Report on Figurative Language Annotation on Tweets

Elena Alvarez Mellado, Julia Cathcart, Qingwen Ye

1. Introduction

Speakers don't always express literally what they mean. Sometimes, what is being said is not exactly what the speaker means. A speaker can say something and, in fact, mean exactly the opposite, as we can see in sarcasm, exaggeration or metaphor. Figurative language is a very creative linguistic process that can be easily found in everyday language: on the press, on literature, in casual conversation. Humans are perfectly capable of dealing with this mismatch between what is literally being said and what is actually being conveyed by the message. Machines, however, find it hard to figure out whether a statement literally means what it expresses or carries extended meaning through figurative language. Therefore, figurative language poses a big challenge for Natural Language Processing applications.

The goal of this project is to explore the phenomenon of non literal language on Twitter. We will annotate a corpus of tweets rich of non literal language and distinguish whether a tweet is literal or figurative, and if figurative, it will be labeled one of the six figure of speech tags (sarcasm, hyperbole, simile, metaphor, rhetorical question, and other). With the data from the annotation, we will train a classifier to do automatic classification of figurative and literal tweets.

2. Corpus

The starting point for this project is the corpus from the Sentiment Analysis of Figurative Language in Twitter task at SemEval 2015[1]. The task consisted on performing sentiment analysis classification of non literal tweets. The original corpus consist of more than 4700 tweets, most of them non literal in some way. The corpus was annotated with sentiment analysis scores, but contained no annotations whatsoever regarding literality.

[1] Corpus and task available at http://alt.qcri.org/semeval2015/task11/

The corpus was scrapped from Twitter using keywords and hashtags that tend to be associated with figurative language, such as *#sarcasm*, *#not* or even *literally* (which, ironically, is in the midst of a semantic change process and tends to be used to express that something is not literal). The way the corpus was collected, however, causes the corpus to be almost entirely comprised of sarcastic tweets. This is, in fact, interesting from a linguistic point of view: because sarcasm is extremely context-dependent, speakers need to mark or make it obvious somehow that the message is actually sarcastic (otherwise, the message could be misunderstood). This means that sarcastic tweets can be easily retrieved automatically (scrapping tweets that contain such hashtags or keywords), while other figurative language (like metaphor, exaggeration or rhetorical questions) tend to be more obvious and don't require a specific hashtag or keyword that warns the reader that the message is not literal.

Although this issue is linguistically interesting, it makes the corpus extremely biased towards sarcasm, leaving other figurative uses in the corpus extremely underrepresented. As a result, our classification annotation, gold standard and classifier are also extremely biased towards sarcasm.

3. Scheme and guidelines

For our annotation scheme we have considered three main tags: `Literal` (for tweets that are literal), `Figurative` (for non literal tweets) and `Not_enough_context`(if not enough information was provided). The `Figurative` class had six different possibilities depending on the type of non literal language:

- `Literal:` what the tweet says is literally what the speaker means.
- `Figurative:` what the tweet says is not literally what the speaker means.
    - `Sarcasm:` the tweet says exactly the opposite of what the speaker means.

      Example*: Having to run to the train first thing in the morning is a great way to start the day #not*
    - `Hyperbole:` the tweet is an exaggeration over what the speaker really means.

Example*: Literally, I'm about to offer up my first born child just to the stress gods. #breakingdown#gradschool #finals #happybirthday #not*

- `Simile:` the tweet explicitly portrays a comparison between two things, establishing an analogy (*This is like something else*).

  Example*: A racist NBA owner makes about as much sense as a homophobic theater producer.*

- `Metaphor:` the tweet implicitly portrays a comparison between two things.

  Example*: When u allow people to control u and/or ur emotions, u then become a puppet & they will pull ur strings whenever they like..... #not #here*

- `Rhetorical question:` the tweet contains a question that does not inted to be answered.

