

## *Stance Detection Dengan Algoritme Gated Recurrent Unit (GRU)*

Maharani Aulia Syifa<sup>1</sup>, Dewi Retno Sari Saputro<sup>2</sup>

<sup>1,2</sup>Program Studi Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam,  
Universitas Sebelas Maret, Indonesia

*Corresponding author:* maharani.aulia@student.uns.ac.id

**Abstrak.** Pentingnya menangani berita palsu menjadi isu yang perlu mendapatkan perhatian khusus. Salah satu sarana yang mudah untuk menyebarkan suatu informasi yang belum tentu akurat adalah melalui media sosial. Diperlukan teknik untuk mendeteksi berita palsu, salah satunya dengan deteksi sikap (*stance detection*). *Stance detection* berfokus pada sikap penulis teks dalam menanggapi suatu klaim informasi, apakah sikap tersebut mendukung (*favor*), menentang (*against*), atau *none*. Salah satu metode untuk memproses data teks berdasarkan *stance detection* adalah *Recurrent Neural Network* (RNN) dengan algoritme *Gated Recurrent Unit* (GRU). Secara umum, RNN merupakan salah satu jenis *Neural Network* yang digunakan untuk memproses data berurutan (*sequential data*). Dalam struktur arsitekturnya, RNN menggunakan *looping* untuk dapat mengelola informasi dari masa lalu sehingga secara otomatis memungkinkan informasi dari masa lalu tetap tersimpan. Kelebihan GRU yaitu memiliki sedikit parameter, memiliki kemampuan yang lebih baik dalam menangani *long-term dependencies* dan membuatnya sesuai untuk melakukan tugas seperti *stance detection*, serta memiliki kemampuan dalam mengatasi data yang hilang dibandingkan dengan algoritme lain pada RNN. Berdasarkan hal tersebut, penelitian ini bertujuan untuk melakukan kajian tentang algoritme GRU dalam melakukan *stance detection*.

**Kata Kunci:** *stance detection, RNN, GRU, vanishing gradient, long-term dependencies.*

## 1 PENDAHULUAN

Media berita, termasuk portal berita *online* dan media sosial, tidak hanya memungkinkan suatu individu dengan mudah mengakses berita dan berbagi informasi secara *real-time*, tetapi juga memungkinkan berita palsu tersebar dengan cepat tanpa adanya verifikasi kebenaran [1]. Berita palsu, atau yang sering dikenal sebagai *hoax*, merupakan informasi yang direayasa untuk menyembunyikan fakta sebenarnya demi membentuk kepercayaan publik dengan tujuan mendukung kepentingan suatu individu atau kelompok. Kemampuan berita palsu dalam menggiring opini masyarakat memberikan dampak yang berbahaya, di antaranya dapat menyebabkan perpecahan di kalangan masyarakat, bahkan tak jarang terjadi pemberian ancaman (*hate speech*) antara satu kelompok atau individu dengan kelompok atau individu lain. Oleh karena itu, hal ini menjadi masalah yang serius bagi media pers beserta masyarakat.

Mendeteksi berita palsu, di sisi lain, merupakan hal yang sulit dilakukan sebab untuk mengklasifikasikan sebuah berita sebagai berita palsu, diperlukan model yang dapat meringkas berita dan membandingkannya dengan berita asli [2]. Berdasarkan hal tersebut, digunakan pendekatan yang berbeda dalam mendeteksi berita palsu, yakni dengan memanfaatkan salah satu tugas pada *Natural Language Processing* (NLP), yaitu *stance detection*. *Stance detection* merupakan proses untuk mengklasifikasikan sikap penulis teks dalam menanggapi suatu klaim informasi, di mana sikap tersebut dikategorikan menjadi tiga, yaitu mendukung (*favor*), menentang (*against*), atau *none* [3].

