, UP, NP, PU, PN$ are the cells of the confusion matrix shown in Table 7.

|               | Gold Standard |          |
|---------------|---------------|----------|
| **Predicted** | POSITIVE      | NEUTRAL  | NEGATIVE |
| POSITIVE      | PP            | PU       | PN       |
| NEUTRAL       | UP            | UU       | UN       |
| NEGATIVE      | NP            | NU       | NN       |

Table 7: The confusion matrix for Subtask A. Cell $XY$ stands for “the number of tweets that the classifier labeled $X$ and the gold standard labels as $Y$”. $P, U, N$ stand for POSITIVE, NEUTRAL, NEGATIVE, respectively.

$F_1^N$ is defined analogously, and the measure we finally adopt is $F_1^{PN}$ as from Equation 1.

4.2 Subtask B: Tweet classification according to a two-point scale

Subtask B is a binary classification task. Each tweet must be classified as either POSITIVE or NEGATIVE.

For this subtask we adopt macroaveraged recall:

$$\rho^{PN} = \frac{1}{2}(\rho^P + \rho^N)$$  \hspace{1cm} (5)

$$= \frac{1}{2}(PP + \frac{NN}{NP + PN})$$

In the above formula, $\rho^P$ and $\rho^N$ are the positive and the negative class recall, respectively. Note that $U$ terms are entirely missing in Equation 5; this is because we do not have the NEUTRAL class for SemEval-2016 Task 4, subtask A.

$\rho^{PN}$ ranges in $[0, 1]$, where a value of 1 is achieved only by the perfect classifier (i.e., the classifier that correctly classifies all items), a value of 0 is achieved only by the perverse classifier (the classifier that misclassifies all items), while 0.5 is both $(i)$ the value obtained by a trivial classifier (i.e., the classifier that assigns all tweets to the same class – be it POSITIVE or NEGATIVE), and $(ii)$ the expected value of a random classifier. The advantage of $\rho^{PN}$ over “standard” accuracy is that it is more robust to class imbalance. The accuracy of the majority-class classifier is the relative frequency (aka “prevalence”) of the majority class, that may be much higher than 0.5 if the test set is imbalanced. Standard $F_1$ is also sensitive to class imbalance for the same reason. Another advantage of $\rho^{PN}$ over $F_1$ is that $\rho^{PN}$ is invariant with respect to switching POSITIVE with NEGATIVE, while $F_1$ is not. See (Sebastiani, 2015) for more details on $\rho^{PN}$.

As we noted before, the training dataset, the development dataset, and the test dataset are each subdivided into a number of topics, and Subtask B needs to be carried out independently for each topic. As a result, the evaluation measures discussed in this section are computed individually for each topic, and the results are then averaged across topics to yield the final score.

4.3 Subtask C: Tweet classification according to a five-point scale

Subtask C is an ordinal classification (OC – also known as ordinal regression) task, in which each tweet must be classified into exactly one of the classes in $C=\{\text{HIGHLYPOSITIVE, POSITIVE, NEUTRAL, NEGATIVE, HIGHLYNEGATIVE}\}$, represented in our dataset by numbers in $\{+2,+1,0,-1,-2\}$, with a total order defined on $C$. The essential difference between SLMC (see Section 4.1 above) and OC is that not all mistakes weigh equally in the latter. For example, misclassifying a HIGHLYNEGATIVE example as HIGHLYPOSITIVE is a bigger mistake than misclassifying it as NEGATIVE or NEUTRAL.
As our evaluation measure, we use macroaveraged mean absolute error ($MAE^M$):

$$MAE^M(h, Te) = \frac{1}{|C|} \sum_{j=1}^{|C|} \frac{1}{|Te_j|} \sum_{x_i \in Te_j} |h(x_i) - y_i|$$

where $y_i$ denotes the true label of item $x_i$, $h(x_i)$ is its predicted label, $Te_j$ denotes the set of test documents whose true class is $c_j$, $|h(x_i) - y_i|$ denotes the “distance” between classes $h(x_i)$ and $y_i$ (e.g., the distance between HIGHLY POSITIVE and NEGATIVE is 3), and the “M” superscript indicates “macroaveraging”.

The advantage of $MAE^M$ over “standard” mean absolute error, which is defined as:

$$MAE^s(h, Te) = \frac{1}{|Te|} \sum_{x_i \in Te} |h(x_i) - y_i|$$

is that it is robust to class imbalance (which is useful, given the imbalanced nature of our dataset). On perfectly balanced datasets $MAE^M$ and $MAE^s$ are equivalent.

Unlike the measures discussed in Sections 4.1 and 4.2, $MAE^M$ is a measure of error, and not accuracy, and thus lower values are better. See (Baccianella et al., 2009) for more detail on $MAE^M$.

Similarly to Subtask B, Subtask C needs to be carried out independently for each topic. As a result, $MAE^M$ is computed individually for each topic, and the results are then averaged across all topics to yield the final score.

4.4 Subtask D: Tweet quantification according to a two-point scale

Subtask D also assumes a binary quantification setup, in which each tweet is classified as POSITIVE or NEGATIVE. The task is to compute an estimate $\hat{p}(c_j)$ of the relative frequency (in the test set) of each of the classes.

The difference between binary classification (as from Section 4.2) and binary quantification is that errors of different polarity (e.g., a false positive and a false negative for the same class) can compensate each other in the latter. Quantification is thus a more lenient task since a perfect classifier is also a perfect quantifier, but a perfect quantifier is not necessarily a perfect classifier.

We adopt normalized cross-entropy, better known as Kullback-Leibler Divergence (KLD). KLD was proposed as a quantification measure in (Forman, 2005), and is defined as follows:

$$KLD(\hat{p}, p, C) = \sum_{c_j \in C} p(c_j) \log_e \frac{p(c_j)}{\hat{p}(c_j)}$$

$KLD$ is a measure of the error made in estimating a true distribution $p$ over a set $C$ of classes by means of a predicted distribution $\hat{p}$. Like $MAE^M$ in Section 4.3, $KLD$ is a measure of error, which means that lower values are better. $KLD$ ranges between 0 (best) and $+\infty$ (worst).

