Persian Stance Classification Dataset

Majid Zarharan, Samane Ahangar, Fatemeh Sadat Rezvaninejad, Mahdi Lotfi Bidhendi
Shaghayegh Sadat Jalali, Sauleh Eetemadi, Mohammad Taher Pilehvar, Behrouz Minaei-Bidgoli
Iran University of Science and Technology
{majid_zarharan,ahangar_s, rezvaninejad_f
mahdi_lotfi, Shaghayegh_Jalali}@comp.iust.ac.ir
{sauleh, pilehvar, b_minaei}@iust.ac.ir

Abstract

We present the first stance detection dataset in Persian which has applications in fact-checking and summarization (Ferreira and Vlachos, 2016). We developed a web-based tool for importing rumored claims, collecting associated news-articles and labeling their stance against the claims. We used this tool to label 2,124 news articles against 534 rumored claims. We provide a number of baseline classification methods based on Ferreira and Vlachos (2016). In addition we introduce language specific features that outperform all baseline systems on this dataset.

1 Introduction

Social Media plays an important role in the society and it is rich in rumors and fake news. Fake news and false claims can have disastrous consequences. In Oct of 2018, The New York Times reported that genocide in Myanmar was incited by campaign of fake news on Facebook (Mozur, 2018).

Fake news spreads more promptly than the truth (Dizikes, 2018) and their credibility cannot be verified manually as it is time-consuming (Popat et al., 2018). Vosoughi et al. (2018) indicate that fake news was 70% more likely to be spread in Twitter than the truth. Thus, we need to have a tool for automatic detection and verification of claims.

It is complex to detect fake news, even for trained experts. But we can divide the process into several smaller steps. Stance classification is the first step in this process (Pomerleau and Rao., 2017). Therefore, in this paper, we focus on the stance detection task and developing a dataset for it.

There is no dataset for stance classification and fake news detection in Persian. Some related works in English like the works of Qazvinian et al. (2011); Lukasik et al. (2015); Zhao et al. (2015), focus on stance detection for Twitter data. Thorne et al. (2018) provide a fact extraction dataset that uses facts extracted from Wikipedia to generate factual and false claims. In 'Liar, Liar Pants on Fire', Wang (2017) provides a dataset extracted from `PolitiFact`[1]. The aforementioned works are all in English.

In absence of any fake news dataset for Persian, we collect claims from `Fakenews`[2] and `Shayeaat`[3] websites. Then we look for articles related to claims. After collecting articles, for each claim we allocate three labels; first label is article (body text) stance according to the claim (article-claim stance), second label is article's headline stance according to the claim (headline-claim stance) and the third one is article (body text) stance according to its headline (article-headline stance). At the end, we assess the veracity of each claim with respect to its related articles. Our dataset is useful for stance detection, fact-checking and summarization. (Ferreira and Vlachos, 2016).

In this paper, we use our dataset to investigate the task of classifying article-claim stance and headline-claim stance. In particular, for each article headline and article body text we assign a stance label which is one of `agree`, `disagree`, `discuss` or `unrelated`, indicating whether the article is supporting, rejecting, just reporting the claim or it is unrelated to the claim, respectively.

2 Stance Classification

Automating the process of understanding what other news organizations are saying about the claim is called "stance detection" (Pomerleau and

[1] PolitiFact.com
[2] Fakenews.ir
[3] Shayeaat.ir

Rao., 2017). In order to do stance detection, we provide a textual claim (input A) and an article's body text (input B) as inputs to the stance detection system which outputs the stance of the article with respect to the claim:

- **Agree**: The article states that the claim is true, without any kind of hedging and quotation.
- **Disagree**: The article states that the claim is false, without any kind of hedging and quotation.
- **Discuss**: The claim is reported in the article, but without evaluating its truth.
- **Unrelated**: The claim is not reported in the article.

We provide an example of headline-claim stance from our dataset in Figure 1. The full text of the articles is omitted for brevity.