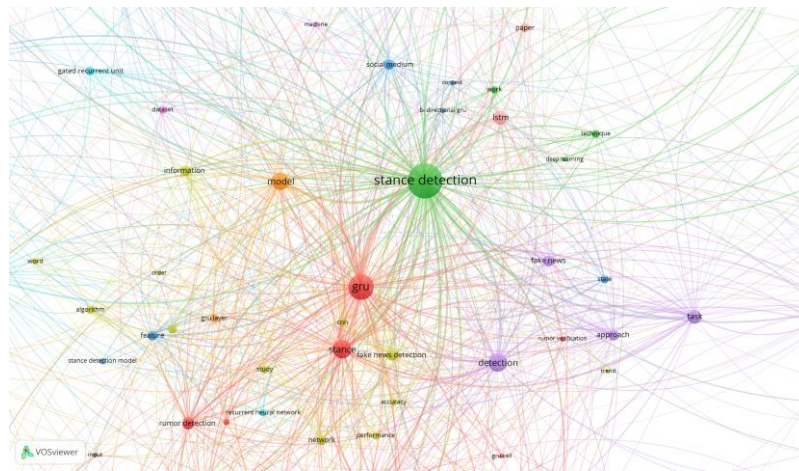
Salah satu model *deep learning*, yaitu *Recurrent Neural Network* (RNN) banyak digunakan pada *stance detection* untuk mendeteksi berita palsu [4]. Hal ini ditunjukkan melalui penelitian yang dilakukan oleh Karande *et al.* [5], Chen *et al.* [6], dan Mrowca *et al.* [7]. Karande *et al.* [5] menerapkan algoritme *Long Short-Term Memory* (LSTM) pada *stance detection* menggunakan *dataset MCIntire Fake and Real News*. Tidak hanya pada penelitian Karande *et al.* [5], algoritme LSTM juga diterapkan oleh Chen *et al.* [6] pada penelitiannya untuk melakukan *stance detection* pada *dataset Fake News Challenge* (FNC). Berbeda dari dua penelitian sebelumnya, Mrowca *et al.* [7] memanfaatkan algoritme *Bidirectional Long Short-Term Memory* (BiLSTM) untuk mengidentifikasi berita palsu dengan *stance detection* pada *dataset FNC*. Berdasarkan penelitian-penelitian sebelumnya, pada penelitian ini dilakukan kajian mengenai algoritme *Gated Recurrent Unit* (GRU) dalam melakukan *stance detection* sebagai bentuk pengembangan terhadap penelitian-penelitian terdahulu.

## 2 TINJAUAN PUSTAKA

### 2.1 Vosviewer

Vosviewer adalah *software* yang digunakan untuk melakukan analisis bibliometrik, yaitu metode untuk menganalisis dan mengeksplorasi sebaran jumlah publikasi serta sitasi ilmiah dari berbagai literatur [8]. Pada penelitian ini, Vosviewer digunakan untuk mengidentifikasi *research gap* mengenai algoritme

GRU dan *stance detection* pada penelitian-penelitian terdahulu yang diambil melalui Scopus.

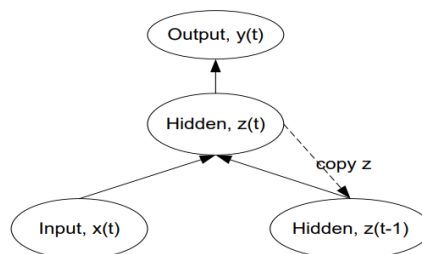


**Gambar 1.** Hasil Vosviewer Terkait Stance Detection dan GRU

Terdapat beberapa penelitian yang telah dilakukan terkait dengan *stance detection* dan GRU. Berdasarkan Gambar 1, dapat dilihat bahwa *stance detection* memiliki hubungan yang erat dengan GRU. Selain itu, *stance detection* dan GRU berkaitan dengan beberapa bidang penerapan, seperti *fake news*, *fake news detection*, dan *rumor detection*. Berdasarkan hasil Vosviewer mengenai perkembangan jurnal terkait GRU dan *stance detection* pada tahun 2014-2022 yang terindeks scopus, tahun 2020 menjadi tahun dengan publikasi tertinggi sebanyak 212 publikasi.

