



## تشخیص موضع در زبان فارسی

مجید زهرن  
دانشکده مهندسی کامپیوتر  
دانشگاه علم و صنعت ایران  
majid\_zarharan@comp.iust.ac.ir

سمانه آهانگر  
دانشکده مهندسی کامپیوتر  
دانشگاه علم و صنعت ایران  
ahangar\_s@comp.iust.ac.ir

### چکیده

تشخیص موضع یک متن نسبت به یک ادعا، اولین و مهم ترین قدم در تشخیص اخبار جعلی است [۱]. همچنین تشخیص موضع در قسمت عمده‌ای از چک کردن حقیقت سهیم می باشد. در زبان انگلیسی مدل های بسیاری برای انجام این کار معرفی شده اند. یکی از بهترین مدل ها stackLSTM [۲] میباشد که توانسته است به دقتی بالاتر از دقت سه مدل برنده در چالش اخبار جعلی دست پیدا کند. در زبان فارسی تاکنون تشخیص موضع پیاده سازی نشده است و نمونه بومی وجود ندارد. ما در زبان فارسی با الهام از این مدل آزمایشات خود را انجام داده ایم. بعد از تغییر پارامترها و تنظیمات مدل و همینطور آزمایش ویژگی های مختلف توانستیم به دقت ۷۲٪ دست پیدا کنیم.

### ۱ مقدمه

رسانه های اجتماعی نقش مهمی را در جامعه ایفا میکنند و مملوء از شایعات و اخبار جعلی هستند. اخبار و ادعاهای دروغین میتواند پیامدهای فاجعه آمیز داشته باشد. این اخبار سریعتر از حقیقت گسترش می یابند [۳] و نمیتوان اعتبار آنها را به صورت دستی بررسی کرد زیرا بسیار زمان بر است [۴]. وثوقی و همکارانش [۵] نشان دادند که اخبار جعلی ۷۰ درصد بیشتر از حقیقت در توییتر پخش می شوند. بنابراین ما باید یک ابزار برای شناسایی خودکار و تایید ادعاها داشته باشیم.

فرآیند تشخیص جعلی بودن خبر حتی برای کارشناسان آموزش دیده نیز دشوار است. با این حال ما میتوانیم این فرآیند را به چند مرحله کوچکتر تقسیم کنیم. تشخیص موضع اولین گام در این فرآیند است. [۱]

به فرآیند خودکار فهمیدن اینکه سازمان های خبری درباره ی یک ادعا چه میگویند، تشخیص موضع گفته میشود [۱]. برای این منظور ما یک ادعا(عنوان خبر) و یک متن خبری را بعنوان ورودی به سیستم تشخیص موضع میدهم تا موضع متن خبر را نسبت به ادعا مشخص کند.

این موضع میتواند یکی از چهار حالت زیر باشد:

- Agree : متن خبر ادعا را بدون هیچگونه ابهام یا نقل قول تایید کند.
- Disagree : متن خبر ادعا را بدون هیچگونه ابهام یا نقل قول رد کند.
- Discuss: ادعا در متن خبر گزارش شده است بدون اینکه نویسنده خبر به طور قاطع و صریح آن را تأیید یا رد کرده باشد.
- Unrelated : متن و ادعا ربطی به هم نداشته باشند.

نمونه ای از دیتا در شکل ۱ نشان داده شده است.

<b>Claim:</b> Kamal Kharrazi meeting with John Kerry in Paris <small>ملاقات کامال خرازی با جان کری در پاریس</small>
<b>Headline:</b> Kamal Kharrazi encounters with John Kerry in Paris <small>دیدار کامال خرازی و جان کری در پاریس</small>
<b>Stance:</b> Agree
<b>Headline:</b> The visit of Kamal Kharrazi to John Kerry was denied <small>تکذیب خبر دیدار کامال خرازی با جان کری</small>
<b>Stance:</b> Discuss
<b>Headline:</b> The news of Kharrazi meeting with John Kerry is a big lie <small>خبر دیدار خرازی با کری دروغ محض است</small>
<b>Stance:</b> Disagree
<b>Headline:</b> Kamal Kharrazi said that Iran seeks peace and stability in the region <small>کمال خرازی عنوان کرد که ایران به دنبال صلح و آرامش در منطقه است</small>
<b>Stance:</b> Unrelated
<b>Veracity:</b> False

شکل ۱: نمونه ای از داده

## ۲ کارهای مرتبط / پیش‌زمینه

در سال‌های اخیر کارهای بسیاری در این زمینه بر روی دادگان انگلیسی انجام شده است.