Note that the upper bound of $KLD$ is not finite since Equation [7] has predicted prevalences, and not true prevalences, at the denominator: that is, by making a predicted prevalence $\hat{p}(c_j)$ infinitely small we can make $KLD$ infinitely large. To solve this problem, in computing $KLD$ we smooth both $p(c_j)$ and $\hat{p}(c_j)$ via additive smoothing, i.e.,

$$p^s(c_j) = \frac{p(c_j) + \epsilon}{(\sum_{c_j \in C} p(c_j)) + \epsilon \cdot |C|} = \frac{p(c_j) + \epsilon}{1 + \epsilon \cdot |C|}$$

where $p^s(c_j)$ denotes the smoothed version of $p(c_j)$ and the denominator is just a normalizer (same for the $\hat{p}^s(c_j)$’s); the quantity $\epsilon = \frac{1}{2 |Te|}$ is used as a smoothing factor, where $Te$ denotes the test set.

The smoothed versions of $p(c_j)$ and $\hat{p}(c_j)$ are used in place of their original versions in Equation [7]; as a result, $KLD$ is always defined and still returns a value of 0 when $p$ and $\hat{p}$ coincide.

$KLD$ is computed individually for each topic, and the results are averaged to yield the final score.

4.5 Subtask E: Tweet quantification according to a five-point scale

Subtask E is an ordinal quantification (OQ) task, in which (as in OC) each tweet belongs exactly to one of the classes in $C=$\{HIGHLY POSITIVE, POSITIVE, NEUTRAL, NEGATIVE, HIGHLY NEGATIVE\}, where there is a total order on $C$. As in binary quantification, the task is to compute an estimate $\hat{p}(c_j)$ of the relative frequency $p(c_j)$ in the test tweets of all the classes $c_j \in C$. 

The measure we adopt for OQ is the Earth Mover’s Distance (Rubner et al., 2000) (also known as the Vaseršteın metric (Rüschendorf, 2001)), a measure well-known in the field of computer vision. EMD is currently the only known measure for ordinal quantification. It is defined for the general case in which a distance \(d(c', c'')\) is defined for each \(c', c'' \in \mathcal{C}\). When there is a total order on the classes in \(\mathcal{C}\) and \(d(c_i, c_{i+1}) = 1\) for all \(i \in \{1, ..., |\mathcal{C} - 1\}\) (as in our application), the Earth Mover’s Distance is defined as

\[
EMD(\hat{p}, p) = \sum_{j=1}^{|\mathcal{C}|-1} \sum_{i=1}^{j} |\hat{p}(c_i) - \sum_{i=1}^{j} p(c_i)| \tag{9}
\]

and can be computed in \(|\mathcal{C}|\) steps from the estimated and true class prevalences.

Like \(KLD\) in Section 4.4, \(EMD\) is a measure of error, so lower values are better; \(EMD\) ranges between 0 (best) and \(|\mathcal{C}| - 1\) (worst). See (Esuli and Sebastiāni, 2010) for more details on \(EMD\).

As before, \(EMD\) is computed individually for each topic, and the results are then averaged across all topics to yield the final score.

5 Participants and Results

A total of 43 teams (see Table 15 at the end of the paper) participated in SemEval-2016 Task 4, representing 25 countries; the country with the highest participation was China (5 teams), followed by Italy, Spain, and USA (4 teams each). The subtask with the highest participation was Subtask A (34 teams), followed by Subtask B (19 teams), Subtask D (14 teams), Subtask C (11 teams), and Subtask E (10 teams).

It was not surprising that Subtask A proved to be the most popular – it was a rerun from previous years; conversely, none among Subtasks B to E had previously been offered in precisely the same form. Quantification-related subtasks (D and E) generated 24 participations altogether, while subtasks with an ordinal nature (C and E) attracted 21 participations. Only three teams participated in all five subtasks; conversely, no less than 23 teams took part in one subtask only (with a few exceptions, Subtask A). Many teams that participated in more than one subtask used essentially the same system for all of them, with little tuning to the specifics of each subtask.

Few trends stand out among the participating systems. In terms of the supervised learning methods used, there is a clear dominance of methods based on deep learning, including convolutional neural networks and recurrent neural networks (and, in particular, long short-term memory networks); the software libraries for deep learning most frequently used by the participants are Theano and Keras. Conversely, kernel machines seem to be less frequently used than in the past, and the use of learning methods other than the ones mentioned above is scarce.

The use of distant supervision is ubiquitous; this is natural, since there is an abundance of freely available tweets labelled according to sentiment (possibly with silver labels only, e.g., emoticons), and it is intuitive that their use as additional training data could be helpful. Another ubiquitous technique is the use of word embeddings, usually generated via either word2vec (Mikolov et al., 2013) or GloVe (Pennington et al., 2014); most authors seem to use general-purpose, pre-trained embeddings, while some authors also use customized word embeddings, trained either on the Tweet 2016 dataset or on tweet datasets of some sort.

Nothing radically new seems to have emerged with respect to text preprocessing; as in previous editions of this task, participants use a mix of by now obvious techniques, such as negation scope detection, elongation normalization, detection of amplifiers and diminishers, plus the usual extraction of word n-grams, character n-grams, and POS n-grams. The use of sentiment lexicons (alone or in combination with each other; general-purpose or Twitter-specific) is obviously still frequent.

In the next five subsections, we discuss the results of the participating systems in the five subtasks, focusing on the techniques and tools that the top-ranked participants have used. We also focus on how the participants tailored (if at all) their approach to the specific subtask. When discussing a specific subtask, we will adopt the convention of adding to a team name a subscript which indicates the position in the ranking for that subtask that the team obtained; e.g., when discussing Subtask E, “Fink12” indicates team “Fink, which placed 2nd in the ranking for Subtask E”. The papers describing the participants’ approach are quoted in Table 15.
5.1 Subtask A: Message polarity classification

Table 8 ranks the systems submitted by the 34 teams who participated in Subtask A “Message Polarity Classification” in terms of the official measure $F_{1}^{PN}$. We further show the result for two other measures, $\rho_{PN}$ (the measure that we adopted for Subtask B) and accuracy ($Acc = \frac{TP+TN}{TP+TN+FP+FN}$). We also report the result for a baseline classifier that assigns to each tweet the POSITIVE class. For Subtask A evaluated using $F_{1}^{PN}$, this is the equivalent of the majority class classifier for (binary or SLMC) classification evaluated via vanilla accuracy, i.e., this is the “smartest” among the trivial policies that attempt to maximize $F_{1}^{PN}$.