Claim:
Kamal Kharrazi meeting with John Kerry in Paris
ملاقات کمال خرازی با جان کری در پاریس

Headline:
Kamal Kharrazi encounters with John Kerry in Paris
دیدار کمال خرازی و جان کری در پاریس
Stance: Agree

Headline:
The visit of Kamal Kharrazi to John Kerry was denied
تکذیب خبر دیدار کمال خرازی با جان کری
Stance: Discuss

Headline:
The news of Kharrazi meeting with John Kerry is a big lie
خبر دیدار خرازی با کری دروغ محض است
Stance: Disagree

Headline:
Kamal Kharrazi said that Iran seeks peace and stability in the region
کمال خرازی عنوان کرد که ایران به دنبال صلح و آرامش در منطقه است
Stance: Unrelated

Veracity: False

Figure 1: Example of stance classification.

3 Methodology

The dataset was constructed in the following stages:

Source	Claim Count	Headline as Claim	Modified Headline as Claim	Headline as Claim Percentage
Shayeaat	513	513	0	100%
Fakenews	87	37	50	42.52%

Table 1: The distribution of unchanged vs updated claims in the dataset.

- **Claim collection**: Collect Persian rumors and fake news from `Fakenews` and `Shayeaat`.
- **Labeling**: For each claim we allocated three stance labels. At the end, we assess veracity of each claim.

The web based tool developed for data collection and labeling is available on github[4]. This tool was used to collect claims and articles. Also, we used this tool to label the articles and to estimate the veracity of each claim.

3.1 Claim collection

The target of this task is to collect and generate claims from rumors and news headlines extracted from `Fakenews` and `Shayeaat`. These two websites collect rumors from the community and social networks and then evaluate their veracity with evidence manually.

After extracting rumors from `Shayeaat`, we use each rumor as a claim without any changing. For `Fakenews` we extracted news headlines and used 42% of them as claims without change. The remaining 58% of the headlines were manually changed by super-annotators[5] to be appropriate as a claim. For example a super-annotator removed the rumor word from some news headlines or completed some others with a verb. Table 1 shows the number of claims we collected in our dataset and the distribution of unchanged vs updated claims.

3.2 Labeling

Annotators first examined each claim. If the stance of a claim could not be verified by the textual content of any articles (e.g., it was verified by image or video) the claim was labeled as `useless` and was not used in training.

[4]https://github.com/majidzarharan/persian-stance-classification

[5]Expert annotators and the first two authors of this article: They created the guideline for stance labels. There is no difference between the value of labels provided by super-annotators and regular annotators.

After looking for every claim on the web, annotators find articles related to the claim and save the headline, body text and URL of the articles. After gathering related articles for each claim, the annotators were asked to find and save at least one article unrelated to the claim. They allocate three stance labels, two stance labels for each claim and one for each article. The first label is article-claim stance, the second label is headline-claim stance and the third one is article-headline stance. The label of article-claim stance is different from the label of headline-claim stance, because the article's headlines are more concise compared to the article's body text. Therefor, the article's headlines contain fewer hedging and quotations. Stance labels consist of `agree`, `disagree`, `discuss` and `unrelated` where we followed Ferreira and Vlachos (2016) and added `unrelated` as an additional label to them like the work of Pomerleau and Rao. (2017).

In addition to stance labeling, we also assess veracity of each claim with respect to its related articles. We assign a label to each claim with an estimation of its veracity with `true`, `false` and `unknown`, where `unknown` indicates we were unable to verify whether the claim was true or not.

3.3 Annotators

The annotation team had 7 members, 5 of which are authors of this paper. 2 of the 7 annotators were super-annotators. All annotators are native Persian speakers and were trained directly by the first author. We prepared a guideline in both English and Persian language, which consists of notes, suggestions and examples about stance labels. The guidelines are provided as supplementary material to this paper. The guideline is also available on github[6].

3.4 Data Validation

Due to the sensitivity of the subject (stance labeling), we used three forms of data validation: Overlap, Agreement against super-annotators and Majority voting. The validation of claims was done during claim labeling. As a result of claim validation, we collect 600 claims, 11% of which were labeled as `useless` and skipped, leaving 534 claims.

[6]https://github.com/majidzarharan/persian-stance-classification

3.4.1 Overlap

For all claims, we created an overlap so that each claim is labeled by two different people. All claims were labeled by at least one of the super-annotators.