چن و همکاران [۶] پیشنهاد دادند که یک ادعا را از دیدگاه‌های متنوع و جامع بررسی کنیم. آنها معتقد بودند که دیدگاه‌های مختلف ممکن است شدت تایید یا رد متفاوتی داشته باشند. آنها برای هر ادعا ابتدا تمامی متن‌های مرتبط را بدست آوردند. سپس هر متن را به قسمت‌های مهمش که در پیش‌بینی تاثیرگذار بود، خلاصه کردند. بعد از آن متن‌ها را از نظر تایید یا رد ادعا درجه‌بندی کردند.

ما و همکاران [۷] تشخیص شایعه و موضع را به عنوان یک کار مشترک در نظر گرفتند. آنها ثابت کردند که بین صحت ادعا و موضع بیان شده یک ارتباط قوی وجود دارد. آنها در شبکه‌ی عمیق استفاده شده، ویژگی‌های مشترک دو وظیفه را بصورت وزن‌های مشترک در نظر گرفتند. این در حالی است که هر کدام از کارها میتواند ویژگی‌های خاص خودش را نیز داشته باشد.

محترمی و همکاران [۸] مدلی را ارائه دادند که علاوه بر پیش‌بینی موضع خبر نسبت به ادعا، آن قسمتی از متن را که نتیجه‌ی این پیش‌بینی را رقم‌میزند استخراج میکند. مدل آنها در سطح پاراگراف عمل میکند و هر کدام را جداگانه مورد بررسی قرار میدهد.

سدیک و همکارانش [۹] عنوان کردند که در زبان طبیعی انسان‌ها، معانی پنهانی وجود دارد. آنها روشی را ارائه دادند که بتوان متن‌های خبری پیچیده را نیز مورد بررسی قرار داد مانند طنز، طعنه و یا محتوای گمراه‌کننده.

ژانگ و همکاران [۱۰] ادعا کردند که ممکن است موضع یک متن نسبت به یک ادعا، دقیقاً در یکی از چهار کلاس موافق، مخالف، بحث‌شده و نامرتب قرار نگیرد. آنها احتمال مواضع را برای هر متن و ادعا در نظر گرفتند و بر اساس این احتمال، مواضع را رتبه‌بندی کردند. در مقاله‌ی دیگر او و همکارانش [۱۱] یک مدل سلسله‌مراتبی را برای بهبود پیش‌بینی کلاس اقلیت استفاده کردند. به این صورت که در یک مرحله ابتدا موضع مرتبط یا غیر مرتبط بودن متن به ادعا را تشخیص میدهند، و در مرحله‌ی بعد، اگر موضع مرتبط بود، نوع مرتبط بودن را بررسی میکنند. نوع مرتبط بودن یعنی موافق، مخالف، یا بحث‌شده.

### ۳ مدل پیشنهاد شده

#### ۱.۳ پیش پردازش

ما برای پیش پردازش دادگان تنها علائمی که به دلیل استخراج داده ها از وب وجود داشتند را حذف کردیم. همچنین بعضی از ادعاها، عبارت “ شایعه” را کنار خود داشتند که جزیی از ادعا محسوب نمیشدند که این عبارات را حذف کردیم. نرمال سازی دادگان نیز با stanfordnlp [۱۲] برای زبان فارسی انجام شده است. کلمات ایست ، نقاط و علائم دیگر را به دلیل ویژگی هایی که به آنها نیاز داشتند نگه داشتیم و در بعضی ویژگی ها حذف کردیم.

