Fortia-FBK at SemEval-2017 Task 5

Bullish or Bearish? Inferring Sentiment towards Brands from Financial News Headlines

Youness Mansar⋆, Lorenzo Gatti°, Sira Ferradans⋆, Marco Guerini° and Jacopo Staiano⋆

⋆Fortia Financial Solutions, Paris, France °Fondazione Bruno Kessler, Povo, Italy

{youness.mansar,sira.ferradans,jacopo.staiano}@fortia.fr
{l.gatti,guerini}@fbk.eu

Goal: predicting sentiment of financial news

The goal of SemEval-2017 Task 5 is to perform sentiment detection on financial headlines (Subtask 2).
Given a headline and a target company, the system has to predict how positive (*bullish*; e.g. believing that the stock price will increase) or how negative (*bearish*; e.g. believing that the stock price will decline) the sentence is with respect to the target company.
For example *(targets in bold)*:

Very positive (+0.814)
Sainsbury's and **Glencore** give FTSE a three-digit lift - London Report

Positive (+0.314)
Insurers: **Admiral** blows hot and cold but Aviva soars pre-Friends Life merger

Neutral (+0.002)
RSA Insurance Hires **Towergate**'s Egan as Chief Financial Officer

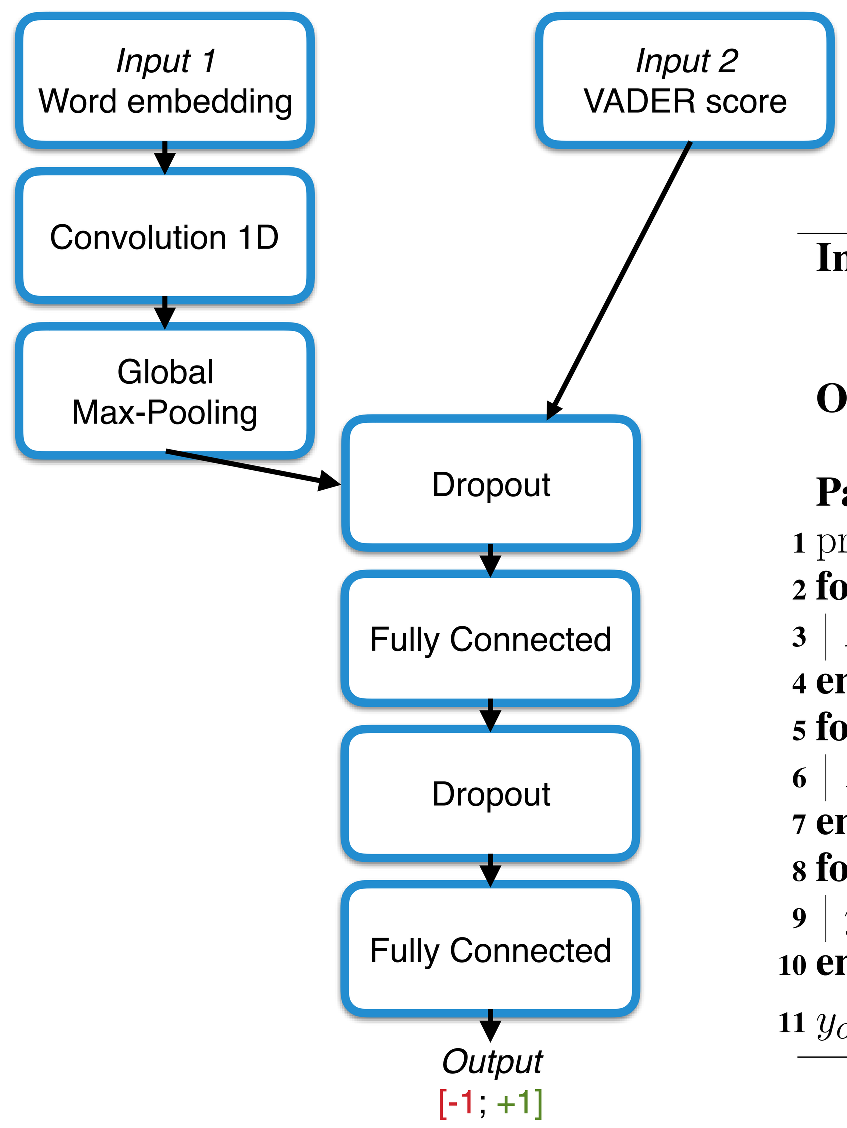
Negative (-0.314)
REFILE-Aviva Investors to move 34 bln euros in assets from **AXA** fund arm

