

**KLASIFIKASI TEKS MENGGUNAKAN LONG SHORT-TERM
MEMORY DENGAN FITUR WORD EMBEDDING GLOVE
DAN WORD2VEC**



OLEH:

**WINDA KURNIA SARI
09042611822004**

**PROGRAM STUDI MAGISTER ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER
UNIVERSITAS SRIWIJAYA
2020**

**KLASIFIKASI TEKS MENGGUNAKAN LONG SHORT-TERM
MEMORY DENGAN FITUR WORD EMBEDDING GLOVE
DAN WORD2VEC**

TESIS

**Diajukan Untuk Melengkapi Salah Satu Syarat
Memperoleh Gelar Magister**



OLEH:

**WINDA KURNIA SARI
09042611822004**

**PROGRAM STUDI MAGISTER ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER
UNIVERSITAS SRIWIJAYA
2020**

LEMBAR PENGESAHAN

**KLASIFIKASI TEKS MENGGUNAKAN LONG SHORT-TERM
MEMORY DENGAN FITUR WORD EMBEDDING GLOVE DAN
WORD2VEC**

TESIS

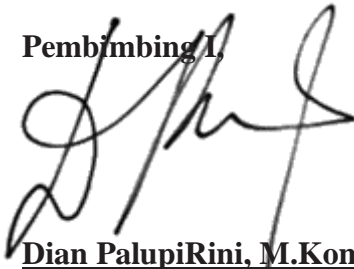
Diajukan Untuk Melengkapi Salah Satu Syarat
Memperoleh Gelar Magister

OLEH:

**WINDA KURNIA SARI
09042611822004**

Palembang, April 2020

Pembimbing I,



**Dian Palupi Rini, M.Kom., Ph.D
NIP197802232006042002**

Pembimbing II,



**Dr. Reza Firsandaya Malik, M.T
NIP 197604252010121001**

**Mengetahui,
Ketua Program Studi Magister Ilmu Komputer**



**Dr. Ir. H. Sukemi, M.T
NIP 196612032006041001**

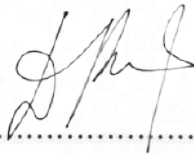
HALAMAN PERSETUJUAN

Pada Sabtu tanggal 28 Maret 2020 telah dilaksanakan ujian sidang tesis secara daring oleh Magister Ilmu Komputer Universitas Sriwijaya.

Nama : Winda Kurnia Sari
NIM : 09042611822004
Judul : Klasifikasi Teks Menggunakan Long Short-Term Memory
dengan Fitur Word Embedding Glove dan Word2vec

1. Pembimbing I

Dian Palupi Rini, M.Kom., Ph.D
NIP 197802232006042002



.....

2. Pembimbing II

Dr. Reza Firsandaya Malik, M.T
NIP 197604252010121001



.....

3. Penguji I


Samsuryadi, M.Kom., Ph.D
NIP 197102041997021003



.....

4. Penguji II

Dr. Yusuf Hartono, M.Sc
NIP 196411161990031002



.....

Mengetahui,
Ketua Program Studi Magister Ilmu Komputer



Dr. Ir. H. Sukemi, M.T
NIP 196612032006041001

LEMBAR PERNYATAAN

Yang bertanda tangan di bawah ini :

Nama : Winda Kurnia Sari
NIM : 09042611822004
Program Studi : Magister Ilmu Komputer
Judul Tesis : Klasifikasi Teks Menggunakan Long Short-Term Memory
dengan Fitur Word Embedding Glove dan Word2vec
Hasil Pengecekan Software iThenticate/Turnitin : 4 %

Menyatakan bahwa laporan tesis saya merupakan hasil karya sendiri dan bukan hasil penjiplakan/plagiat. Apabila ditemukan unsur penjiplakan/plagiat dalam laporan tesis ini, maka saya bersedia menerima sanksi akademik dari universitas Sriwijaya sesuai dengan ketentuan yang berlaku.

Demikian pernyataan ini saya buat dengan sebenarnya dan tidak ada paksaan oleh siapapun.