## 2.2 Recurrent Neural Network (RNN)

*Recurrent Neural Network* (RNN) merupakan jenis dari *neural network* yang digunakan untuk menangani data berurutan (*sequential data*) seperti *time series*. Berbeda dengan jenis *feedforward neural network*, RNN memiliki perulangan/*loops* yang membuatnya mampu mengenali informasi dari masa lalu saat melakukan proses *model training* sehingga memungkinkan suatu informasi dapat bertahan dari waktu ke waktu [9]. Oleh [10], Arsitektur RNN ditunjukkan pada Gambar 2.



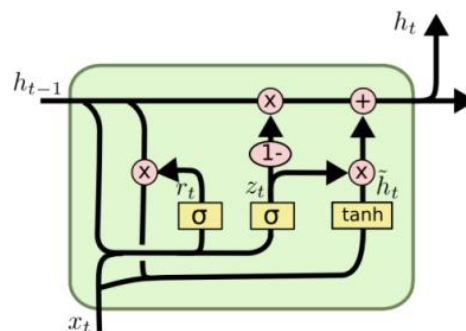
**Gambar 2.** Arsitektur *Recurrent Neural Network* (RNN)

### 2.3 Vanishing Gradient Problem

*Vanishing gradient problem* adalah masalah yang sering dijumpai dalam RNN. Permasalahan ini muncul ketika *gradient* yang digunakan untuk memperbarui nilai bobot selama proses *backpropagation* menjadi sangat kecil, sehingga membuat *cost function*, fungsi yang dapat mengukur seberapa baik performa dari *task* yang dijalankan oleh model *neural network*, berada pada nilai minimum lokal yang menyebabkan jaringan tidak dapat mempelajari *task*-nya dengan efektif [11] dan menyulitkan jaringan untuk dapat mempelajari *long-term dependencies* [12]. Akibatnya, jaringan memiliki kemungkinan untuk menghasilkan performa yang buruk.

### 2.4 Gated Recurrent Unit (GRU)

GRU adalah variasi dari model *neural network Long Short-Term Memory* (LSTM) yang diciptakan oleh Kyunghyun Cho pada tahun 2014 sebagai modifikasi dari struktur sel LSTM dengan menggabungkan tiga *gating* unit LSTM menjadi dua *gating* unit, yakni *update gate* ( $z_t$ ) dan *reset gate* ( $r_t$ ) [13]. Hal ini menyebabkan parameter model GRU relatif sedikit, sumber daya komputasi tambahan yang diperlukan untuk melatih model di luar arsitektur model (*training overhead*) menjadi berkurang, dan *long-term dependencies* dapat ditangani dengan baik [14]. Tidak hanya itu, arsitektur sel pada GRU yang memungkinkan untuk melakukan *update* dan *reset* nilai *hidden state* secara selektif, membuatnya dapat mengatasi *vanishing gradient problem* saat melakukan proses *backpropagation* dengan cukup efektif [15]. Pada arsitektur sel GRU, *reset gate* ( $r_t$ ) berperan untuk menentukan bagian dari informasi yang tidak relevan yang perlu dihilangkan, sementara *update gate* ( $z_t$ ) berperan dalam menentukan jumlah memori sebelumnya yang perlu disimpan [10]. Arsitektur sel GRU dapat dilihat pada Gambar 3.