All 34 participating systems were able to outperform the baseline on all three measures, with the exception of one system that scored below the baseline on Acc. The top-scoring team (SwissCheese1) used an ensemble of convolutional neural networks, differing in their choice of filter shapes, pooling shapes and usage of hidden layers. Word embeddings generated via word2vec were also used, and the neural networks were trained by using distant supervision. Out of the 10 top-ranked teams, 5 teams (SwissCheese1, SENSEI-LIF2, UNIMELB3, INESC-ID4, INSIGHT-1) and 7 teams (SwissCheese1, SENSEI-LIF2, UNIMELB3, INESC-ID4, aueb.twitter.sentiment5, i2RNTU7, INSIGHT-18) used either general-purpose or task-specific word embeddings, generated via word2vec or GloVe.

### Historical results.

We also tested the participating systems on the test sets from the three previous editions of this subtask. Participants were not allowed to use these test sets for training. Results (measured on $F_{1}^{PN}$) are reported in Table 9. The top-performing systems on Tweet 2016 are also top-ranked on the test datasets from previous years. There is a general pattern: the top-ranked system in year $x$ outperforms the top-ranked system in year $(x−1)$ on the official dataset of year $(x−1)$. Top-ranked systems tend to use approaches that are universally strong, even when tested on out-of-domain test sets such as SMS, LiveJournal, or sarcastic tweets (yet, for sarcastic tweets, there are larger differences in rank compared to systems rankings on Tweet 2016). It is unclear where improvements come from: (a) the additional training data that we made available this year (in addition to TweetTrain-2013, which was used in 2013–2015), thus effectively doubling the amount of training data, or (b) because of advancement of learning methods.

We further look at the top scores achieved by any system in the period 2013–2016. The results are shown in Table 10. Interestingly, the results for a test set improve in the second year it is used (i.e., the year after it was used as an official test set) by 1–3 points absolute, but then do not improve further and stay stable, or can even decrease a bit. This might be due to participants optimizing their systems primarily on the test set from the preceding year.

| # | System                  | $F_{1}^{PN}$ | $\rho_{PN}$ | Acc  |
|---|-------------------------|-------------|-------------|------|
| 1 | SwissCheese             | 0.633       | 0.667       | 0.646 |
| 2 | SENSEI-LIF              | 0.630       | 0.670       | 0.617 |
| 3 | UNIMELB                 | 0.617       | 0.641       | 0.616 |
| 4 | INESC-ID                | 0.610       | 0.662       | 0.600 |
| 5 | aueb.twitter.sentiment  | 0.605       | 0.644       | 0.629 |
| 6 | SentiSys                | 0.598       | 0.641       | 0.609 |
| 7 | i2RNTU                  | 0.597       | 0.637       | 0.593 |
| 8 | INSIGHT-1               | 0.593       | 0.610       | 0.635 |
| 9 | TwiSE                   | 0.589       | 0.598       | 0.528 |
| 10 | ECNU (*)                | 0.585       | 0.617       | 0.571 |
| 11 | NTNUSentEval           | 0.583       | 0.619       | 0.643 |
| 12 | MDSENT                  | 0.580       | 0.592       | 0.545 |
| 13 | CUFE                    | 0.580       | 0.619       | 0.637 |
| 14 | THUIR                   | 0.576       | 0.605       | 0.596 |
| 15 | PUT                     | 0.576       | 0.607       | 0.584 |
| 16 | LYS                     | 0.575       | 0.615       | 0.585 |
| 17 | IIP                     | 0.574       | 0.579       | 0.537 |
| 18 | UniPI                   | 0.571       | 0.607       | 0.639 |
| 19 | DIELGOLab16 (*)         | 0.554       | 0.593       | 0.549 |
| 20 | GIT                     | 0.559       | 0.557       | 0.518 |
| 21 | INESC-ID                | 0.505       | 0.560       | 0.541 |
| 22 | DSIC-ELIRF              | 0.502       | 0.511       | 0.513 |
| 23 | UofL                    | 0.499       | 0.537       | 0.572 |
| 24 | ELIRF                   | 0.499       | 0.516       | 0.543 |
| 25 | ISTI-CNR                | 0.494       | 0.529       | 0.567 |
| 26 | SteM                    | 0.478       | 0.496       | 0.452 |
| 27 | Tweeter                 | 0.455       | 0.503       | 0.523 |
| 28 | Minions                 | 0.415       | 0.485       | 0.556 |
| 29 | Aicyber                 | 0.402       | 0.457       | 0.506 |
| 30 | mib                     | 0.401       | 0.438       | 0.480 |
| 31 | VCU-TSA                 | 0.372       | 0.390       | 0.382 |
| 32 | SentimentalTists        | 0.339       | 0.424       | 0.480 |
| 33 | WR                      | 0.330       | 0.333       | 0.298 |
| 34 | CICBUAPlp               | 0.303       | 0.377       | 0.374 |
|   | Baseline                | **0.255**   | **0.333**   | **0.342** |

Table 8: Results for Subtask A “Message Polarity Classification” on the Tweet 2016 dataset. The systems are ordered by their $F_{1}^{PN}$ score. In each column the rankings according to the corresponding measure are indicated with a subscript. Teams marked as “(*)” are late submitters, i.e., their original submission was deemed irregular by the organizers, and a revised submission was entered after the deadline.
Table 9: Historical results for Subtask A “Message Polarity Classification”. The systems are ordered by their score on the Tweet 2016 dataset; the rankings on the individual datasets are indicated with a subscript. The meaning of “(*)” is as in Table 8.

### 5.2 Subtask B: Tweet classification according to a two-point scale

Table 10 ranks the 19 teams who participated in Subtask B “Tweet classification according to a two-point scale” in terms of the official measure $\rho^{PN}$. Two other measures are reported, $F_i^{PN}$ (the measure adopted for Subtask A) and accuracy ($\text{Acc}$). We also report the result of a baseline that assigns to each tweet the positive class. This is the “smartest” among the trivial policies that attempt to maximize $\rho^{PN}$. This baseline always returns $\rho^{PN} = 0.500$.