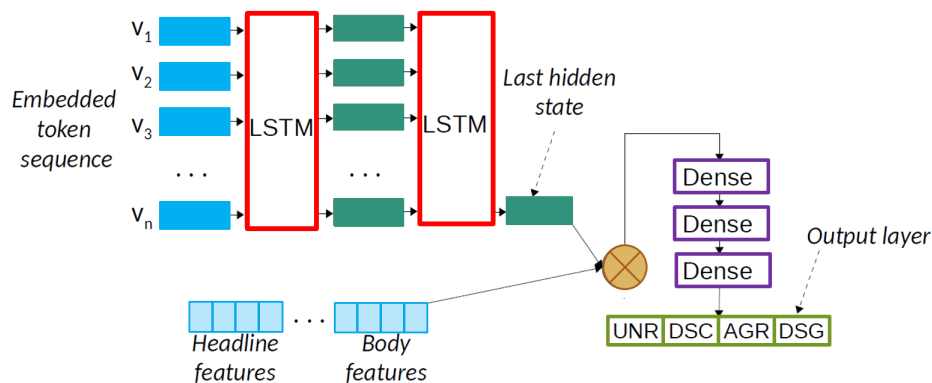
#### ۲.۳ ویژگی ها

ویژگی هایی که ما استفاده کردیم در سه دسته ی زیر قرار میگردند:

- BOW : شامل ویژگی هایی میشود از قبیل تعداد دفعاتی که یک توکن در ادعا، در متن هم ظاهر میشود. این تعداد بدون در نظر گرفتن کلمات ایست هم محاسبه میشود. تعداد دفعاتی که یک توالی از توکن ها ( با سایز های ۲ تا ۶ ) یا کاراکتر ها ( با سایزهای ۲، ۴، ۸ و ۱۶) در ادعا، در متن هم ظاهر میشوند. Tf-idf برای توالی سه تایی توکن ها.
- Topic Modeling : شامل ویژگی هایی میشود از قبیل فاصله کسینوسی بین بردارهای موضوعی ادعا و متن خبر. رابطه پنهان معنایی بین متن و ادعا. فاکتورگیری ۳۰۰ موضوع مهم. فاصله کسینوسی بین ۳۰۰ موضوع مهم عنوان و متن خبر.
- Word embedding : بردار کلمات از پیش آموزش داده شده ی زبان فارسی [۱۳]

### ۳.۳ مدل

در شکل زیر مدل stackLSTM [۲] نشان داده شده است. مدلی که صرفا بر اساس ویژگی لغوی باشد فاقد درک معنایی است. در این جا یک مدل مبتنی بر ویژگی با یک مدلی که بهتر بتواند معنا را با استفاده از embedding word و انکودینگ تشخیص دهد ترکیب شده است. پردازش متوالی اطلاعات برای فهمیدن معنی جمله مهم است. از این رو ۱۰۰ ویژگی اول که ویژگی های embedded word ها (v) هستند، به دو LSTM stack و یک وضعیت مخفی با سایز ۱۰۰ و یک دراپ اوت ۲۰ داده میشود. وضعیت مخفی با بقیه ی ویژگی ها ترکیب میشود و به ۳ لایه شبکه عصبی dense با ۶۰۰ نورون در هر یک داده میشود. در پایان یک لایه دنس با ۴ نورون و تابع فعال ساز softmax قرار دارد تا احتمال ۴ کلاس را برگرداند.

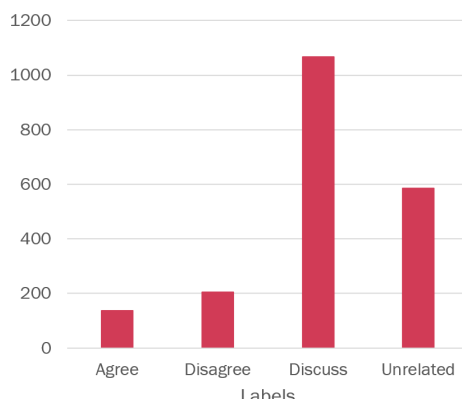


شکل ۲: مدل stackLSTM [۲]

## ۴ نتایج

### ۱.۴ دادگان

داده‌ی جمع‌آوری شده شامل ۱۹۹۷ نمونه می‌باشد که هر نمونه شامل متن خبر، ادعا، عنوان و موضع (کلاس) متن خبر نسبت به عنوانش می‌باشد [۱۴]. در این پیاده‌سازی از ادعا و متن خبر استفاده شده است. توزیع داده‌ها در کلاس‌ها مطابق شکل ۳ است.



شکل ۳: توزیع کلاس‌ها

### ۲.۴ بیس لاین‌ها

ما از ۵ مدل زیر بعنوان بیس لاین استفاده کرده‌ایم:

- Majority
- logistic regression classifier with L1 regularization [15]
- support vector machine classifier (SVM) [16]
- random forest model [17]
- Naive Bayes model [18]

### ۳.۴ مدل‌ها و تنظیمات

ما با ترکیب‌های مختلف ویژگی‌ها و مقادیر مختلف پارامترها آزمایشات گوناگونی انجام دادیم.