Palembang, April 2020



Winda Kurnia Sari
NIM 09042611822004

KATA PENGANTAR

Puji dan syukur penulis panjatkan kehadiran Allah SWT atas Rahmat-Nya sehingga penulis dapat menyelesaikan tesis yang berjudul “Klasifikasi Teks Menggunakan Long Short-Term Memory dengan Fitur Word Embedding Glove dan Word2vec” di susun untuk memenuhi sebagian persyaratan kelulusan untuk memperoleh gelar Magister Komputer pada Program Studi Magister Ilmu Komputer Universitas Sriwijaya.

Pada kesempatan ini penulis menyadari keterbatasan dan kelemahan yang ada dalam menyelesaikan tesis ini sehingga penulis ingin menyampaikan ucapan terima kasih kepada pihak-pihak yang telah memberikan dukungan, bimbingan dan motivasi kepada penulis untuk menyelesaikan tesis ini, kepada:

1. Ibu Dian Palupi Rini, M.Kom., Ph.D dan Bapak Dr. Reza Firsandaya Malik selaku pembimbing yang selama ini telah meluangkan waktu untuk memberikan saran dan kritik serta membantu dalam penyusunan tesis ini.
2. Bapak Dr. Ir. H. Sukemi, M.T., sebagai Ketua Program Studi Magister Ilmu Komputer Universitas Sriwijaya.
3. Bapak Jaidan Jauhari, S.Pd, M.T selaku Dekan Fakultas Ilmu Komputer Universitas Sriwijaya.
4. Terkhusus Orang Tua yang selalu mendoakan dan mendukung untuk penyelesaian S2 ini.
5. Iman Saladin B. Azhar yang selalu memberi dukungan dan bantuan selama penyelesaian tesis.
6. Seluruh teman-teman Magister Ilmu Komputer yang telah membantu dan memberikan semangat pada masa-masa perkuliahan dan tesis.

Akhir kata, penulis menyadari bahwa tesis ini masih banyak kekurangan baik dari isi maupun susunan. Semoga tesis ini dapat bermanfaat untuk kita semua.

Palembang, April 2020

Penulis

LEMBAR PERSEMBAHAN

“Hate no one, no matter how much they have wronged you. Live humbly, no matter how wealthy you become. Think positively, no matter how hard life is.”

- Ali Bin Abi Thalib -

Kupersembahkan untuk

Mama

Almarhum Papa

Almamater ku

TEXT CLASSIFICATION USING LONG SHORT-TERM MEMORY WITH WORD EMBEDDING FEATURES GLOVE AND WORD2VEC

ABSTRACT

Multilabel text classification is a task of categorizing text into one or more categories. Like other machine learning, multilabel classification performance is limited when trained small dataset and leads to the difficulty of capturing semantic relationships. It requires a multilabel text classification technique that can group four labels from news articles. Deep Learning is a proposed method for solving problems in multilabel text classification techniques. Some of the deep learning methods used for text classification include Convolutional Neural Networks, Autoencoders, Deep Belief Networks, and Recurrent Neural Networks (RNN). RNN is one of the most popular architectures used in natural language processing (NLP) because the recurrent structure is appropriate for processing variable-length text. One of the deep learning methods proposed in this study is RNN with the application of the Long Short-Term Memory (LSTM) architecture. By tuning the LSTM parameters using feature GloVe and Word2vec will make the model into eight models, it is useful to show that LSTM with additional features can achieve good performance in text classification. The results showed that text classification using LSTM with Glove obtained the highest accuracy in the sixth model with 95.17% and Word2Vec obtained the highest accuracy in the fifth model with 95.38%, while the average value of precision, recall, and F1-score was 95 %.