3.4.2 Agreement against super-annotators

After labeling the claims, we inspected labels from annotators and super-annotators for agreement. The first row in Table 2 indicates label agreement percentage between annotators and super-annotators. In case of disagreement, the annotators and super-annotators were asked to review labels. After reviewing, if they discerned anything wrong, they were asked to correct labels. The second row in Table 2 indicates label agreement percentage between all annotators after reviewing the label of articles and claims.

3.4.3 Majority voting

After collecting label agreement data, the remaining claims that differ between two annotators are considered by a third annotator. After assigning the third label by the next person, we use majority vote if two annotators voted the same. If all three annotators voted differently, we do not use that claim or article. The third row in Table 2 indicates label agreement percentages between annotators after majority voting. We discarded 1.95% instances of the dataset for headline-claim, 2.5% instances for article-claim and 1.51% instances for article-headline after majority voting.

3.5 Data collection result

After skipping `useless` claims, our dataset consists of 534 claims and 2,124 associated news articles with an average ratio of 3.98 articles per claim. The minimum number of articles per claim is 1 and the maximum number is 10.

Distribution of stance classes is illustrated in Table 3. The first row indicates class distribution of article-claim stance. The second row indicates class distribution of headline-claim stance. The last row indicates class distribution of article-headline stance. As expected headlines are more likely to state the claim (`agree`) and there are few articles that are not relevant to their headline. In all three rows, most articles discuss the claim because quotation phrases are used repeatedly in most Persian articles.

Although this dataset can be used for fact-checking and summarization (Ferreira and Vla-

Agreement time	Headline-claim	Article-claim	Article-Headline	Claims veracity
Before adjudication	82.15%	80.01%	81.49%	92.25%
After rechecking	93.21%	92.01%	92.49%	96.25%
After majority vote	98.05%	97.50%	98.49%	99.60%

Table 2: Label agreement percentage at different stages of data validation.

Type	Agree	Discuss	Disagree	Unrelated
article-claim stance	7.43%	54.85%	11.16%	28.53%
headline-claim stance	20.17%	39.75%	8.08%	31.98%
article-headline stance	29.24%	63.51%	6.56%	0.64%

Table 3: Class distribution of article-claim stance, headline-claim stance and article-headline stance.

chos, 2016) but the focus of this work is on stance detection. In the next section, we report experiment results on article-claim stance detection and headline-claim stance detection.

4 Experiments

For pre-processing the data, we removed characters which existed due to data gathered from the web. Also, some of the claims included the word "rumor" which is not part of a claim. So, we deleted these phrases. Data normalization on Persian language has also been done with StanfordNLP (Qi et al., 2018).

4.1 Features

We used Bag-of-words representation (BoW) and TF-IDF[7] in order to extract features from our text. In addition, we extracted two features from news headlines and claims. The first is whether news headline or claim ends in a question mark (IsAQuestion) and the second is whether the sentence is more than one part or not (HasMoreThanOnePart).

We use "RootDist" feature-set from Ferreira and Vlachos (2016). For creating this feature, we collected the refuting, hedging and reporting words in the Persian language and computed minimum distance between those words and root of the sentences. In order to find the root of sentences we used StanfordNLP's dependency parser for Persian language (Qi et al., 2018).

The last feature we define is the similarity between two inputs of the stance detection system (textual claim and article's body text). In order to implement this feature, we used pre-trained vectors (Bojanowski et al., 2016) for Persian words. We computed the cosine similarity between vectors of claim and article's body text word by word.

4.2 Classification Methods

We compare several classifiers for stance classification according to the aforementioned features: majority baseline, logistic regression with L1 regularization (Pedregosa et al., 2011), support vector machine (SVM) (Crammer and Singer, 2001), random forest model(Breiman, 2001), and Naive Bayes (Zhang, 2004).

In addition to the baseline classification methods above we investigate top three submissions to the Fake News Challenge[8]. All three use a deep learning approach for stance detection. In pursuit of the highest performing deep learning architecture for this task we use the stackLSTM architecture proposed by Hanselowski et al. (2018) which outperforms all FNC1 submissions.