**Gambar 3.** Arsitektur Sel *Gated Recurrent Unit* (GRU)

Model GRU terdiri dari 3 lapisan, yaitu lapisan *input* (*input layer*), lapisan *output* (*output layer*), dan lapisan implisit (*implicit layer*). Data *input* dari GRU merupakan data yang berjenis *time series* pada waktu  $t$  setelah dilakukan *preprocessing* data. Algoritme pada sel GRU diuraikan sebagai berikut;

- 1) Menentukan *reset gate* ( $r_t$ ) yang berfungsi untuk menghitung banyak informasi masa lalu yang kurang relevan dan perlu dihapuskan,

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t]) \quad (1)$$

Pada persamaan (1), *input*  $x_t$  dan  $h_{t-1}$  atau *input* yang menyimpan informasi pada waktu  $t-1$ , dikalikan dengan bobot  $W_r$ .

- 2) Menentukan *update gate* pada waktu  $t$  ( $z_t$ ) dengan mengalikan *input*  $x_t$  dan  $h_{t-1}$  yang masuk ke dalam jaringan dengan bobot  $W_z$  untuk menghitung berapa banyak informasi dari masa lalu yang akan diteruskan ke masa depan. *Gate* ini membuat model mampu menyalin semua informasi masa lalu dan terhindar dari *vanishing gradient problem*,

$$z_t = \sigma(W_z \cdot [h_{t-1}, x_t]) \quad (2)$$

- 3) Menetapkan nilai *candidate hidden state* ( $\tilde{h}_t$ ), yakni *output* pada waktu  $t$  dengan *tanh layer* dan *reset gate* ( $r_t$ ) untuk mengembalikan informasi yang relevan di masa lalu,

$$\tilde{h}_t = \tanh(W_h \cdot [r_t \cdot h_{t-1}, x_t]) \quad (3)$$

- 4) Menetapkan *output* standar unit GRU pada waktu  $t$  yang berfungsi untuk menyimpan informasi dari unit saat ini menggunakan *output* dari *update gate* ( $z_t$ ), *candidate hidden state* ( $\tilde{h}_t$ ), dan standar unit GRU pada waktu  $t-1$ ,

$$h_t = (1 - z_t) \cdot h_{t-1} + z_t \cdot \tilde{h}_t \quad (4)$$

## 2.5 Stance Detection

*Stance detection* didefinisikan sebagai proses mengklasifikasikan sikap penulis terhadap suatu klaim informasi. *Input* dari *stance detection* berupa pasangan dari *body text* (isi) dan *headline* (klaim informasi), serta *outputnya* berupa kategori yang terdiri dari mendukung (*favor*), menentang (*against*), dan tidak keduanya (*none*) [3]. Pada pembagian 3 kategori ini, kategori *none* diberikan jika sikap *body text* (isi) terhadap *headline* (klaim informasi) tidak mendukung atau tidak menentang. Berbeda dari 3 kategori dalam *output stance detection* sebelumnya, kategori *output* pada tugas *stance detection* yang diperkenalkan *Fake News Challenge* (FNC1) dari Emergent terdiri dari 4 label, yaitu *agree*, *disagree*, *unrelated*, dan *discuss* [16]. Pada 4 kategori *output stance detection* ini, *unrelated* dikategorikan untuk *body text* (isi) yang tidak memiliki kaitan dengan *headline* (klaim informasi) dan kategori *discuss* diberikan jika *body text* (isi) hanya mendiskusikan topik yang serupa dengan *headline*, tanpa memberikan sikap mendukung ataupun menentang [17].

## 3 METODE

Metode penelitian menggunakan studi literatur atau *literature review* dengan mencari referensi teori dari penelitian-penelitian sebelumnya yang berkaitan terhadap topik penelitian, yaitu kajian mengenai algoritme GRU untuk melakukan *stance detection*.