Keywords: *RNN, LSTM, GloVe, Word2Vec*

KLASIFIKASI TEKS MENGGUNAKAN LONG SHORT-TERM MEMORY DENGAN FITUR WORD EMBEDDING GLOVE DAN WORD2VEC

ABSTRAK

Klasifikasi teks multilabel adalah tugas mengategorikan teks ke dalam satu atau lebih kategori. Seperti pembelajaran mesin lainnya, kinerja klasifikasi multilabel terbatas ketika ada data kecil berlabel dan mengarah pada kesulitan menangkap hubungan semantik. Dibutuhkan teknik klasifikasi teks multilabel yang dapat mengelompokkan empat label dari artikel berita. Deep Learning adalah metode yang diusulkan untuk memecahkan masalah dalam teknik klasifikasi teks multilabel. Beberapa contoh metode deep learning yang digunakan untuk pengklasifikasian teks antara lain Convolutional Neural Networks, Autoencoder, Deep Belief Networks, dan Recurrent Neural Networks (RNN). RNN merupakan salah satu arsitektur yang paling populer yang digunakan dalam Pemrosesan Bahasa Alami (PBA) karena struktur recurrent cocok untuk proses teks variabel panjang. Salah satu metode deep learning yang diusulkan pada penelitian ini adalah RNN dengan penerapan arsitektur Long Short-Term Memory (LSTM). Dengan dilakukan *tuning* parameter LSTM menggunakan fitur GloVe dan Word2vec akan membuat model menjadi delapan model, gunanya untuk menunjukkan bahwa LSTM dengan fitur tambahan dapat mencapai kinerja yang baik dalam klasifikasi teks. Hasil penelitian menunjukkan bahwa klasifikasi teks menggunakan LSTM dengan Glove memperoleh akurasi tertinggi pada model keenam dengan 95,17% dan Word2Vec memperoleh akurasi tertinggi pada model kelima dengan 95,38%, sedangkan rata-rata nilai presisi, recall, dan F1-score adalah 95%.

Kata Kunci: *RNN, LSTM, GloVe, Word2Vec*

DAFTAR ISI

	Hal
DAFTAR ISI	iii
DAFTAR TABEL	iv
DAFTAR GAMBAR	v
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan Penelitian	3
1.4 Manfaat Penelitian	3
1.5 Batasan Masalah	3
1.6 Metodologi Penulisan	4
BAB II TINJAUAN PUSTAKA	6
2.1 Penelitian Terkait	6
2.2 Representasi Teks	8
2.3 Klasifikasi Teks	9
2.4 Ekstraksi Fitur	9
2.5 Recurrent Neural Network (RNN)	10
2.5.1 Long-short Term Memory (LSTM)	10
2.5.2 Gated Recurrent Units (GRU)	13
2.6 Confusion Matrix	15
BAB III METODOLOGI PENELITIAN	17
3.1 Kerangka Kerja Penelitian	17
3.2 Penelusuran Pustaka	17
3.3 Persiapan Data	20
3.4 Proses Pra-Pemrosesan	21
3.4.1 <i>One-hot Encoding</i>	21
3.4.2 <i>Tokenization</i> dan Penghapusan Tanda Baca (<i>Punctuation</i>)	21
3.5 Pengklasifikasian LSTM	22
3.5.1 Proses Pelatihan	22

3.5.2 Proses Validasi	22
3.5.3 Proses Pengujian	23
3.6 Analisis Hasil	23
3.7 Kesimpulan dan Saran	24
BAB IV HASIL DAN ANALISA	25
4.1 Hasil Model Fine-Tuning LSTM – Glove	25
4.1.1 Model 1	26
4.1.2 Model 2	28
4.1.3 Model 3	29
4.1.4 Model 4	31
4.1.5 Model 5	33
4.1.6 Model 6	34
4.1.7 Model 7	35
4.1.8 Model 8	37
4.2 Hasil Model Fine-Tuning LSTM – Word2Vec	40
4.2.1 Model 1	40
4.2.2 Model 2	41
4.2.3 Model 3	42
4.2.4 Model 4	44
4.2.5 Model 5	45
4.2.6 Model 6	47
4.2.7 Model 7	49
4.2.8 Model 8	50
4.3 Hasil Model Fine-Tuning LSTM	52
4.4 Analisis Perbandingan Model	58
4.5 Studi Perbandingan	63
BAB V KESIMPULAN DAN SARAN	65
5.1 Kesimpulan	65
5.2 Saran	66
DAFTAR PUSTAKA	75