We experiment with several modifications and hyper-parameters for the stackLSTM architecture using Keras[9]. The final model that yields the highest overall accuracy uses pre-trained 300-dimensional word embedding from (Bojanowski et al., 2016). The overall architecture is shown in Figure 2. In the stackLSTM architecture (Hanselowski et al., 2018), a feature-based model is combined with a structure which can better represent meaning using word embedding and encoding. To take word-sequence into account, the first 100 features, which are word embedded features (V), are given to two LSTMs[10]. Then, the combination of last hidden state with size 100 and rest of the features is assigned to 3 dense neural net-

[7]Term Frequency-Inverse Document Frequency

[8]http://www.fakenewschallenge.org/

[9]https://keras.io

[10]Long Short-Term Memory (Hochreiter and Schmidhuber, 1997)

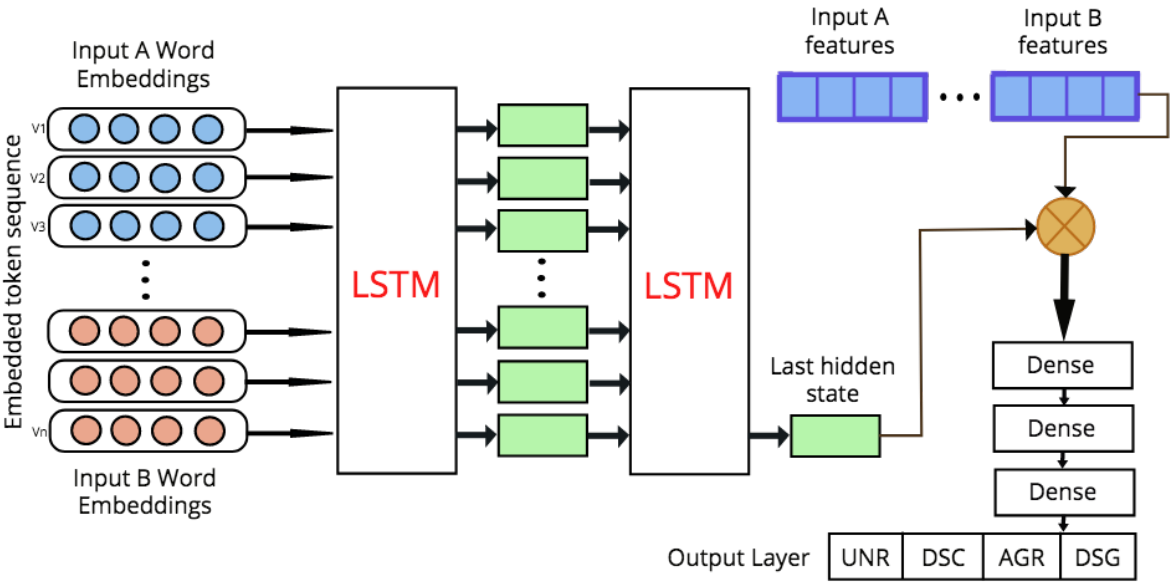

Figure 2: Model architecture of the stackLSTM.

	Features based on Bag-of-words				Features based on TF-IDF			
Models	**pre.**	**Recall**	**F1**	**acc.**	**pre.**	**Recall**	**F1**	**acc.**
Random Forest	0.70	0.69	0.67	**0.69**	0.69	0.68	0.68	**0.68**
Logistic Regression	0.63	0.63	0.63	0.63	0.63	0.64	0.63	0.64
SVM	0.64	0.63	0.63	0.63	0.64	0.64	0.64	0.64
Naive Bayes	0.50	0.50	0.50	0.50	0.60	0.49	0.49	0.49
Majority	0.15	0.39	0.22	0.39	0.15	0.39	0.22	0.39
stackLSTM	0.63	0.62	0.62	0.62	–	–	–	–

Table 4: Headline-claim stance classification. We extract features based on Bag-of-words and TF-IDF. For all of the models we present recall, precision, F1-score and accuracy.