- Dropout {0.2, 0.5, 0.8} برای وزنه‌های درونی و بیرونی LSTM ها و همینطور بین لایه‌های dense
- لایه‌های dense با تعداد کمتر و آزمایش مقادیر نوروں‌ها بین ۱۰۰ تا ۷۰۰.
- تغییر تعداد LSTM ها و تعداد یونیت‌های آنان با توجه به ورودی.
- تعداد توکن‌هایی که بعد از قرار گرفتن عنوان و متن در کنار هم و تولید بردار embedding آنها بعنوان ورودی LSTM استفاده شد. {50, 100, 150, 200}
- Optimizer نهایی rms انتخاب شد با نرخ یادگیری 0.01 و همینطور از cross entropy بعنوان categorical تابع loss استفاده شد.
- لایه‌ی آخر شامل ۴ نوروں و تابع فعال ساز softmax برای کلاس بندی می‌باشد.

چهار تا از مدل‌هایی که بهترین نتایج را گرفتند به شرح زیر است:

- stackLSTM\_org : برای این مدل از تمامی ویژگی‌های استخراج شده استفاده کردیم. embedding Word
- برای ۱۰۰ توکن اول متن و ادعا بعنوان ورودی به stackLSTM داده شد. هر کدام از بلاک‌های LSTM ۱۰۰ یونیت دارند به همراه ۲. dropout. لایه‌های dense شامل ۶۰۰ نوروں به همراه تابع فعال سازی ReLu می‌باشند.

- stackLSTM : فقط از دو سری ویژگی BOW و embedding word برای این مدل استفاده شده. بعد از حذف ویژگی های modeling Topic تعداد ویژگی ها نصف حالت کلی شد.
- LSTM : برای ساده تر کردن مدل تصمیم گرفتیم از یک lstm به جای دو تا استفاده کنیم. این lstm هم شامل ۱۰۰ یونیت و dropout ۰.۲ می باشد.
- stackLSTM\_reg : ۳ لایه dense به همراه ۳۰۰ نورون و dropout ۰.۵. بین آنها. بر روی دیتاست فارسی این مدل بهترین نتیجه را بدست آورد.

#### ۴.۴ معیار ارزیابی

ما از چهار معیار زیر برای ارزیابی استفاده کرده ایم:

- Accuracy : تعداد نمونه هایی که درست پیش بینی شده اند نسبت به تعداد کل نمونه ها.
- Macro-F1 : برای هر کلاس F1 محاسبه شده و سپس میانگین آنها گرفته میشود. (F1m)
- Weighted-F1 (F1) : برای هر کلاس F1 محاسبه میشود و سپس میانگین وزن دار آنها با توجه به وزن کلاس ها گرفته میشود.
- FNC-1 : برای مدل های جدید ما معیاری را که برگزار کنندگان چالش اخبار جعلی ارائه دادند را نیز محاسبه کردیم. این معیار ارزیابی به صورت سلسله مراتبی است. ابتدا اگر متن به درستی مربوط یا نامربوط بودنش تشخیص داده شود ۰.۲۵ امتیاز میگیرد. سپس اگر بین کلاس های مرتبط، به درستی پیش بینی شود که موافق، مخالف یا در حال بحث است، ۰.۷۵ امتیاز دیگر میگیرد.

#### ۵.۴ نتایج آزمایشات

جدول ۱ کارایی تمام مدل ها را روی داده ی تست گزارش میدهد. در این جدول ما مقادیر Accuracy، F1 و توانایی مدل برای پیش بینی هر کلاس را نشان داده ایم.

Method	Acc	F1	AGR	DSG	DSC	UNR
Majority	.522	.36	0	0	1	0
SVM	.61	.58	.834	.405	.361	.04
Logistic_Reg	.597	.54	.855	.369	.128	0
Random Forest	.605	.55	<b>.921</b>	.243	.298	.04
Naïve Bayes	.51	.575	.866	.369	.021	0
stackLSTM_org	.665	.628	.074	.514	.924	.406
stackLSTM	.63	.57	.185	.486	<b>.942</b>	.25
LSTM	.63	.65	.444	.486	.695	.601
stackLSTM_reg	<b>.72</b>	<b>.71</b>	.185	<b>.571</b>	.79	<b>.758</b>

Table 1: Total accuracy, F1\_score and classes accuracy

همچنین مقادیر Macro F1 و FNC-1 برای مدل های پیشنهادی جدید در جدول ۲ نشان داده شده است.