DAFTAR TABEL

	Hal
TABEL 2.1 Kesimpulan dari <i>review paper</i> diatas	8
TABEL 2.2 Penelitian menggunakan model <i>word embedding</i>	14
TABEL 2.3 <i>Confusion matrix</i> Kelas 0	15
TABEL 2.4 <i>Confusion matrix</i> Kelas 1	15
TABEL 2.5 <i>Confusion matrix</i> Kelas 2	15
TABEL 2.6 <i>Confusion matrix</i> Kelas 3	16
TABEL 3.1 Spesifikasi Dataset	20
TABEL 3.2 Model Validasi LSTM dengan Fitur GloVe dan Word2vec	23
TABEL 3.3 Model Validasi LSTM	24
TABEL 4.1 Hasil Kinerja Evaluasi Proses Pelatihan LSTM dengan Parameter Aktivasi Relu, Optimizer Adam dengan learning rate 0,001	26
TABEL 4.2 Hasil Kinerja Evaluasi Proses Pengujian LSTM dengan Parameter Aktivasi Relu, Optimizer Adam dengan learning rate 0,001	27
TABEL 4.3 Hasil Kinerja Evaluasi Proses Pelatihan LSTM dengan Parameter Aktivasi Tanh, Optimizer Adam dengan learning rate 0,001	28
TABEL 4.4 Hasil Kinerja Evaluasi Proses Pengujian LSTM dengan Parameter Aktivasi Tanh, Optimizer Adam dengan learning rate 0,001	28
TABEL 4.5 Hasil Kinerja Evaluasi Proses Pelatihan LSTM dengan Parameter Aktivasi Relu, Optimizer RMSpro dengan learning rate 0,001 ..	30
TABEL 4.6 Hasil Kinerja Evaluasi Proses Pengujian LSTM dengan Parameter Aktivasi Relu, Optimizer RMSpro dengan learning rate 0,001 ..	30
TABEL 4.7 Hasil Kinerja Evaluasi Proses Pelatihan LSTM dengan Parameter Aktivasi Tanh, Optimizer RMSpro dengan learning rate 0,001 ..	31
TABEL 4.8 Hasil Kinerja Evaluasi Proses Pengujian LSTM dengan Parameter Aktivasi Tanh, Optimizer RMSpro dengan learning rate 0,001 ..	32
TABEL 4.9 Hasil Kinerja Evaluasi Proses Pelatihan LSTM dengan Parameter Aktivasi Relu, Optimizer Adam dengan learning rate 0,0001 ..	33
TABEL 4.10 Hasil Kinerja Evaluasi Proses Pengujian LSTM dengan Parameter Aktivasi Relu, Optimizer Adam dengan learning rate 0,001 ..	33
TABEL 4.11 Hasil Kinerja Evaluasi Proses Pelatihan LSTM dengan Parameter Aktivasi Tanh, Optimizer Adam dengan learning rate 0,0001 ..	34
TABEL 4.12 Hasil Kinerja Evaluasi Proses Pengujian LSTM dengan Parameter Aktivasi Tanh, Optimizer Adam dengan learning rate 0,0001 ..	34

