Documentation for the Data

Data description

The data is provided in `train.zip`, which contains two parts:

- `train.json`: metadata for each claim in the training set
- `train_articles`: folder of related articles for all the claims in the training set

In `train.json`, the following fields are provided for each claim (NB: some fields might be empty for some claims):

1) “claim”: statement

   Example: "claim": "Viral posts claim that climate change is a \"made-up catastrophe.\""

2) “claimant”: entity who made the claim

   Example: "claimant": "Social media posts"

3) “date”: when the claim was made

   Example: "date": "2019-05-08"

4) “label”: truth labeling of the claim (0:false, 1:partly true, 2:true)

   Example: "label": 0

5) “related_articles”: list of source and supporting article ids (that point to files in `train_articles`). Source articles denote articles that contain the claim, although the claim may be phrased differently in the article. Supporting articles denote articles that provide evidence that support the rating of the claim. Source and supporting articles are not distinguished in the list.

   Example: "related_articles": [20540, 38088, 14596, 20539, 20546, 38393, 20541, 41754]

6) “id”: unique identifier for each claim

   Example: “id”: 5

The folder `train_articles` contains text files where the name of each text file is an id to associate the article with a specific claim in `train.json`.

Example: 20540.txt

Data Origin

The dataset was obtained by downloading claims and associated metadata from 9 fact checking websites: politifact.com, snopes.com, washingtonpost.com, weeklystandard.com, africacheck.org, factscan.ca, factcheck.afp.com, polygraph.info, factcheck.org. On those websites, professional fact checkers publish a truth rating for each claim with a review article that explains the rating and links to related articles. The truth ratings provided by each fact checking website were mapped to the labels 0 (false), 1 (partly true) and 2 (true) as follows:

```
POLITIFACT_LABEL_MAP
   'Pants on Fire!': 0,
   'False': 0,
   'Mostly False': 1,
   'Half-True' : 1,
   'Mostly True': 1,
   'True': 2
```

```
WEEKLY_STANDARD_LABEL_MAP
   'FALSE': 0,
   'MIXED': 1,
   'TRUE': 2
```

```
FACTSCAN_LABEL_MAP
   'farcical': 0,
   'false': 0,
   'misleading': 1,
   'true': 2,
```

```
AFP_LABEL_MAP
   'faux': 0,
   'falso': 0,
   'false': 0,
   'fake': 0,
   'misrepresentation': 1,
   'misleading': 1,
   'mixed': 1,
   'vrai': 2,
```

```
WASHINGTON_POST_LABEL_MAP
   'four pinocchios': 0,
   'false': 0,
   'not true': 0,
   'wrong': 0,
   'fake news': 0,
   'mostly false': 1,
   'spinning the facts': 1,
   'not exactly': 1,
   'three pinocchios': 1,
   'two pinocchios': 1,
   'one pinocchio': 1,
   'not the whole story': 1,
   'needs context': 1,
   'lacks context': 1,
   'mis- leading': 1,
   'spins the facts': 1,
   'not quite right': 1,
   'twists the facts': 1,
   'disputed data': 1,
   'not quite': 1,
   'correct': 2,
   'geppetto checkmark': 2,
   'true': 2,

SNOPES_LABEL_MAP
   'false': 0,
   'fiction': 0,
   'legend': 0,
   'scam': 0,
   'probably false': 0,
   'mostly false': 1,
   'mixture': 1,
   'mixture of true and false information': 1,
   'partly true': 1,
   'mostly true': 1,
   'true': 2,

AFRICA_LABEL_MAP
   'false': 0,
   'incorrect': 0,
   'hoax': 0,
   'fake': 0,
   'false headline': 0,
   'mixture': 1,
   'understated/exaggerated': 1,
   'misleading': 1,
   'mostly correct': 1,
   'correct': 2,
   'checked': 2,
   'true': 2,

POLYGRAPH_LABEL_MAP
   'false': 0,
   'misleading': 1,
   'unclear': 1,
   'partly false': 1,
   'likely false': 1,
   'mostly false': 1,
   'partially false': 1,
   'partially false and misleading': 1,
   'unclear but likely partially true': 1,
   'unclear and partially true': 1,
   'questionable': 1,
   'partially true': 1,
   'likely true': 1,
   'true': 2,
```

Although those truth ratings are debatable, we treat them as the ground truth since a professional fact checker did some research and published a review article with links to related articles that support each rating. Furthermore, this fact checking process has withstood the test of time.

With each claim, we downloaded the related articles that were linked to by the fact checking website whenever possible. Since it is difficult to rate a claim just based on the metadata, we made sure that each claim has at least two related articles. Ideally, the related articles contain sufficient information to determine the truth of each claim, but this may not always be the case. Furthermore, there is no guarantee that the related articles provide relevant information. To reduce noise, we removed all related articles that had both zero cosine similarity and low topic overlap (by latent Dirichlet allocation) with the claim.

Scoring Formula

In phase 1, the goal is to predict the truth ratings that human fact checkers would assign to each claim based on some related articles and the metadata. The score of each submission will be the macro average F1 score of the claim ratings according to the following formula:

$$score1 = 2 * P * R \, / \, (P + R)$$

where:

$P = (P_{true} + P_{part} + P_{false})/3$ (macro-average precision)

$R = (R_{true} + R_{part} + R_{false})/3$ (macro-average recall)

$P_{class} = TP_{class}/(TP_{class} + FP_{class})$ (precision for a class)

$R_{class} = TP_{class}/(TP_{class} + FN_{class})$ (recall for a class)

$TP_{class} = \sum_{n=1}^{N} \delta(rating_n = class \wedge label_n = class)$ (true positives)

$FP_{label} = \sum_{n=1}^{N} \delta(rating_n = class \wedge label_n \neq class)$ (false positives)

$FN_{class} = \sum_{n=1}^{N} \delta(rating_n \neq class \wedge label_n = class)$ (false negatives)

N : # of claims and n : index of a claim

$class \in \{true, part, false\}$

$rating_n \in \{true, part, false\}$: rating produced by the code submitted for the n^{th} claim

$label_n \in \{true, part, false\}$: ground truth label for the n^{th} claim

$\delta(x) = 1$ (when $x = true$) and 0 (when $x = false$)

Baselines

We trained the following baseline classifiers on the training dataset and obtained the following F1 scores on the initial validation dataset used for ranking submissions on the leaders board.

- **Naive Bayes: 38.2%**. The related articles and the metadata are concatenated in one sequence of tokens. Then a Naive Bayes model is trained to maximize the joint likelihood of the token sequence and the label.
- **Bi-LSTM with attention: 49.4%**. A Bi-LSTM model with word-level attention [2] is trained on each (claim,article) pair. Then the logits (Bi-LSTM outputs) for n (# of related articles for a given claim) such pairs are averaged and passed through a softmax to predict the probability of each label.
- **BERT: 51.1%**. The claim and the related articles are preprocessed by converting each sentence into a TF-IDF representation. The 5 sentences that have the highest cosine similarity with a claim are extracted and concatenated with the metadata. Then,

Bi-directional Encoder Representations from Transformers (BERT) [1] is fine-tuned based on these sentences and the metadata to predict the label of each claim.

References

1. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.
2. Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., & Hovy, E. (2016, June). Hierarchical attention networks for document classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies* (pp. 1480-1489).