	Features based on Bag-of-words				Features based on TF-IDF			
Models	**pre.**	**Recall**	**F1**	**acc.**	**pre.**	**Recall**	**F1**	**acc.**
SVM	0.55	0.56	0.55	0.565	0.59	0.61	0.58	**0.610**
Logistic Regression	0.57	0.59	0.57	0.592	0.59	0.60	0.54	0.597
Random Forest	0.52	0.57	0.49	0.575	0.58	0.60	0.55	0.605
Naive Bayes	0.50	0.58	0.52	0.580	0.57	0.57	0.51	0.575
Majority	0.27	0.52	0.36	0.522	0.27	0.52	0.36	0.522
stackLSTM	0.57	0.62	0.71	**0.72**	–	–	–	–

Table 5: Article-claim stance classification. We extract features based on Bag-of-words and TF-IDF. For all of the models we present recall, precision, F1-score and accuracy.

work layers with 300 neurons each. Finally, there is a dense layer with softmax activation function to specify the proper class.

We used these classification methods once for headline-claim stance classification (illustrated in Table 4) and once for article-claim stance classification (illustrated in Table 5). For headline-claim stance classification we used headline text for input A. The stackLSTM classification method performs best in article-claim stance with accuracy 72% but in headline-claim stance it does not work well and Random Forest performs best with overall accuracy 69%.

Model	F1m	FNC-1 Score
TalosComb	0.582	0.820
Athene	0.604	0.820
UCLMR	0.583	0.817
(Hanselowski et al., 2018)	**0.609**	**0.821**
Persian Stance Detection Model	0.583	0.747

Table 6: Comparison of our article-claim stance classification results (macro F1 and FNC-1 score) with other similar works in English.

	Count
All Claims	**600**
All Claims Without Useless	**534**
All Articles	**2124**
Articles Per Claim	**3.98**
Min Articles Per Claim	**1**
Max Articles Per Claim	**10**

Table 7: Summary of our persian stance classification dataset.

4.3 Comparison

There is no similar work in Persian stance detection. therefor, we compare our article-claim stance results with Hanselowski et al. (2018) and top three submissions to the Fake News Challenge. Table 6 shows this comparison. We implemented FNC-1 score as Fake News Challenge Defined it:

The FNC-1 score will be incremented by 0.25 if stance detection system detect an unrelated instance correctly.

The FNC-1 score will be incremented by 0.5 if stance detection system detect a related (agree, disagree or discuss) instance correctly, without respect to exact label.

The FNC-1 score will be incremented by 0.75 if stance detection system detect a related (agree, disagree or discuss) instance correctly and detect the exact label of agree, disagree or discuss.

5 Related Work

We collect real rumors from the websites like `Shayeaat` and `FakeNews`. These websites collect real rumors from anywhere such as social media, web logs and news outlets. Then we assessed veracity of each rumor. The major difference between our work and recent works like dataset of Wang (2017) is the dataset language.

Our work also differs from other works in stance classification like works of Qazvinian et al. (2011); Lukasik et al. (2015); Zhao et al. (2015) in sources where we gathered data. These works have limited their data sources to social networks such as Twitter, but we have used various web-based articles for each claim, most of them are from official news sources. Our work differs from Qazvinian et al. (2011) in the number of claims, too. they collected a dataset based on Twitter and manually annotated five rumors.

Our Persian fake news dataset is more realistic than datasets such as Thorne et al. (2018) since they extract information from Wikipedia and then fabricate true/false claims from the extracted information. However, in this work we collect claims from Persian rumor websites. They also use Wikipedia to classify whether a claim is supported or refuted, but we look for articles that mention our claims in the web and decide on the stance of each article according to its related claim.

We also investigated the Persian dataset provided by Derakhshi et al. (2019). However, this work is a collection of alleged rumors published on Telegram[11] without any labels. The work of Zamani et al. (2017) is notable for collecting and annotating a dataset from Twitter. However, although this work does contain some content features, but it relies heavily on Twitter specific features such as user profile information and response/retweet structure. Our approach differs significantly from this work, since we focus on stance detection and rely solely on content based features.