TABEL 4.13	Hasil Kinerja Evaluasi Proses Pelatihan LSTM dengan Parameter Aktivasi Relu, Optimizer RMSpropdengan learning rate 0,0001	36
TABEL 4.14	Hasil Kinerja Evaluasi Proses Pengujian LSTM dengan Parameter Aktivasi Relu, Optimizer RMSpropdengan learning rate 0,0001	36
TABEL 4.15	Hasil Kinerja Evaluasi Proses Pelatihan LSTM dengan Parameter Aktivasi Tanh, Optimizer RMSpropdengan learning rate 0,0001	37
TABEL 4.16	Hasil Kinerja Evaluasi Proses Pengujian LSTM dengan Parameter Aktivasi Tanh, Optimizer RMSpropdengan learning rate 0,0001	38
TABEL 4.17	Akurasi pengujian dari kedelapan model LSTM Menggunakan Fitur <i>Word Embedding GloVe</i>	39
TABEL 4.18	Hasil Kinerja Evaluasi Proses Pelatihan LSTM dengan Parameter Aktivasi Relu, Optimizer Adam dengan learning rate 0,001	40
TABEL 4.19	Hasil Kinerja Evaluasi Proses Pengujian LSTM dengan Parameter Aktivasi Relu, Optimizer Adam dengan learning rate 0,001	40
TABEL 4.20	Hasil Kinerja Evaluasi Proses Pelatihan LSTM dengan Parameter Aktivasi Tanh, Optimizer Adam dengan learning rate 0,001	41
TABEL 4.21	Hasil Kinerja Evaluasi Proses Pengujian LSTM dengan Parameter Aktivasi Tanh, Optimizer Adam dengan learning rate 0,001	42
TABEL 4.22	Hasil Kinerja Evaluasi Proses Pelatihan LSTM dengan Parameter Aktivasi Relu, Optimizer RMSpropdengan learning rate 0,001	43
TABEL 4.23	Hasil Kinerja Evaluasi Proses Pengujian LSTM dengan Parameter Aktivasi Relu, Optimizer RMSpropdengan learning rate 0,001	43
TABEL 4.24	Hasil Kinerja Evaluasi Proses Pelatihan LSTM dengan Parameter Aktivasi Tanh, Optimizer RMSpropdengan learning rate 0,001	44
TABEL 4.25	Hasil Kinerja Evaluasi Proses Pengujian LSTM dengan Parameter Aktivasi Tanh, Optimizer RMSpropdengan learning rate 0,001	44
TABEL 4.26	Hasil Kinerja Evaluasi Proses Pelatihan LSTM dengan Parameter Aktivasi Relu, Optimizer Adam dengan learning rate 0,0001	46
TABEL 4.27	Hasil Kinerja Evaluasi Proses Pengujian LSTM dengan Parameter AktivasiRelu, Optimizer Adam dengan learning rate 0,0001	46
TABEL 4.28	Hasil Kinerja Evaluasi Proses Pelatihan LSTM dengan Parameter Aktivasi Tanh, Optimizer Adam dengan learning rate 0,0001	47
TABEL 4.29	Hasil Kinerja Evaluasi Proses Pengujian LSTM dengan Parameter Aktivasi Tanh, Optimizer Adam dengan learning rate 0,0001	47
TABEL 4.30	Hasil Kinerja Evaluasi Proses Pelatihan LSTM dengan Parameter Aktivasi Relu, Optimizer RMSpropdengan learning rate 0,0001	49

TABEL 4.31 Hasil Kinerja Evaluasi Proses Pengujian LSTM dengan Parameter Aktivasi Relu, Optimizer RMSprop dengan learning rate 0,0001	49
TABEL 4.32 Hasil Kinerja Evaluasi Proses Pelatihan LSTM dengan Parameter Aktivasi Tanh, Optimizer RMSprop dengan learning rate 0,0001	50
TABEL 4.33 Hasil Kinerja Evaluasi Proses Pengujian LSTM dengan Parameter Aktivasi Tanh, Optimizer RMSprop dengan learning rate 0,0001	50
TABEL 4.34 Akurasi pengujian dari kedelapan model LSTM Menggunakan Fitur <i>Word Embedding</i> Word2vec	51
TABEL 4.35 <i>Confusion matrix</i> Lr = 0,001	52
TABEL 4.36 Hasil Kinerja Evaluasi Proses Pengujian LSTM dengan learning rate 0,001	53
TABEL 4.37 <i>Confusion matrix</i> Lr = 0,0001	55
TABEL 4.38 Hasil Kinerja Evaluasi Proses Pengujian LSTM dengan learning rate 0,0001	57
TABEL 4.39 Akurasi Pengujian Delapan Model LSTM	58
TABEL 4.40 Tabel Perbandingan Penelitian Sebelumnya	64