Method	F1m	FNC-1
stackLSTM_org	.497	.719
stackLSTM	.486	.692
LSTM	.541	.68
stackLSTM_reg	<b>.583</b>	<b>.747</b>

Table 2: Macro F1 and Fnc score for proposed models

#### ۶.۴ آزمایشات دیگر

ما همچنین از Embedding Bert [۱۹] برای استخراج ویژگی در تشخیص موضع عنوان خبر به یک ادعا استفاده کردیم. به دلیل محدودیت زمان و GPU گوگل؛ امکان استفاده از این مدل برای تشخیص موضع متن خبر نسبت به ادعا میسر نبود. به همین دلیل تنظیمات و نتایج را بصورت جداگانه در این قسمت آورده ایم.

به دو روش برای تشخیص موضع عنوان خبر نسبت به ادعا از Bert در مدل stackLSTM استفاده کردیم:

۱. ویژگی ها : گرفتن ۱۵ کلمه از ادعا و ۱۵ کلمه از عنوان خبر و بدست آوردن Embedding Bert تمام این ۳۰

کلمه بوسیله کتابخانه Flair [۲۰] + ویژگی سوالی بودن ادعا + ویژگی چندقسمتی بودن ادعا  
تنظیمات و نتیجه : در این حالت دو ویژگی سوالی بودن و چندقسمتی بودن را با خروجی LSTM دوم concat  
کردیم و سپس این ویژگی ها را از سه لایه dense عبور داده و نهایتاً به لایه خروجی دادیم. تمامی پارامترهای  
تاثیر گذار را با مقدار های مختلف تست کردیم که بهترین نتیجه با این تنظیمات حاصل شد:

batch\_size = 200, kernel\_initializer= glorot\_uniform , regularizer = regularizers.l2(0.01) , epochs = 100, dropout for dense layer = 0 , dropout for both lstm = 0.1, neuron count for each dense = 500, neuron count for each LSTM = 30

Results : F1-score = .501 , Accuracy = .54 , Loss = 1.57

۲. ویژگی ها : همانند روش قبل؛ با این تفاوت که به جای ۱۵ کلمه، از ادعا ۱۲ کلمه ابتدا و از عنوان خبر نیز ۱۲ کلمه ابتدا را انتخاب کرده ایم.

تنظیمات و نتیجه : تمامی پارامترهای تاثیر گذار را با مقدار های مختلف تست کردیم که بهترین حالت و نتیجه  
بر روی داده تست به صورت زیر می باشد:

batch\_size = 200, kernel\_initializer= None , regularizer = regularizers.l2(0.01) , epochs = 100, dropout for dense layer = 0.5 , dropout for both lstm = 0.2, neuron count for each dense = 600, neuron count for each LSTM = 24

Results : F1-score = .53 , Accuracy = .551 , Loss = 1.15

یکی از دلایل پایین بودن معیارهای Accuracy و F1-Score نسبت به بیس لاین ها، ممکن است به دلیل ابعاد خیلی زیاد Embedding Bert باشد. ( Embedding Bert پیاده سازی شده در کتابخانه Flair برای هر کلمه یک بردار ۳۰۷۲ تایی تولید می کند.)

## ۵ تحلیل

تشخیص موضع بسیار به معنی خیر وابسته است. پس ما باید کلمات را در کنار یکدیگر ببینیم تا معنی واقعی جمله را بیابیم. ما تصمیم گرفتیم از شبکه ای مانند lstm استفاده کنیم که توالی کلمات را در نظر میگیرد. استفاده از دو lstm نیز باعث تقویت ویژگی های معنایی و همینطور پیچیده تر کردن مدل میشود. از این رو در این مقاله ما از مدل stackLSTM [۲] بعنوان مدل پایه استفاده کردیم و تنظیمات و پارامترهای آن را به گونه ای تغییر دادیم که به نتایج بهتری دست پیدا کنیم. همچنین ما ویژگی های مختلفی را از متن استخراج کردیم که همه یا تعدادی از آنها را در هر آزمایش مورد استفاده قرار دادیم.