6 Conclusions - Future work

In this paper we introduced a Persian dataset which can be used for a number of NLP tasks in the context of fact-checking. Although this dataset can be used for fact-checking and summarization, the focus of this work is on stance classification as a stepping stone for fake news detection in Persian language. In addition to the dataset, our data collection tools are also available for other data collection efforts. Table 7 shows the summary of corpus statistics.

We implemented multiple classification methods for stance detection using Bag-of-words (BoW) and TF-IDF. We plan to train word embedding on Persian news and use it as feature in future works. In addition, we intend to use BERT (Devlin

[11] It is a social media. Telegram.org

et al., 2018) and other state-of-the-art deep neural networks to improve the accuracy of the stance classification task on this dataset. Finally, our goal is to use this stance classifier to build an end-to-end fake news detection pipeline.

7 Acknowledgments

The authors would like to thank Zahra Sayedi and Fatemeh Sadat Shahrabadi for their significant contribution in collection and labeling of the articles.

References

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2016. Enriching word vectors with subword information. *arXiv preprint arXiv:1607.04606*.

Leo Breiman. 2001. Random forests. *Machine Learning*, 45:5–32.

Koby Crammer and Yoram Singer. 2001. On the algorithmic implementation of multiclass kernel-based vector machines. *Journal of Machine Learning Research*, 2:265–292.

AliReza Feizi Derakhshi, MohammadReza Feizi Derakhshi, Mehrdad Ranjbar Khadivi, Narjes Nikzad Khasmakhi, Majid Ramezani, Taymaz Rahkar Farshi, Elnaz Zafarani Moattar, Meysam Asgari Chenaghlu, and Zoleikha Jahanbakhsh Nagadeh. 2019. Sepehr rumtel01.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *ArXiv*, abs/1810.04805.

Peter Dizikes. 2018. Study: On twitter, false news travels faster than true stories.

William Ferreira and Andreas Vlachos. 2016. Emergent: a novel data-set for stance classification. In *HLT-NAACL*.

Andreas Hanselowski, S. AvineshP.V., Benjamin Schiller, Felix Caspelherr, Debanjan Chaudhuri, Christian M. Meyer, and Iryna Gurevych. 2018. A retrospective analysis of the fake news challenge stance-detection task. In *COLING*.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9:1735–1780.

Michal Lukasik, Trevor Cohn, and Kalina Bontcheva. 2015. Classifying tweet level judgements of rumours in social media. In *EMNLP*.

Paul Mozur. 2018. A genocide incited on facebook, with posts from myanmars military.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jacob VanderPlas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Edouard Duchesnay. 2011. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12:2825–2830.

Dean Pomerleau and Delip Rao. 2017. The fake news challenge: Exploring how artificial intelligence technologies could be leveraged to combat fake news.

Kashyap Popat, Subhabrata Mukherjee, Andrew Yates, and Gerhard Weikum. 2018. Declare: Debunking fake news and false claims using evidence-aware deep learning. In *EMNLP*.

Vahed Qazvinian, Emily Rosengren, Dragomir R. Radev, and Qiaozhu Mei. 2011. Rumor has it: Identifying misinformation in microblogs. In *EMNLP*.

Peng Qi, Timothy Dozat, Yuhao Zhang, and Christopher D. Manning. 2018. Universal dependency parsing from scratch. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 160–170, Brussels, Belgium. Association for Computational Linguistics.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. Fever: a large-scale dataset for fact extraction and verification. In *NAACL-HLT*.

Soroush Vosoughi, Brandon C. Roy, and Sinan Aral. 2018. The spread of true and false news online. *Science*, 359:1146–1151.

William Yang Wang. 2017. "liar, liar pants on fire": A new benchmark dataset for fake news detection. In *ACL*.

Somayeh Zamani, Masoud Asadpour, and Dara Moazzami. 2017. Rumor detection for persian tweets. *2017 Iranian Conference on Electrical Engineering (ICEE)*, pages 1532–1536.

Harry Zhang. 2004. The optimality of naive bayes. In *FLAIRS Conference*.

Zhe Zhao, Paul Resnick, and Qiaozhu Mei. 2015. Enquiring minds: Early detection of rumors in social media from enquiry posts. In *WWW*.