Note however that this is also (i) the value returned by the classifier that assigns to each tweet the negative class, and (ii) the expected value returned by the random classifier; for more details see (Sc Bastiani, 2015 Section 5), where $\rho^{PN}$ is called $K$.

The top-scoring team (Tweester) used a combination of convolutional neural networks, topic modeling, and word embeddings generated via word2vec. Similar to Subtask A, the main trend among all participants is the widespread use of deep learning techniques.
### Table 10: Historical results for the best systems for Subtask A “Message Polarity Classification” over the years 2013–2016.

| Year      | Tweet | SMS | 2013 | Tweet | Sarcasm | Live-Journal | 2015 | Tweet | 2016 | Tweet |
|-----------|-------|-----|------|-------|---------|--------------|------|-------|------|-------|
| Best in 2016 | 0.723 | 0.641 | 0.744 | 0.566 | 0.741 |                        | 0.671 |       | 0.633 |
| Best in 2015 | 0.728 | 0.685 | 0.744 | 0.591 | 0.753 |                        | 0.648 |       |       |
| Best in 2014 | 0.721 | 0.703 | 0.710 | 0.582 | 0.748 |                        |       |       |       |
| Best in 2013 | 0.690 | 0.685 |       |       |       |                        |       |       |       |

### Table 11: Results for Subtask B “Tweet classification according to a two-point scale” on the Tweet 2016 dataset. The systems are ordered by their $\rho_{PN}$ score (higher is better). The meaning of “(*)” is as in Table 8.

| # | System       | $\rho_{PN}$ | $F_{1}^{PN}$ | Acc  |
|---|--------------|--------------|--------------|------|
| 1 | Tweester     | 0.797        | 0.799        | 0.862 |
| 2 | LYS          | 0.791        | 0.720        | 0.862 |
| 3 | thecerealkiller | 0.784      | 0.762        | 0.823 |
| 4 | ECNU (*)     | 0.768        | 0.770        | 0.843 |
| 5 | INSIGHT-1    | 0.767        | 0.786        | 0.864 |
| 6 | PUT          | 0.763        | 0.732        | 0.794 |
| 7 | UNIMELB      | 0.758        | 0.789        | 0.870 |
| 8 | TwiSE        | 0.756        | 0.752        | 0.826 |
| 9 | GTI          | 0.736        | 0.731        | 0.811 |
| 10 | Finki       | 0.720        | 0.748        | 0.848 |
| 11 | pkudblab    | 0.689        | 0.716        | 0.832 |
| 12 | CUFEE       | 0.679        | 0.708        | 0.834 |
| 13 | ISTI-CNR     | 0.671        | 0.690        | 0.811 |
| 14 | SwissCheese  | 0.648        | 0.674        | 0.820 |
| 15 | SentimentalITists | 0.624   | 0.643        | 0.802 |
| 16 | PotTS        | 0.618        | 0.610        | 0.712 |
| 17 | OPAL         | 0.616        | 0.633        | 0.792 |
| 18 | WR           | 0.522        | 0.502        | 0.577 |
| 19 | VCU-TSA      | 0.502        | 0.448        | 0.775 |
|    | Baseline     | 0.500        | 0.438        | 0.778 |

### 5.3 Subtask C: Tweet classification according to a five-point scale

Table 12 ranks the 11 teams who participated in Subtask C “Tweet classification according to a five-point scale” in terms of the official measure $MAE_M$; we also show $MAE^\mu$ (see Equation 6). We also report the result of a baseline system that assigns to each tweet the middle class (i.e., NEUTRAL); for ordinal classification evaluated via $MAE_M$, this is the majority-class classifier for (binary or SLMC) classification evaluated via vanilla accuracy, i.e., this is (Baccianella et al., 2009) the “smartest” among the trivial policies that attempt to maximize $MAE_M$.

| # | System       | $MAE_M$ | $MAE^\mu$ |
|---|--------------|---------|-----------|
| 1 | TwiSE        | 0.719   | 0.632     |
| 2 | ECNU (*)     | 0.806   | 0.726     |
| 3 | LYS          | 0.860   | 0.773     |
| 4 | PUT          | 0.864   | 0.694     |
| 5 | Finki        | 0.869   | 0.672     |
| 6 | INSIGHT-1    | 1.006   | 0.607     |
| 7 | ISTI-CNR     | 1.074   | 0.580     |
| 8 | YZU-NLP      | 1.111   | 0.588     |
| 9 | SentimentalITists | 1.148   | 0.625     |
| 10 | PotTS        | 1.237   | 0.860     |
| 11 | pkudblab     | 1.697   | 1.300     |
|    | Baseline     | 1.200   | 0.537     |

Out of the 10 top-ranked participating teams, 5 teams (Tweester1, LYS2, INSIGHT-15, UNIMELB7, Finki10) used convolutional neural networks; 3 teams (thecerealkiller3, UNIMELB7, Finki10) submitted systems using recurrent neural networks; and 7 teams (Tweester1, LYS2, INSIGHT-15, UNIMELB7, Finki10) incorporated in their participating systems either general-purpose or task-specific word embeddings (generated via toolkits such as GloVe or word2vec).

Conversely, the use of classifiers such as support vector machines, which were dominant until a few years ago, seems to have decreased, with only one team (TwiSE8) in the top 10 using them.

The top-scoring team (TwiSE1) used a single-label multi-class classifier to classify the tweets according to their overall polarity. In particular, they used logistic regression that minimizes the multinomial loss across the classes, with weights to cope with class imbalance. Note that they ignored the given topics altogether.
### 5.4 Subtask D: Tweet quantification according to a two-point scale

Table 13 ranks the 14 teams who participated in Subtask D “Tweet quantification according to a two-point scale” on the official measure $KLD$. Two other measures are reported, absolute error ($AE$):

$$AE(p, \hat{p}, C) = \frac{1}{|C|} \sum_{c \in C} |\hat{p}(c) - p(c)| \quad (10)$$

and relative absolute error ($RAE$):

$$RAE(p, \hat{p}, C) = \frac{1}{|C|} \sum_{c \in C} \frac{|\hat{p}(c) - p(c)|}{p(c)} \quad (11)$$

where the notation is the same as in Equation 7.