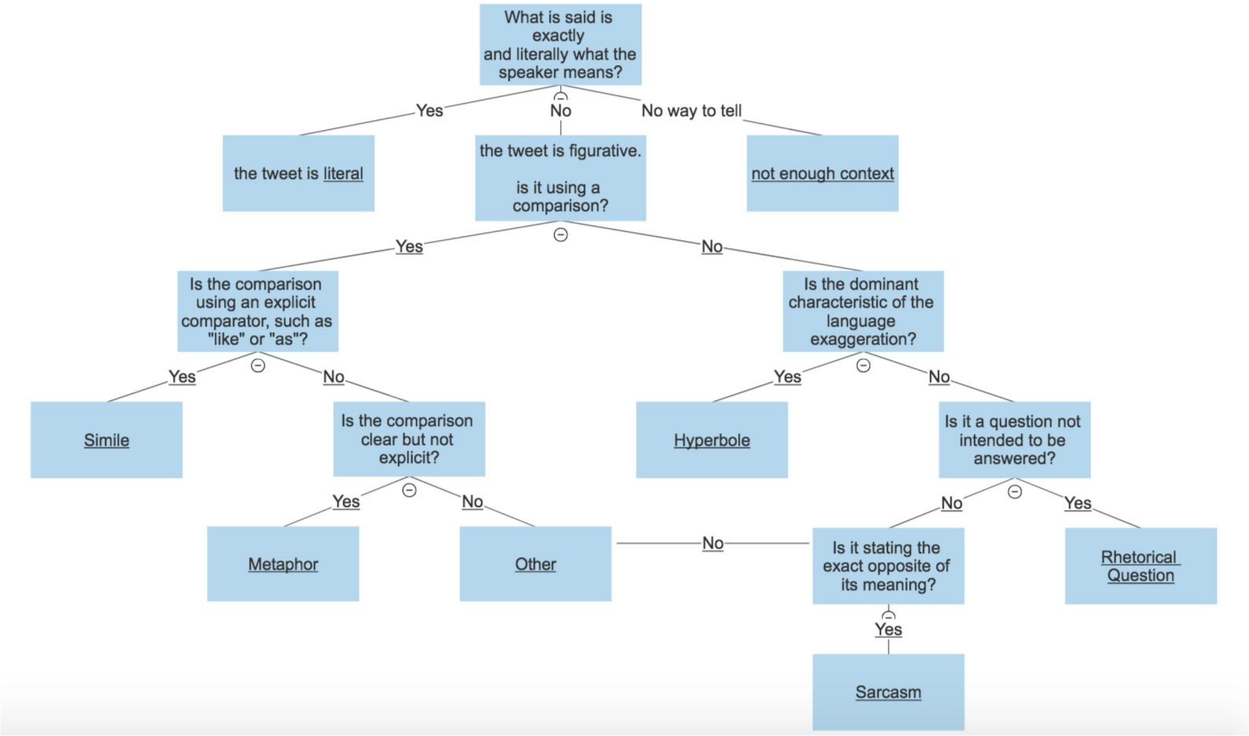
  Example: *Really hurting right now... How does one wake up feeling fine and acquire a hangover throughout the day? It #does #not #make #sense*

- `Other:` for other non literal phenomena that may occur.

- `Not_enough_context:` figurative language is extremely context-dependent. Sometimes, the tweet might be a reply to a previous message, or may context a link or an image that is not part of the corpus. In this situation, even a human could not be able of knowing whether a certain message is literal or not. This category covers those situations.

The scheme included a certainty attribute aimed at capturing the degree of certainty that the annotator had over their annotation. This attribute was originally intended to measure the solidity of the scheme throughout the sample annotation and see if the guidelines were useful enough for the annotators.

The guidelines described the different tags and provided examples and tips to detect the different classes. One of the main issues encountered through the process was that certain categories could overlap. What if a question is rhetorical and exaggeration at the same time? What if a tweet contains a sarcastic metaphor? In order to cover those cases, we added to the guidelines a decision tree to help annotators decide what label to annotate for every case.


What is said is exactly and literally what the speaker means?
Yes
No
No way to tell
the tweet is literal
the tweet is figurative. is it using a comparison?
not enough context
Yes
No
Is the comparison using an explicit comparator, such as "like" or "as"?
Is the dominant characteristic of the language exaggeration?
Yes
No
Yes
No
Simile
Is the comparison clear but not explicit?
Hyperbole
Is it a question not intended to be answered?
Yes
No
No
Yes
Metaphor
Other
No
Is it stating the exact opposite of its meaning?
Rhetorical Question
Yes
Sarcasm

4. Annotation process

Three annotators took part in the annotation process. All of them received the same annotation guidelines and instructions and used MAE[2] as annotation environment.