DAFTAR GAMBAR

	Hal
Gambar 2.1 Arsitektur LSTM	11
Gambar 2.2 Gerbang Forget.....	11
Gambar 2.3 Gerbang Masukan	11
Gambar 2.4 Status Sel.....	12
Gambar 2.5 Gerbang Keluaran	13
Gambar 2.6 <i>Peephole</i> LSTM.....	13
Gambar 2.7 Gated Recurrent Units (GRU)	14
Gambar 3.1 Kerangka Kerja Penelitian	17
Gambar 3.2 Tahapan Penelusuran Pustaka	18
Gambar 3.3 Kerangka Kerja Penelitian	18
Gambar 3.4 Algoritma Sistem	19
Gambar 3.5 Proses One-hot Encoding.....	20
Gambar 4.1 Kurva Perbandingan Akurasi Pelatihan dan Pengujian 50 epoch	27
Gambar 4.2 Kurva Perbandingan <i>Loss</i> Pelatihan dan Pengujian 50 epoch.....	28
Gambar 4.3 Kurva Perbandingan Akurasi Pelatihan dan Pengujian 50 epoch	29
Gambar 4.4 Kurva Perbandingan <i>Loss</i> Pelatihan dan Pengujian 50 epoch.....	29
Gambar 4.5 Kurva Perbandingan Akurasi Pelatihan dan Pengujian 50 epoch	30
Gambar 4.6 Kurva Perbandingan <i>Loss</i> Pelatihan dan Pengujian 50 epoch.....	31
Gambar 4.7 Kurva Perbandingan Akurasi Pelatihan dan Pengujian 50 epoch	32
Gambar 4.8 Kurva Perbandingan <i>Loss</i> Pelatihan dan Pengujian 50 epoch.....	32
Gambar 4.9 Kurva Perbandingan Akurasi Pelatihan dan Pengujian 50 epoch	33
Gambar 4.10 Kurva Perbandingan <i>Loss</i> Pelatihan dan Pengujian 50 epoch ...	34
Gambar 4.11 Kurva Perbandingan Akurasi Pelatihan dan Pengujian 50 epoch	35
Gambar 4.12 Kurva Perbandingan <i>Loss</i> Pelatihan dan Pengujian 50 epoch ...	35
Gambar 4.13 Kurva Perbandingan Akurasi Pelatihan dan Pengujian 50 epoch	36
Gambar 4.14 Kurva Perbandingan <i>Loss</i> Pelatihan dan Pengujian 50 epoch ...	37
Gambar 4.15 Kurva Perbandingan Akurasi Pelatihan dan Pengujian 50 epoch	38
Gambar 4.16 Kurva Perbandingan <i>Loss</i> Pelatihan dan Pengujian 50 epoch ...	38
Gambar 4.17 Kurva Perbandingan Akurasi Pelatihan dan Pengujian 50 epoch	40