### 3.1 Data Penelitian

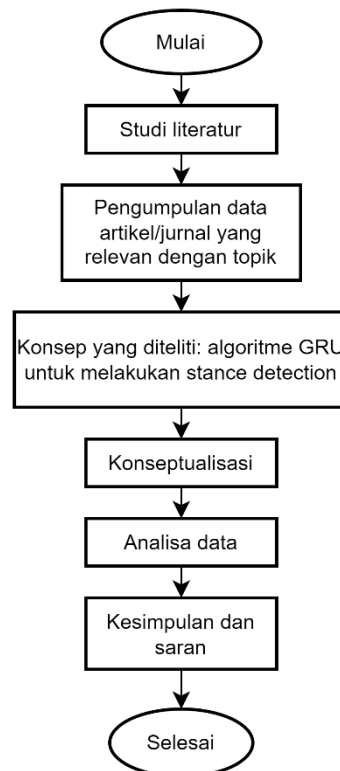
Data yang digunakan adalah artikel ilmiah atau jurnal yang memiliki relevansi dengan topik penelitian. Sumber literatur didapatkan melalui *digital library* Google Scholar, Semantic Scholar, Scopus, arXiv, dan Science Direct.

### 3.2 Prosedur Penelitian

Langkah penelitian yang dilakukan adalah sebagai berikut.

- a. Studi literatur,
- b. Pengumpulan data artikel atau jurnal yang relevan dengan topik,
- c. Konsep yang diteliti: algoritme GRU untuk melakukan *stance detection*,
- d. Konseptualisasi,
- e. Analisa data,
- f. Kesimpulan dan saran

Berikut dilampirkan diagram alir dari langkah penelitian pada Gambar 4.



Gambar 4. Diagram Alir Langkah Penelitian

## 4 PEMBAHASAN

GRU merupakan model *neural network* yang dapat digunakan untuk melakukan *stance detection* karena mampu memodelkan data berurutan seperti teks dan mampu menyimpan *long-term dependencies* antara kata pada sebuah kalimat yang berfungsi untuk memahami suatu konteks atau arti dari sebuah teks. Hal ini berperan penting dalam *stance detection* sebab sikap penulis terhadap suatu topik dapat dipengaruhi oleh konteks dari sebuah teks. Dalam *stance detection*, GRU digunakan sebagai *classifier*. Pada model GRU, *input* yang diberikan umumnya adalah vektor dari *body text*/penggalan kalimat/dokumen yang diperoleh setelah melewati proses *pre-processing* dan *word embedding* [2]. Fokus utama dalam pengklasifikasian *stance detection* dengan GRU adalah membandingkan apakah *body text* atau penggalan kalimat memiliki tanggapan

terhadap *headline*/topik. Nantinya, tanggapan tersebut akan dikategorikan menjadi 3 label, yaitu mendukung (*favor*), menentang (*against*), atau tidak keduanya (*none*).

Selanjutnya, diuraikan mengenai *syntax* membangun model GRU untuk melakukan *stance detection* dengan menggunakan Python sebagai berikut.

```
embedding_size= 100
model= Sequential()
model.add(Embedding(max_features, embedding_size,
input_length=maxlen))
model.add(GRU(128,dropout=0.2, recurrent_dropout=0.2))
model.add(Dense(3, activation='softmax'))
model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])
```

*Embedding* merupakan representasi vektor yang dipelajari model dari kata-kata dalam suatu teks. Melalui *syntax* `embedding_size= 100`, diberikan nilai *embedding size* sebesar 100 sebagai nilai *default* untuk dimensi *embedding*. Selanjutnya, dilakukan pendefinisian model menjadi *sequential* model, yaitu model yang memuat serangkaian lapisan yang ditambahkan secara berurutan menggunakan *syntax* `model= Sequential()`. Kemudian pada *syntax* `model.add(Embedding(max_features,embedding_size,input_length=maxlen))`, ditambahkan lapisan *embedding* ke model dengan tujuan untuk mengonversikan *input* teks menjadi representasi vektor menggunakan dimensi *embedding* yang sudah ditentukan sebelumnya.