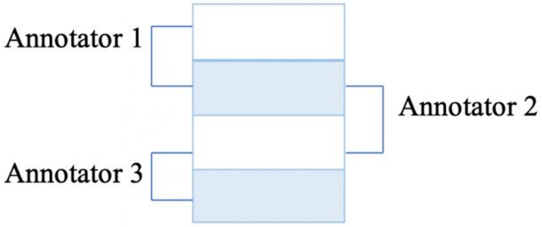
4.1 Data preparation

The corpus consisted of 600 random tweets extracted from the original corpus from the SemEval task. The annotation was divided in two batches: the first one was a smaller batch

[2] https://github.com/keighrim/mae-annotation/wiki

that was intended as a sample annotation to test the guidelines and check how comfortable annotators were with the suggested scheme. The second annotation batch was the full and bigger annotation batch. Each batch was evenly divided into four parts and assigned among the three annotators as follows:

1) tweets for sample annotation (120 tweets)
2) tweets for full annotation (480 tweets)



The aim of this distribution (with overlapping pairs) was to ensure having data annotated twice, while having as many tweets as possible. Due to our limited amount of time, we gave Annotators 1 and 3 files that had about 100 tweets each overlapping with Annotator 2, in order to calculate inter-annotator agreement and establish a gold standard. The remaining tweets in each of their files were not overlapping with the other annotators. This was a trade-off we made in order to get more data in the limited amount of time that we had.

4.2 Inter-annotator agreement

Due to the way we split the data, we calculated the IAA between Annotator 1 and 2 as well as between Annotator 2 and 3. Detailed results with the IAA are shown in Table 1.

IAA on full annotation	Annotator 1 & 2	Annotator 2 & 3
<Cross-tag>Multi-Pi (Fleiss' Kappa) [Figurative, Literal, Not_enough_context]	0.6133	0.2206
<Cross-tag> Multi-Kappa (Huberts' Kappa) [Figurative, Literal, Not_enough_context]	0.6184	0.2804
<Tag-level> Multi-Pi (Fleiss' Kappa) Figurative::type	0.7874	0.7264

Table 1: IAA results

Overall, taking all the categories into consideration, we got substantial agreement between Annotator 1 and 2, and uneven agreement for Annotator 2 and 3.

We got some interesting numbers focusing on the figurative tag only. As we can see from row 4, we got some negative numbers for the IAA, while in row 5, we got pretty high agreement in both groups of annotators.

About the low IAA for the figurative tag, one reason may be that there are six subtags under the figurative category, yet there may be more than one figure of speech applied in a tweet, hence the disagreement. Another explanation is the influence of the confidence attribute we defined for the task. In addition to tagging the tweet, annotators are asked to indicate their level of certainty (with a scale of 1~3 matching high to low confidence). Even when two annotators got the same tag for a tweet, their levels of certainty may differ. Thus, we recalculated the IAA opting out the certainty level attribute and got a much better result.

4.3 Adjudication and gold standard

Once the corpus was annotated and IAA calculated, we went manually through all the tweets on the annotated corpus using the adjudication feature from. Tweets that had been labeled with the same tag by both annotators went straight into the gold standard (ignoring the certainty attribute). For thoses tweets where there was a disagreement between annotators, we decided which one of the two options was best and assign that one for the gold standard. In general terms, the major disagreement was found for `literal` vs `figurative` vs `not enough context`, while the agreement among figurative types was quite consistent (also due to the fact that it was mainly `sarcasm`).