Gambar 4.18	Kurva Perbandingan <i>Loss</i> Pelatihan dan Pengujian 50 epoch ...	41
Gambar 4.19	Kurva Perbandingan Akurasi Pelatihan dan Pengujian 50 epoch	42
Gambar 4.20	Kurva Perbandingan <i>Loss</i> Pelatihan dan Pengujian 50 epoch ...	42
Gambar 4.21	Kurva Perbandingan Akurasi Pelatihan dan Pengujian 50 epoch	43
Gambar 4.22	Kurva Perbandingan <i>Loss</i> Pelatihan dan Pengujian 50 epoch ...	44
Gambar 4.23	Kurva Perbandingan Akurasi Pelatihan dan Pengujian 50 epoch	45
Gambar 4.24	Kurva Perbandingan <i>Loss</i> Pelatihan dan Pengujian 50 epoch ...	45
Gambar 4.25	Kurva Perbandingan Akurasi Pelatihan dan Pengujian 50 epoch	46
Gambar 4.26	Kurva Perbandingan <i>Loss</i> Pelatihan dan Pengujian 50 epoch ...	47
Gambar 4.27	Kurva Perbandingan Akurasi Pelatihan dan Pengujian 50 epoch	48
Gambar 4.28	Kurva Perbandingan <i>Loss</i> Pelatihan dan Pengujian 50 epoch ...	48
Gambar 4.29	Kurva Perbandingan Akurasi Pelatihan dan Pengujian 50 epoch	49
Gambar 4.30	Kurva Perbandingan <i>Loss</i> Pelatihan dan Pengujian 50 epoch ...	50
Gambar 4.31	Kurva Perbandingan Akurasi Pelatihan dan Pengujian 50 epoch	51
Gambar 4.32	Kurva Perbandingan <i>Loss</i> Pelatihan dan Pengujian 50 epoch ...	51
Gambar 4.33	Kurva <i>loss</i> LSTM Optimizer Adam (a) (b) dan RMSprop(c) (d) Learning Rate 0,001	54
Gambar 4.34	Kurva <i>loss</i> LSTM Optimizer Adam (a) (b) dan RMSprop(c) (d) Learning Rate 0,001	55
Gambar 4.35	Kurva Akurasi LSTM Optimizer Adam (a) (b) dan RMSprop(c) (d) Learning Rate 0,0001	56
Gambar 4.36	Kurva <i>loss</i> LSTM Optimizer Adam (a) (b) dan RMSprop(c) (d) Learning Rate 0,0001	57
Gambar 4.37	Kurva <i>loss</i> GloVe Optimizer Adam (a) (b) dan RMSprop(c) (d) Learning Rate 0,001	59
Gambar 4.38	Kurva <i>loss</i> Word2Vec Optimizer Adam (a) (b) dan RMSprop(c) (d) Learning Rate 0,001	59
Gambar 4.39	Kurva <i>loss</i> LSTM Optimizer Adam (a) (b) dan RMSprop(c) (d) Learning Rate 0,001	60
Gambar 4.40	Kurva <i>loss</i> GloVe Optimizer Adam (a) (b) dan RMSprop(c) (d) Learning Rate 0,0001	60
Gambar 4.41	Kurva <i>loss</i> Word2Vec Optimizer Adam (a) (b) dan RMSprop(c) (d) Learning Rate 0,0001	62
Gambar 4.42	Kurva <i>loss</i> LSTM Optimizer Adam (a) (b) dan RMSprop(c) (d) Learning Rate 0,0001	62

BAB I

PENDAHULUAN

1.1 Latar Belakang

Klasifikasi teks termasuk bagian penting dalam Pemrosesan Bahasa Alami dengan banyak penerapan (Li dkk., 2018; Zhang dkk., 2015), seperti sentimen analisis (Socher dkk., 2013; Yuan dkk., 2018), pencarian informasi (Lilleberg dkk., 2015), *ranking* (Chen dkk., 2017; Acharya dkk., 2018), *indexing* (Rios dkk., 2015) dan klasifikasi dokumen (Rossi dkk., 2016; Yang dkk., 2016). Klasifikasi teks secara luas dilihat sebagai bagian *supervised learning* (Tang dkk., 2014; Tang dkk., 2015) yang diartikan sebagai identifikasi kategori dokumen baru berdasarkan probabilitas yang disarankan dengan *training corpus* khusus yang telah terlabel. Karena jumlah informasi teks yang tersedia dari dokumen baru secara *online* meningkat (Chen dkk., 2016; Pei dkk., 2018), mengelola pengklasifikasikannya dengan benar menjadi lebih sulit. Oleh sebab itu, kemampuan untuk mengambil kategori yang benar untuk dokumen baru sangat bergantung pada jumlah dokumen berlabel yang sudah tersedia untuk referensi (Yogatama dkk., 2017).

Model klasifikasi teks umumnya dibagi menjadi dua kategori: *machine learning* dan *deep learning*. Banyak penelitian pada klasifikasi teks telah melibatkan algoritma tradisional *machine learning* seperti *k-Nearest Neighbors* (Pratama & Sarno, 2015; Azam dkk., 2018), *Naive Bayes* (Jiang, L dkk 2016; Xu, 2016), *Support Vector Machine* (Fanjin dkk., 2017; Goudjil dkk., 2016), *Logistic Regression* (Onan dkk., 2016). Selain itu, dibandingkan dengan algoritma klasifikasi tradisional *machine learning* memiliki karakteristik efisiensi dan stabilitas tinggi. Namun, memiliki batasan tertentu dalam hal pelatihan *Dataset* skala besar (Gao dkk., 2019).