Only 2 of the 11 participating teams tuned their systems to exploit the ordinal (as opposed to binary, or single-label multi-class) nature of this subtask. The two teams who did exploit the ordinal nature of the problem are PUT, which uses an ensemble of ordinal regression approaches, and ISTI-CNR, which uses a tree-based approach to ordinal regression. All other teams used general-purpose approaches for single-label multi-class classification, in many cases relying (as for Subtask B) on convolutional neural networks, recurrent neural networks, and word embeddings.

We also report the result of a “maximum likelihood” baseline system (dubbed Baseline1). This system assigns to each test topic the distribution of the training tweets (the union of TRAIN, DEV, DEVTEST) across the classes. This is the “smartest” among the trivial policies that attempt to maximize $KLD$. We also report the result of a further (less smart) baseline system (dubbed Baseline2), i.e., one that assigns a prevalence of 1 to the majority class (which happens to be the POSITIVE class) and a prevalence of 0 to the other class.

The top-scoring team (Fink1i) adopts an approach based on “classify and count”, a classification-oriented (instead of quantification-oriented) approach, using recurrent and convolutional neural networks, and GloVe word embeddings.

Indeed, only 5 of the 14 participating teams tuned their systems to the fact that it deals with quantification (as opposed to classification). Among the teams who do rely on quantification-oriented approaches, teams LYS and HSENN used an existing structured prediction method that directly optimizes $KLD$; teams QCRI and ISTI-CNR use existing probabilistic quantification methods; team NRU-HSE uses an existing iterative quantification method based on cost-sensitive learning. Interestingly, team TwiSE uses a “classify and count” approach after comparing it with a quantification-oriented method (similar to the one used by teams LYS and HSENN) on the development set, and concluding that the former works better than the latter. All other teams used “classify and count” approaches, mostly based on convolutional neural networks and word embeddings.

### 5.5 Subtask E: Tweet quantification according to a five-point scale

Table 14 lists the results obtained by the 10 participating teams on Subtask E “Tweet quantification according to a five-point scale”. We also report the result of a “maximum likelihood” baseline system (dubbed Baseline1), i.e., one that assigns to each test topic the same distribution, namely the distribution of the training tweets (the union of TRAIN, DEV, DEVTEST) across the classes; this is the “smartest” among the trivial policies (i.e., those that do not require any genuine work) that attempt to maximize $EMD$.  

| # | System   | $KLD$   | $AE$   | $RAE$   |
|---|----------|---------|--------|---------|
| 1 | Fink1    | 0.034,  | 0.074, | 0.707,  |
| 2 | LYS      | 0.053,  | 0.099, | 0.844,  |
| 4 | TwiSE    | 0.053,  | 0.101, | 0.864,  |
| 5 | INSIGHT-I| 0.054,  | 0.085, | 0.423,  |
| 7 | NRU-HSE  | 0.084,  | 0.120, | 0.767,  |
| 8 | PotTS    | 0.094,  | 0.150, | 1.838,  |
| 9 | pkudlab  | 0.099,  | 0.109, | 0.947,  |
| 10 | ECNU (*) | 0.121,  | 0.148, | 1.171,  |
| 11 | ISTI-CNR | 0.127,  | 0.147, | 1.371,  |
| 12 | SwissCheese | 0.191, | 0.147, | 0.638,  |
| 13 | UDLAP    | 0.261,  | 0.274, | 2.973,  |
| 14 | HSENN    | 0.399,  | 0.336, | 3.930,  |

Table 13: Results for Subtask D “Tweet quantification according to a two-point scale” on the Tweet 2016 dataset. The systems are ordered by their $KLD$ score (lower is better). The meaning of “(*)” is as in Table 8.
We further report the result of less smart baseline system (dubbed Baseline2) – one that assigns a prevalence of 1 to the majority class (which coincides with the POSITIVE class) and a prevalence of 0 to all other classes.

| #  | System     | EMD  |
|----|------------|------|
| 1  | QCRI       | 0.243|
| 2  | Finki      | 0.316|
| 3  | pkudlab    | 0.331|
| 4  | NRU-HSE    | 0.334|
| 5  | ECNU (*)   | 0.341|
| 6  | ISTI-CNR   | 0.358|
| 7  | LYS        | 0.360|
| 8  | INSIGHT-1  | 0.366|
| 9  | HSENN      | 0.545|
| 10 | PotTS      | 0.818|
|    | Baseline1  | 0.474|
|    | Baseline2  | 0.734|

Table 14: Results for Subtask E “Tweet quantification according to a five-point scale” on the Tweet 2016 dataset. The systems are ordered by their EMD score (lower is better). The meaning of “(*)” is as in Table 8.

Only 3 of the 10 participants tuned their systems to the specific characteristics of this subtask, i.e., to the fact that it deals with quantification (as opposed to classification) and to the fact that it has an ordinal (as opposed to binary) nature.

In particular, the top-scoring team (QCRI1) used a novel algorithm explicitly designed for ordinal quantification, that leverages an ordinal hierarchy of binary probabilistic quantifiers.

Team NRU-HSE4 uses an existing quantification approach based on cost-sensitive learning, and adapted it to the ordinal case.

Team ISTI-CNR6 instead used a novel adaptation to quantification of a tree-based approach to ordinal regression.

Teams LYS7 and HSENN9 also used an existing quantification approach, but did not exploit the ordinal nature of the problem.

The other teams mostly used approaches based on “classify and count” (see Section 5.4), and viewed the problem as single-label multi-class (instead of ordinal) classification; some of these teams (notably, team Finki2) obtained very good results, which testifies to the quality of the (general-purpose) features and learning algorithm they used.

6 Conclusion and Future Work

We described SemEval-2016 Task 4 “Sentiment Analysis in Twitter”, which included five subtasks including three that represent a significant departure from previous editions. The three new subtasks focused, individually or in combination, on two variants of the basic “sentiment classification in Twitter” task that had not been previously explored within SemEval. The first variant adopts a five-point scale, which confers an ordinal character to the classification task. The second variant focuses on the correct estimation of the prevalence of each class of interest, a task which has been called quantification in the supervised learning literature. In contrast, previous years’ subtasks have focused on the correct labeling of individual tweets. As in previous years (2013–2015), the 2016 task was very popular and attracted a total of 43 teams.