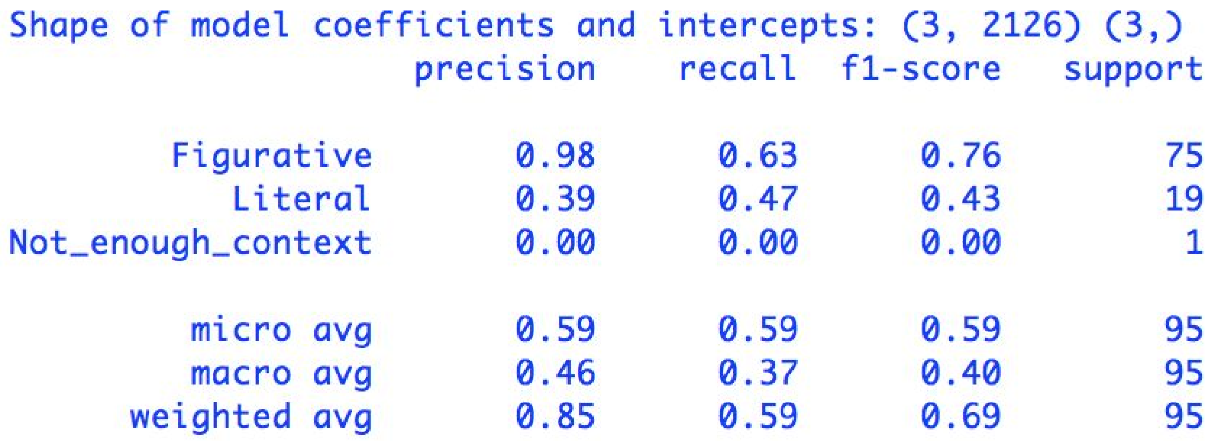
5. Classifier

The resulting gold standard was used to feed our figurative language classifier. In order to transform the gold standard we used the Python module xml. The aim was to parse the xml format produced by MAE and also link the tweet IDs (that had the annotation tag) to the actual tweet texts (to feed the classifier). The gold standard was parsed, transformed via Python into a dictionary, splitted into training set (80%) and test set (20%) and saved for feeding the classifier as a csv. We did this three times to get three seperate data sets. The first was the tweets labeled with either figurative, literal, or not enough context. The second with labeled data for each type of figurative language (as well as the same labels for the literal tweets and tweets without enough context). The third was created after we received the results from our first two data sets on the classifier. When we realized that we had unexpectedly so much data for sarcasm, we created a dataset for all the tweets labeled as sarcasm, and all of the other tweets as non-sarcasm. We removed the not enough context files from this data set because that gave us better accuracy.

The classifier we built is a logistic classifier that uses a Bag Of Words (BOW) as features. The classifier was implemented using the Python library sklearn. It was based on the framework used for our first assignment in this class.

Two classification tasks were performed using the classifier: a general literal vs figurative classification task (ignoring the attributes types for figurative tweets: sarcasm, hyperbole, etc); a specific one that took into account the different figurative types. The general classification produced fair results, especially for the `Figurative` detection. The results for the categories was more modest, with some very disappointing zero results. These results, however, are very likely caused by the fact that the figurative tweets consisted mostly of sarcastic tweets, with hardly any other categories represented. This issue is a consequence of the way the original SemEval corpus was created.

The following are the results for classifying our tweets into categories for figurative and literal. Our F1 score for figurative tweets was 0.76 and for literal tweets it was 0.43.

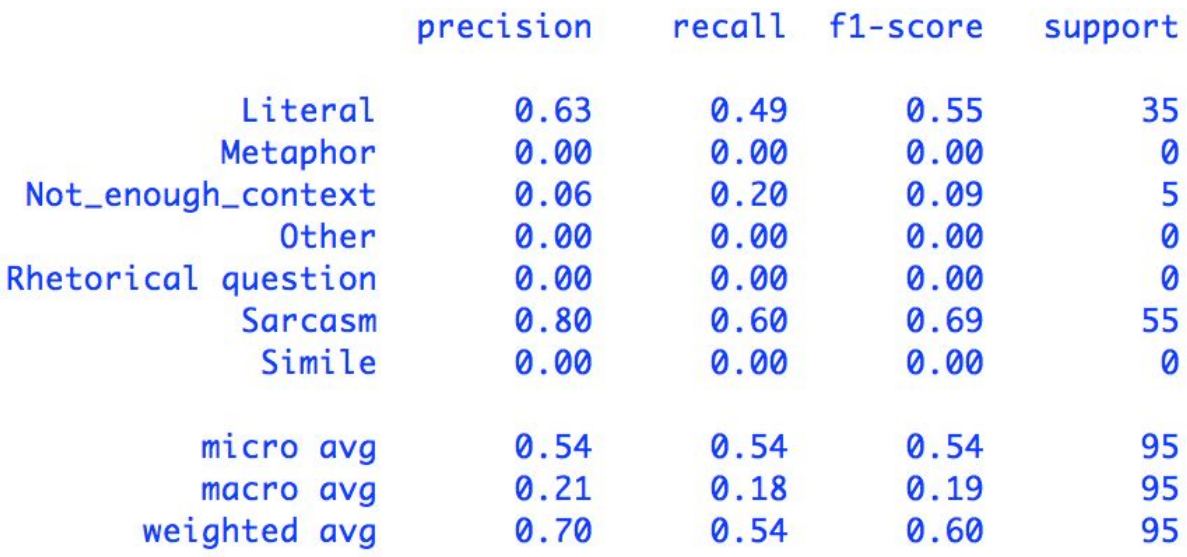


```
Shape of model coefficients and intercepts: (3, 2126) (3,)
                   precision    recall  f1-score   support

       Figurative       0.98      0.63      0.76        75
          Literal       0.39      0.47      0.43        19
Not_enough_context       0.00      0.00      0.00         1

        micro avg       0.59      0.59      0.59        95
        macro avg       0.46      0.37      0.40        95
     weighted avg       0.85      0.59      0.69        95
```

Following testing our classifier on classifying literal vs figurative tweets, we used it to classify each type of figurative label that we received for the annotators. Our results for this (below) were not very good, but we noticed that the F1 for sarcasm was noticeably high compared to the other categories. We believe this is due in large part to the fact that we ended up with a large amount of data labeled as sarcastic. Also, sarcastic tweets tend to have a very common format where they are often given a hashtag such as “#not” or “#sarcasm” which makes them easier to detect.