Setelah menambahkan lapisan *embedding*, ditambahkan lapisan GRU ke dalam model untuk memproses urutan data di dalam teks menggunakan *syntax* `model.add(GRU(128, dropout=0.2, recurrent_dropout=0.2))`. Angka 128 pada *syntax* tersebut menunjukkan jumlah *output* pada lapisan GRU yang dihasilkan dalam setiap waktu. Diberikan *dropout* dan *recurrent dropout* sebesar 0,2 untuk mengatasi *overfitting* pada model dengan mematikan secara acak 20% unit GRU dan 20% koneksi rekurensi selama pelatihan. Karena terdapat 3 kategori pengklasifikasian dalam *stance detection*, maka ditambahkan 3 unit lapisan *dense* ke model dan fungsi aktivasi *softmax* dengan *syntax* `model.add(Dense(3,activation='softmax'))`.

Langkah terakhir adalah melakukan kompilasi model dengan menggunakan konfigurasi *optimizer* 'Adaptive Moment Estimation (Adam)', dan *loss function* 'categorical\_crossentropy', serta metrik evaluasi 'accuracy' untuk melihat tingkat akurasi model yang ditunjukkan pada *syntax* `model.compile(optimizer='adam', loss='categorical_crossentropy', metrics=['accuracy'])`.

## 5 KESIMPULAN

Berdasarkan hasil dan pembahasan, disimpulkan bahwa GRU memiliki kemampuan untuk mengatasi permasalahan *vanishing gradient problem* yang sering muncul pada RNN. Tidak hanya itu, GRU merupakan variasi dari LSTM dengan parameter yang lebih sederhana sehingga membuat GRU memiliki kemampuan menangani *long-term dependencies* yang sama baiknya dengan

LSTM, tetapi memiliki kompleksitas yang lebih sederhana dari LSTM. Parameter yang lebih sedikit ini membuat GRU tidak membutuhkan waktu yang lama dalam melatih model. Oleh karena itu, algoritme GRU mampu diterapkan sebagai pengklasifikasi atau *classifier* untuk melakukan *stance detection* dengan kompleksitas yang lebih sederhana, tetapi memiliki performa yang sama baiknya dengan LSTM. Namun, diperlukan penelitian lebih lanjut menggunakan *dataset* berita palsu untuk mengetahui seberapa baik tingkat akurasi model GRU untuk *stance detection* pada berita palsu.

#### DAFTAR PUSTAKA

- [1] Q. Zeng, Q. Zhou, and S. Xu, "Neural Stance Detectors for Fake News Challenge," *CS224n Nat. Lang. Process. with Deep Learn.*, pp. 1–9, 2017.
- [2] E. I. Setiawan and I. Lestari, "Stance Classification Pada Berita Berbahasa Indonesia Berbasis Bidirectional LSTM," *J. Intell. Syst. Comput.*, vol. 3, no. 1, pp. 41–48, 2021, doi: 10.52985/insyst.v3i1.148.
- [3] N. Alturayef, H. Luqman, and M. Ahmed, *A systematic review of machine learning techniques for stance detection and its applications*, vol. 35, no. 7. Springer London, 2023. doi: 10.1007/s00521-023-08285-7.
- [4] M. Umer, Z. Imtiaz, S. Ullah, A. Mehmood, G. S. Choi, and B. W. On, "Fake news stance detection using deep learning architecture (CNN-LSTM)," *IEEE Access*, vol. 8, pp. 156695–156706, 2020, doi: 10.1109/ACCESS.2020.3019735.
- [5] H. Karande, R. Walambe, V. Benjamin, K. Kotecha, and T. S. Raghu, "Stance detection with BERT embeddings for credibility analysis of information on social media," *PeerJ Comput. Sci.*, vol. 7, pp. 1–20, 2021, doi: 10.7717/peerj-cs.467.
- [6] J. Y. Chen, J. Johnson, and G. Yennie, "RNNs for Stance Detection between News Articles," *Stanford Univ. California, US, rep.*, pp. 1–7, 2017, [Online]. Available: <https://github.com/shyamupa/snli-entailment/blob/>
- [7] D. Mrowca and E. Wang, "Stance detection for fake news identification," *Stanford Univ. Stanford, CA, Proj. Report*, pp. 1–12, 2017, [Online]. Available: <https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1174/reports/2760496.pdf>
- [8] P. H. -, S. B. Utami, and N. Karlina, "Analisis Bibliometrik: Perkembangan Penelitian Dan Publikasi Mengenai Koordinasi Program Menggunakan Vosviewer," *J. Pustaka Budaya*, vol. 9, no. 1, pp. 1–8, 2022, doi: 10.31849/pb.v9i1.8599.
- [9] A. C. S. Kumar, S. M. Bhandarkar, and M. Prasad, "DepthNet: A recurrent neural network architecture for monocular depth prediction," *IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Work.*, vol. 2018-June, pp. 396–404, 2018, doi: 10.1109/CVPRW.2018.00066.
- [10] A. Hanifa, S. A. Fauzan, M. Hikil, and M. B. Ashfiya, "Perbandingan Metode Lstm Dan Gru (RNN) Untuk Klasifikasi Berita Palsu Berbahasa Indonesia Comparison Of LSTM And GRU (RNN) Methods For Fake News Classification In Indonesian," *Din. Rekayasa*, vol. 17, no. 1, pp. 33–