A general trend that emerges from SemEval-2016 Task 4 is that most teams who were ranked at the top in the various subtasks used deep learning, including convolutional NNs, recurrent NNs, and (general-purpose or task-specific) word embeddings. In many cases, the use of these techniques allowed the teams using them to obtain good scores even without tuning their system to the specifics of the subtask at hand, e.g., even without exploiting the ordinal nature of the subtask – for Subtasks C and E – or the quantification-related nature of the subtask – for Subtasks D and E. Conversely, several teams that have indeed tuned their system to the specifics of the subtask at hand, but have not used deep learning techniques, have performed less satisfactorily. This is a further confirmation of the power of deep learning techniques for tweet sentiment analysis.

Concerning Subtasks D and E, if quantification-based subtasks are proposed again, we think it might be a good idea to generate, for each test topic \( t_i \), multiple “artificial” test topics \( t_{i1}, t_{i2}, \ldots \), where class prevalences are altered with respect to the ones of \( t_i \) by means of selectively removing from \( t_i \) tweets belonging to a certain class. In this way, the evaluation can take into consideration (i) class prevalences in the test set and (ii) levels of distribution drift (i.e., of the divergence of the test distribution from the training distribution) that are not present in the “naturally occurring” data.
By varying the amount of removed tweets at will, one may obtain many test topics, thus augmenting the magnitude of the experimentation at will while at the same time keeping constant the amount of manual annotation needed.

In terms of possible follow-ups of this task, it might be interesting to have a subtask whose goal is to distinguish tweets that are neutral about the topic (i.e., do not express any opinion about the topic) from tweets that express a fair opinion (i.e., lukewarm, intermediate between positive and negative) about the topic.

Another possibility is to have a multi-lingual tweet sentiment classification subtask, where training examples are provided for the same topic for two languages (e.g., English and Arabic), and where participants can improve their performance on one language by leveraging the training examples for the other language via transfer learning. Alternatively, it might be interesting to include a cross-lingual tweet sentiment classification subtask, where training examples are provided for a given language (e.g., English) but not for the other (e.g., Arabic); the second language could be also a surprise language, which could be announced at the last moment.

References

[Abdelwahab and Elmaghraby2016] Omar Abdelwahab and Adel Elmaghraby. 2016. UoL at SemEval-2016 Task 4: Multi domain word2vec for Twitter sentiment classification. In Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016), San Diego, US.

[Astudillo and Amir2016] Ramón Astudillo and Silvio Amir. 2016. INESC-ID at SemEval-2016 Task 4: Reducing the problem of out-of-embedding words. In Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016), San Diego, US.

[Attardi and Sartiano2016] Giuseppe Attardi and Daniele Sartiano. 2016. UniPi at SemEval-2016 Task 4: Convolutional neural networks for sentiment classification. In Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016), San Diego, US.

[Baccianella et al.2009] Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. 2009. Evaluation measures for ordinal regression. In Proceedings of the 9th IEEE International Conference on Intelligent Systems Design and Applications (ISDA 2009), pages 283–287, Pisa, IT.

[Baccianella et al.2010] Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. 2010. SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In Proceedings of the 7th Conference on Language Resources and Evaluation (LREC 2010), Valletta, MT.

[Balahur2016] Alexandra Balahur. 2016. OPAL at SemEval-2016 Task 4: the Challenge of Porting a Sentiment Analysis System to the "Real" World. In Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016), San Diego, US.

[Balikas and Amini2016] Georgios Balikas and Massih-Reza Amini. 2016. TwiSE at SemEval-2016 Task 4: Twitter sentiment classification. In Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016), San Diego, US.

[Bollen et al.2011] Johan Bollen, Huina Mao, and Xiaojun Zeng. 2011. Twitter mood predicts the stock market. Journal of Computational Science, 2(1):1–8.

[Borge-Holthoefer et al.2015] Javier Borge-Holthoefer, Walid Magdy, Kareem Darwish, and Ingmar Weber. 2015. Content and network dynamics behind Egyptian political polarization on Twitter. In Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work and Social Computing (CSCW 2015), pages 700–711, Vancouver, CA.

[Briones and Amarasinghe2016] Gerardo Briones and Kasun Amarasinghe. 2016. VCU-TSA at SemEval-2016 Task 4: Sentiment analysis in Twitter. In Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016), San Diego, US.

[Burton and Soboleva2011] S. Burton and A. Soboleva. 2011. Interactive or reactive? Marketing with Twitter. Journal of Consumer Marketing, 28(7):491–499.

[Castillo et al.2016] Esteban Castillo, Ofelia Cervantes, Darnes Vilariño, and David Báez. 2016. UDLAP at SemEval-2016 Task 4: Sentiment quantification using a graph based representation. In Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016), San Diego, US.

[Ciubotariu et al.2016] Calin-Cristian Ciubotariu, Marius-Valentin Hrisca, Mihail Gliga, Diana Daraban, Diana Trandabat, and Adrian Iftene. 2016. Minions at SemEval-2016 Task 4: Or how to boost a student’s self esteem. In Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016), San Diego, US.

[Cozza and Petrocchi2016] Vittoria Cozza and Marinella Petrocchi. 2016. mib at SemEval-2016 Task 4: Exploiting lexicon based features for sentiment analysis in Twitter. In Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016), San Diego, US.
[Da San Martino et al.2016] Giovanni Da San Martino, Wei Gao, and Fabrizio Sebastiani. 2016. QCRI at SemEval-2016 Task 4: Probabilistic methods for binary and ordinal quantification. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016)*, San Diego, US. Forthcoming.

[Deriu et al.2016] Jan Deriu, Maurice Gonzenbach, Fatih Uzdil, Aurelien Lucchi, Valeria De Luca, and Martin Jaggi. 2016. SwissCheese at SemEval-2016 Task 4: Sentiment classification using an ensemble of convolutional neural networks with distant supervision. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016)*, San Diego, US.

[Dodds et al.2011] Peter S. Dodds, Kameron D. Harris, Isabel M. Kloumann, Catherine A. Bliss, and Christopher M. Danforth. 2011. Temporal patterns of happiness and information in a global social network: Hedonometrics and Twitter. *PLoS ONE*, 6(12).

[Du and Zhang2016] Steven Du and Xi Zhang. 2016. Acyber at SemEval-2016 Task 4: i-vector based sentence representation. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016)*, San Diego, US.

[Esuli and Sebastiani2010] Andrea Esuli and Fabrizio Sebastiani. 2010. Sentiment quantification. *IEEE Intelligent Systems*, 25(4):72–75.