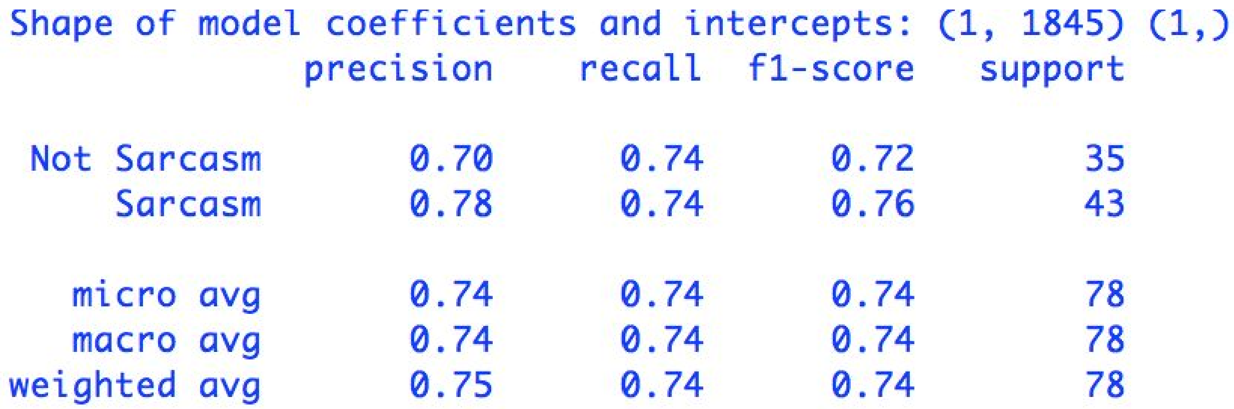


```
                    precision    recall  f1-score   support

           Literal       0.63      0.49      0.55        35
          Metaphor       0.00      0.00      0.00         0
Not_enough_context       0.06      0.20      0.09         5
             Other       0.00      0.00      0.00         0
Rhetorical question       0.00      0.00      0.00         0
           Sarcasm       0.80      0.60      0.69        55
            Simile       0.00      0.00      0.00         0

         micro avg       0.54      0.54      0.54        95
         macro avg       0.21      0.18      0.19        95
      weighted avg       0.70      0.54      0.60        95
```

Since the sarcastic tweets are such a huge part of the data set, we also believed that the detection of sarcastic tweets was also affecting how our classifier was operating on the figurative vs literal labels (above). So, we ran the classifier where we considered each label that was not sarcastic (and not "not enough context") as Not Sarcastic, and used the classifier to classify tweets as either sarcastic or not sarcastic. As the results for this were very good, we then removed all of the "not enough context" tweets to get the final version of our classifier that resulted in an F1 of 0.72 for not sarcastic tweets and 0.76 for sarcastic tweets (results below).



```
Shape of model coefficients and intercepts: (1, 1845) (1,)
             precision    recall  f1-score   support

 Not Sarcasm       0.70      0.74      0.72        35
     Sarcasm       0.78      0.74      0.76        43

   micro avg       0.74      0.74      0.74        78
   macro avg       0.74      0.74      0.74        78
weighted avg       0.75      0.74      0.74        78
```

6. Conclusions

We have explored the difficulties that annotating and detecting non literal language poses. In addition, we have also experimented how determinant the quality of the corpus is terms of the classification tasks: although the SemEval corpus was very interesting as starting point, the way the corpus was collected made it extremely biased towards one type only of non literal language, which has had a big impact in our classification accuracy.

As a result of this project, we have presented: an annotation scheme and guidelines for non literal language annotation on Twitter, a gold standard of 600 tweets annotated with literal/figurative information and a logistic regression classifier that classifies whether a tweet is figurative or not and the type of figurative resource.

Although the classifier results are modest for the different types of figurative language, the literal vs figurative classification task result is pretty decent and the achieved inter-annotator agreement proves that the annotation guidelines and scheme are solid.

Future work that we could explore would include considering more sophisticated features for the classifier and testing both the annotation and the classifier on a more varied corpus that covers more types of non literal phenomena.