با توجه به نتایج، ساده تر کردن مدل باعث بهبود کارایی روی مجموعه دادگان فارسی شده است. با این حال تمامی مدل های پیشنهادی عملکرد ضعیفی در تشخیص کلاس اقلیت (agree) دارند. این ممکن است به این دلیل باشد که اکثر ویژگی های استفاده شده بر پایه ی شباهت کلمات بین متن و ادعا می باشد. زیرا که تمامی متن و ادعاهای مرتبط شامل تعداد کلمات مشابه فراوان هستند. و چون تعداد داده ی agree از همه کمتر است، این مدل شباهت را جزء دسته ی دیگری قرار میدهد.

در هر حال مدل نهایی ما از نظر هر ۴ معیار ارزیابی بهتر از سایر مدل ها عمل کرده است.

## منابع

- [1] Dean Pomerleau and Delip Rao. 2017. The fake news challenge: Exploring how artificial intelligence technologies could be leveraged to combat fake news.
- [2] Andreas Hanselowski, Avinesh PVS, 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 Proceedings of the 27th International Conference on Computational Linguistics, COLING '18, pages 1859–1874, Santa Fe, NM, USA
- [3] Peter Dizikes. 2018. Study: On twitter, false news travels faster than true stories.
- [4] Kashyap Papat, Subhabrata Mukherjee, Andrew Yates, and Gerhard Weikum. 2018. Declare: Debunking fake news and false claims using evidence-aware deep learning. In EMNLP.
- [5] Soroush Vosoughi, Brandon C. Roy, and Sinan Aral. 2018. The spread of true and false news online. Science, 359:1146–1151.

- [6] Sihao Chen, Daniel Khashabi, Wenpeng Yin, Chris Callison-Burch, and Dan Roth. 2019. Seeing Things from a Different Angle: Discovering Diverse Perspectives about Claims. In The 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2019). Minneapolis, Minnesota. <http://www.cis.upenn.edu/~ccb/publications/discovering-diverse-perspectives.pdf>
- [7] Jing Ma, Wei Gao, and Kam-Fai Wong. 2018. Detect rumor and stance jointly by neural multi-task learning. In Companion Proceedings of the The Web Conference 2018. WWW '18, pages 585–593
- [8] Mitra Mohtarami, Ramy Baly, James Glass, Preslav Nakov, Lluís Màrquez, and Alessandro Moschitti. Automatic stance detection using end-to-end memory networks. In Proceedings of the 16th Annual Conference of the North American Chapter of the Association for Computational Linguistics, NAACL-HLT '18, New Orleans, LA, USA, 2018
- [9] S. Sadiq, N. Wagner, M. Shyu and D. Feaster, "High Dimensional Latent Space Variational AutoEncoders for Fake News Detection," 2019 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR), San Jose, CA, USA, 2019, pp. 437-442
- [10] Qiang Zhang, Emine Yilmaz, and Shangsong Liang. 2018. Ranking-based Method for News Stance Detection. In Companion Proceedings of the The Web Conference 2018. ACM Press.
- [11] Qiang Zhang, Shangsong Liang, Aldo Lipani, Zhaochun Ren, and Emine Yilmaz. 2019. From Stances' Imbalance to Their Hierarchical Representation and Detection. In Companion Proceedings of the The Web Conference 2019. ACM Press
- [12] 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.
- [13] fastText.cc. Word vectors for 157 languages. Fasttext
- [14] Majid Zarharan, Samane Ahangar, Fateme rezvani nejad, Mahdi lotfi, Shaghayegh Jalali, Sauleh etemadi, Mohammad taher pilevar and Behrouz Minaei. Persian stance detection. Conference for truth and trust online london October 4-5, 2019.
- [15] Fabian Pedregosa, Gael 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. Scikitlearn: Machine learning in python. Journal of Machine Learning Research, 12:2825–2830.
- [16] Koby Crammer and Yoram Singer. 2001. On the algorithmic implementation of multiclass kernel-based vector machines. Journal of Machine Learning Research, 2:265–292.
- [17] Leo Breiman. 2001. Random forests. Machine Learning, 45:5–32.
- [18] Harry Zhang. 2004. The optimality of naive bayes. In FLAIRS Conference.
- [19] J. Devlin, M. Chang, K. Lee, and K. Toutanova. BERT: pretraining of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805, 2018.
- [20] Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. In Proceedings of the 27th International Conference on Computational Linguistics, pages 1638–1649.