[Esuli and Sebastiani2015] Andrea Esuli and Fabrizio Sebastiani. 2015. Optimizing text quantifiers for multivariate loss functions. *ACM Transactions on Knowledge Discovery and Data*, 9(4):Article 27.

[Esuli2016] Andrea Esuli. 2016. ISTI-CNR at SemEval-2016 Task 4: Quantification on an ordinal scale. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016)*, San Diego, US.

[Florean et al.2016b] Cosmin Florean, Oana Bejenaru, Edgard Apostol, Octavian Ciobanu, Adrian Iftene, and Diana Trandabat. 2016. SentimentalTists at SemEval-2016 Task 4: Building a Twitter sentiment analyzer in your backyard. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016)*, San Diego, US.

[Forman2005] George Forman. 2005. Counting positives accurately despite inaccurate classification. In *Proceedings of the 16th European Conference on Machine Learning (ECML 2005)*, pages 564–575, Porto, PT.

[Forman2008] George Forman. 2008. Quantifying counts and costs via classification. *Data Mining and Knowledge Discovery*, 17(2):164–206.

[Friedrichs2016] Jasper Friedrichs. 2016. IIP at SemEval-2016 Task 4: Prioritizing classes in ensemble classification for sentiment analysis of tweets. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016)*, San Diego, US.

[Gao and Oates2016] Hang Gao and Tim Oates. 2016. MDSENT at SemEval-2016 Task 4: Supervised system for message polarity classification. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016)*, San Diego, US.

[Giorgis et al.2016] Stavros Giorgis, Apostolos Rousas, John Pavlopoulos, Prodomos Malakasiotis, and Ion Androutsopoulos. 2016. aueb.twitter.sentiment at SemEval-2016 Task 4: A weighted ensemble of SVMs for Twitter sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016)*, San Diego, US.

[Gomez et al.2016] Paula Gomez, Darnes Vilarinho, Grigori Sidorov, and David Pinto Avendaño. 2016. CICBUAPnlp at SemEval-2016 Task 4: Discovering Twitter polarity using enhanced embeddings. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016)*, San Diego, US.

[Hamdan2016] Hussam Hamdan. 2016. SentiSys at SemEval-2016 Task 4: Feature-based system for sentiment analysis in Twitter. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016)*, San Diego, US.

[He et al.2016] Yunchao He, Liang-Chih Yu, Chin-Sheng Yang, K. Robert Lai, and Weiyi Liu. 2016. YZUNLP at SemEval-2016 Task 4: Ordinal sentiment classification using a recurrent convolutional network. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016)*, San Diego, US.

[Jahren et al.2016] Brage Ekroll Jahren, Valerij Fredriksen, Björn Gambäck, and Lars Bungum. 2016. NTNUSentEval at SemEval-2016 Task 4: Combining general classifiers for fast Twitter sentiment analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016)*, San Diego, US.

[Juncal-Martín et al.2016] Jonathan Juncal-Martín, Tamara Álvarez-López, Milagros Fernández-Gavilanes, Enrique Costa-Montenegro, and Francisco Javier González-Castaño. 2016. GTI at SemEval-2016 Task 4: Training a naive Bayes classifier using features of an unsupervised system. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016)*, San Diego, US.

[Karpov et al.2016] Nikolay Karpov, Alexander Porshnev, and Kirill Rudakov. 2016. NRU-HSE at SemEval-2016 Task 4: Discovering Twitter polarity using enhanced embeddings. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016)*, San Diego, US.

[Kaya et al.2013] Mesut Kaya, Guven Fidan, and Ismail Hakki Toroslu. 2013. Transfer learning using Twitter data for improving sentiment classification of Turkish political news. In *Proceedings of the 28th In-
9th International Workshop on Semantic Evaluation (SemEval 2015), pages 451–463, Denver, US.

[Rouvier and Favre2016] Mickael Rouvier and Benoit Favre. 2016. SENSEI-LIF at SemEval-2016 Task 4: Polarity embedding fusion for robust sentiment analysis. In Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016), San Diego, US.

[Rubner et al.2000] Yossi Rubner, Carlo Tomasi, and Leonidas J. Guibas. 2000. The Earth Mover’s Distance as a metric for image retrieval. International Journal of Computer Vision, 40(2):99–121.

[Ruder et al.2016] Sebastian Ruder, Parsa Ghaffari, and John G. Breslin. 2016. INSIGHT-1 at SemEval-2016 Task 4: Convolutional Neural Networks for Sentiment Classification and Quantification. In Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016), San Diego, US.

[Rüschendorf2001] Ludger Rüschendorf. 2001. Wasserstein metric. In Michiel Hazewinkel, editor, Encyclopedia of Mathematics. Kluwer Academic Publishers, Dordrecht, NL.

[Sarker2016] Abeed Sarker. 2016. DIEGOLab16 at SemEval-2016 Task 4: Sentiment analysis in Twitter using centroids, clusters, and sentiment lexicons. In Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016), San Diego, US.

[Sebastiani2015] Fabrizio Sebastiani. 2015. An axiomatically derived measure for the evaluation of classification algorithms. In Proceedings of the 5th ACM International Conference on the Theory of Information Retrieval (ICTIR 2015), pages 11–20, Northampton, US.

[Sidarenka2016] Uladzimir Sidarenka. 2016. Pots at SemEval-2016 Task 4: Sentiment analysis of Twitter using character-level convolutional neural networks. In Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016), San Diego, US.

[Stojanovski et al.2016] Dario Stojanovski, Gjorgji Strezoski, Gjorgji Madjarov, and Ivica Dimitrovski. 2016. Finki at SemEval-2016 Task 4: Deep learning architecture for Twitter sentiment analysis. In Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016), San Diego, US.

[Vilares et al.2016] David Vilares, Yerai Doval, Miguel A. Alonso, and Carlos Gómez-Rodríguez. 2016. LYS at SemEval-2016 Task 4: Exploiting neural activation values for Twitter sentiment classification and quantification. In Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016), San Diego, US.

[Xu et al.2016] Steven Xu, Huizhi Liang, and Tim Baldwin. 2016. UNIMELB at SemEval-2016 Task 4: An ensemble of neural networks and a word2vec based model for sentiment classification. In Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016), San Diego, US.