Very negative (-0.902)
UPDATE 1-**Dairy Crest** loses a third of Morrisons milk contract

Fortia-FBK: the best performing system

Here we present the architecture of Fortia-FBK, the best performing system at Semeval-2017 Task 5, subtask 2.
The system is based on 1D convolutions, and uses as input *i)* pre-trained word embeddings (`GloVe` vectors trained on Wikipedia and GigaWord), *ii)* the `DepecheMood` affective lexicon, and *iii)* a rule-based sentence-level sentiment model (`VADER`)

System architecture & test/train algorithm



Input : S (set of training instances) with ground-truth scores y and set of test sentences S_o
Output : M (set of trained models) and predictions y_o for test set S_o
Params: Number N of models to train

preprocess(X)
foreach s_i ***in*** S **do**
  $X_i = \text{sentence\_representation}(s_i)$
end
foreach $n \in N$ **do**
  $M_n = \min \text{Loss}(X)$
end
foreach $n \in N$ **do**
  $y_n = \text{evaluate}(X_o, M_n)$
end
$y_o(u) = \frac{1}{N} \sum_n^N y_n(u)$

Method

Preprocessing: sentences are tokenized, then target companies' names are replaced with `<company>` and numbers with `<number>`.
First layer: each word is represented by a concatenation of the `GloVe` vector for that word and its `DepecheMood` values.
Convolutional layer: a 1D convolutional layer with filters of multiple sizes is applied to the word embeddings sequence. A global max-pooling is then applied across the sequence for each filter output.
Concatenation layer: applied to the output of the global max-pooling and the output of `VADER`.
Activation functions: *ReLU* is used between layers, except for the out layer where *tanh* is used to map output into [-1, 1] range.
Regularization: dropout is used to avoid over-fitting. The output of multiple networks with the same architecture but trained independently is averaged with different random seeds, to reduce noise.
Loss function: $\text{Loss} = \sum_{B \in Batches} 1 - \cos(\hat{\mathbf{V}}_B, \mathbf{V}_B)$, where $\hat{\mathbf{V}}_B$ and $\mathbf{V}_B$ are the predicted and true sentiment scores for batch B.

Results

Cross-validation results (training data)

Algorithm	*cos* (mean±std)
Full	0.701 ±0.023
No embeddings	0.586 ±0.017
No pre-processing	0.648 ±0.022

Final results

Algorithm	*cos*
Full	0.745
No embeddings	0.660
No pre-processing	0.678

Conclusions

The tables above shows the results for three different configurations: *i)* the full system, *ii)* the system without using word embeddings (i.e. `GloVe` and `DepecheMood`) *iii)* and the system without using pre-processing. The measure reported is cosine similarity, i.e. the official evaluation metric of SemEval-2017 Task 5 challenge. The first table shows the model's performances on the challenge training data, in a 5-fold cross-validation setting, while the second table contains the results on the testing data.
In both scenarios we can see that the use of pre-computed word representations helps avoiding over-fitting and achieving significantly better generalization, while some basic pre-processing can further improve the performance.

References

- `DepecheMood`: J. Staiano and M. Guerini. 2014. Depeche Mood: a Lexicon for Emotion Analysis from Crowd Annotated News. In *Proceedings of ACL 2014*.
- `GLOVE`: J. Pennington, R. Socher and C. Manning. 2014. GloVe: Global Vectors for Word Representation. In *Proceedings of EMNLP 2014*.
- `VADER`: C.J. Hutto and E. Gilbert. 2014. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Proceedings of ICWSM 2014*.