- 39, 2021, [Online]. Available: <https://covid19.go.id/p/hoax-buster>.
- [11] K. J. M. Tarnate, M. Devaraj, and J. C. De Goma, "Overcoming the vanishing gradient problem of recurrent neural networks in the ISO 9001 quality management audit reports classification," *Int. J. Sci. Technol. Res.*, vol. 9, no. 3, pp. 6683–6686, 2020.
- [12] P. Le and W. Zuidema, "Quantifying the vanishing gradient and long distance dependency problem in recursive neural networks and recursive lstms," *Proc. Annu. Meet. Assoc. Comput. Linguist.*, no. 2010, pp. 87–93, 2016, doi: 10.18653/v1/w16-1610.
- [13] S. Mahjoub, L. Chrifi-Alaoui, B. Marhic, and L. Delahoche, "Predicting Energy Consumption Using LSTM, Multi-Layer GRU and Drop-GRU Neural Networks," *Sensors*, pp. 1–20, 2022, doi: 10.3390/s22114062.
- [14] X. Wang, J. Xu, W. Shi, and J. Liu, "OGRU: An Optimized Gated Recurrent Unit Neural Network," *J. Phys. Conf. Ser.*, vol. 1325, no. 1, pp. 0–7, 2019, doi: 10.1088/1742-6596/1325/1/012089.
- [15] S. Squartini, A. Hussain, and F. Piazza, "Preprocessing based solution for the vanishing gradient problem in recurrent neural networks," *Proc. - IEEE Int. Symp. Circuits Syst.*, vol. 5, pp. 713–716, 2003, doi: 10.1109/iscas.2003.1206412.
- [16] A. Hanselowski *et al.*, "A retrospective analysis of the fake news challenge stance detection task," *COLING 2018 - 27th Int. Conf. Comput. Linguist. Proc.*, pp. 1859–1874, 2018.
- [17] J. M. Sholar, S. Chopra, and S. Jain, "Towards Automatic Identification of Fake News: Headline-Article Stance Detection with LSTM Attention Models," *Stanford CS224d Deep Learn. NLP Final Proj.*, vol. 1, pp. 1–15, 2017.
- [18] R. Dey and F. M. Salem, "Gate-Variants of Gated Recurrent Unit (GRU)," *Midwest Symp. Circuits Syst. Inst. Electr. Electron. Eng. Inc.*, vol. 784, no. 2017, pp. 1597–1600, 2017.