[Yadav2016] Vikrant Yadav. 2016. thecerealkiller at SemEval-2016 Task 4: Deep learning based system for classifying sentiment of tweets on two point scale. In Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016), San Diego, US.

[Zhang et al.2016] Zhengchen Zhang, Chen Zhang, Dongyan Huang, Wu Fuxiang, Weisi Lin, and Minghui Dong. 2016. I2RNTU at SemEval-2016 Task 4: Classifier fusion for polarity classification in Twitter. In Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016), San Diego, US.

[Zhou et al.2016] Yunxiao Zhou, Zhihua Zhang, and Man Lan. 2016. ECNU at SemEval-2016 Task 4: An empirical investigation of traditional NLP features and word embedding features for sentence-level and topic-level sentiment analysis in Twitter. In Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval 2016), San Diego, US.
| Subtasks | Team ID | Affiliation | Nation | Paper |
|----------|---------|-------------|--------|-------|
| A        | Axyber  | Axyber.com   | Singapore, China | (Du and Zhang, 2016) |
| A        | aaehtwrt-sentiment | Department of Informatics, Athens University of Economics and Business | Greece | (Giorgi et al., 2016) |
| A        | CICBUAUpul | Instituto Politécnico Nacional Benemérita Universidad Autonoma de Puebla | Mexico | (Gómez et al., 2016) |
| A B      | CUFE    | Cairo University | Egypt | (Nabi et al., 2016) |
| A D      | DIEGOLab16 | Arizona State University | USA | (Kartek, 2016) |
| A B C D E | DNIC-ELIRF | Universitat Politècnica de València | Spain | (Mirnait et al., 2016) |
| A B C D E | ECNU | East China Normal University | China | (Zhou et al., 2016) |
| A D      | ELIRF   | Universitat Politècnica de València | Spain | |
| B C D E  | Finki   | Saints Cyril and Methodius University, Skopje | Macedonia | (Bojanovski et al., 2016) |
| A B D    | GTI     | AtlantIC Centre, University of Vigo | Spain | (Juncal-Martínez et al., 2016) |
| D E      | HSENN   | National Research University Higher School of Economics | Russia | |
| A        | ID2NTU  | Institute for Infocomm Research, A*STAR School of Computer Engineering, Nanyang Technological University | Singapore | (Zhang et al., 2016) |
| A        | IIP     | Infosys Limited | India | (Friedrich, 2016) |
| A        | INESC-ID| INESC-ID, Lisboa Instituto Superior Técnico, Universidade de Lisboa | Portugal | (Amudillo and Amir, 2016) |
| A B C D E | INSIGHT-1 | INSIGHT Research Centre, National University of Ireland, Galway AYLien Inc. | Ireland | (Ruder et al., 2016) |
| A B C D E | LYS     | Universidade da Corunha Universidade de Vigo | Spain | (Vilares et al., 2016) |
| A        | MDSENT  | University of Maryland Baltimore County | USA | (Gao and Oates, 2016) |
| A        | mib     | Istituto di Informatica e Telematica, Consiglio Nazionale delle Ricerche | Italy | (Cozza and Petricchi, 2016) |
| A        | Minions | University of Iasi | Romania | (Cuobotaria et al., 2016) |
| A B C D E | IETI-CNRI | Istituto di Scienza e Tecnologie dell’Informazione, Consiglio Nazionale delle Ricerche | Italy | (Koth, 2016) |
| D E      | KIE-HSE | National Research University Higher School of Economics Russia | Russia | (Karpov et al., 2016) |
| A        | NTNUSentEval | Norwegian University of Science and Technology | Norway | (Habib et al., 2016) |
| A B      | OPAL    | European Commission Joint Research Centre | Italy | (Balaban, 2016) |
| B C D E  | pkublab | Peking University | China | |
| B C D E  | PotTS   | University of Potsdam Retresco GmbH | Germany | (Sidorenko, 2016) |
| A B C    | PUT     | Poznan University of Technology | Poland | (Lango et al., 2016) |
| D E      | QCRI (**) | Qatar Computing Research Institute | Qatar | (Da San Martino et al., 2016) |
| A        | SENSEI-LIF | Aix-Marseille University - CNRS - LIF | France | (Rouveret and Favre, 2016) |
| A B C    | SentimentItis | National Research University Higher School of Iasi Romania | Romania | (Florea et al., 2016) |
| A        | SentiSys | Aix-Marseille University | France | (Hamdan, 2016) |
| A        | SteM    | Sabanci University Marmara University Otto-von-Guericke University Magdeburg | Turkey | (Räbiger et al., 2016) |
| A B D    | SwissCheese | ETH Zürich | Switzerland | (Oeri et al., 2016) |
| B        | thecerealkiller | Amazon.in | India | (Yadav, 2016) |
| A        | THUIR   | Tsinghua University | China | |
| A B      | Tweeter | School of ECE, National Technical University of Athens School of ECE, Technical University of Crete Department of Informatics, University of Athens Signal Analysis and Interpretation Laboratory (SAIL) Institute for Language & Speech Processing - ILSP | Greece | (Palogiannis et al., 2016) |
| A B C D E | TwSE | University of Grenoble-Alpes | France | (Balkas and Amini, 2016) |
| D        | UDLAP   | Universidad de las Américas Puebla (UDLAP) | Mexico | (Castillo et al., 2016) |
| A B      | UNIMELB | University of Melbourne | Australia | (Xu et al., 2016) |
| A        | UniPI   | Università di Pisa | Italy | (Attardi and Santiano, 2016) |
| A        | UoiL    | University of Louisville | USA | (Abdulkwabab and Elmaghraby, 2016) |
| A B      | VCU-TSA | Virginia Commonwealth University | USA | (Briones and Amarasinghe, 2016) |
| A B      | WR      | WR | Hong Kong | |
| C        | YZU-NLP | Yuan Ze University, Tanyuan Yunnan University, Kunming | Taiwan | (He et al., 2016) |

Table 15: Participating teams (Column 2), their affiliation (Column 3) and nationality (Column 4), the subtasks they have participated in (Column 1), and the paper they have contributed (Column 5). Teams whose “Affiliation” column is typeset on more than one row include researchers with different affiliations. Teams marked with a (**) include some of the SemEval 2016 Task 4 organizers. An empty entry for the “Paper” column indicates that the team have not contributed a system description paper.