Baru-baru ini, model berbasis jaringan saraf menjadi semakin populer (Kim, 2014; Zhang, Zhao, & LeCun, 2015; Conneau dkk., 2017; Kowsari dkk 2017). Model-model ini mencapai kinerja yang sangat baik dalam praktiknya,

cenderung relatif lambat baik pada saat *training* maupun *testing*, membatasi penggunaannya pada *Dataset* yang sangat besar (Joulin dkk., 2016). Beberapa penelitian terbaru menunjukkan bahwa keberhasilan *deep learning* tentang klasifikasi teks sangat tergantung pada efektivitas *word embedding* (Wieting dkk., 2016; Arora dkk., 2017; Shen dkk., 2018). Khususnya, (Shen dkk. 2018) secara kuantitatif menunjukkan bahwa tugas klasifikasi teks berbasis *word embedding* dapat memiliki tingkat kesulitan yang sama terlepas dari model yang digunakan, menggunakan konsep dimensi intrinsik (Li dkk., 2018). Bagaimanapun, model sederhana lebih dipilih. Sebagai blok bangunan dasar dalam NLP berbasis jaringan syaraf, *word embedding* menangkap kesamaan antara kata-kata (Mikolov dkk., 2013; Pennington dkk., 2014). Gagasan ini telah diperluas untuk menghitung *embeddings* yang menangkap semantik dari urutan kata seperti Frasa, kalimat, paragraf, dan dokumen (Le & Mikolov, 2014).

Beberapa contoh metode *deep learning* yang digunakan untuk pengklasifikasian teks antara lain *convolutional neural network* (Kim, 2014; Zhang dkk., 2015; Bai dkk., 2017), *autoecoder* (Xu dkk., 2017; Soares, 2018), *deep belief network* (Jiang dkk., 2016; Ruangkanokmas dkk., 2016). *Recurrent Neural Network* (RNN) merupakan salah satu arsitektur yang paling populer yang digunakan dalam pemrosesan bahasa alami (NLP) karena struktur *recurrent* cocok untuk proses teks variabel panjang. Salah satu metode *deep learning* yang diusulkan pada penelitian ini adalah RNN dengan penerapan arsitektur *Long Short-Term Memory* (LSTM). RNN dapat menggunakan representasi kata yang terdistribusi dengan terlebih dahulu mengubah token yang terdiri dari setiap teks menjadi vektor, yang membentuk matriks. Sedangkan, LSTM dikembangkan untuk menangani masalah *exploding* dan *vanishing gradient* yang dapat dihadapi saat melatih RNN tradisional (Yan, dkk., 2017). Selain untuk memperluas memori, pengklasifikasian teks menggunakan LSTM pada penelitian ini karena struktur dari LSTM adalah *sequence* dimana suatu kesatuan utuh atau tidak bisa dipotong-potong sama halnya dengan struktur dokumen teks yang jika dipotong akan merubah arti dari kalimat tersebut. Penggunaan *word embedding* akan menjadifitur masukan (*input*) pada LSTM sebelum mengklasifikasikan teks.

1.2 Rumusan Masalah

Berdasarkan latar belakang untuk membuktikan kinerja LSTM dengan fitur tambahan *word embedding* dalam pengklasifikasian teks, maka pertanyaan penelitian dapat dirumuskan sebagai berikut:

1. Bagaimana pengaruh fitur *word embedding* GloVe dan Word2vec pada kinerja klasifikasi LSTM?
2. Bagaimana menentukan parameter LSTM yang optimal untuk pengklasifikasian teks?
3. Bagaimana penggunaan fitur tambahan dapat meningkatkan kinerja pengklasifikasian LSTM berdasarkan tingkat akurasi, presisi, recall dan *F1-score*?

1.3 Tujuan Penelitian

Tujuan penelitian ini adalah sebagai berikut:

1. Meningkatkan performa klasifikasi LSTM dengan *word embedding* GloVe dan Word2vec sebagai fitur masukan.
2. Melakukan percobaan *trial* dan *error* parameter dasar LSTM yang kemudian dilakukan *tuning* parameter untuk mendapatkan parameter yang cocok.
3. Mengukur kinerja klasifikasi LSTM berdasarkan pemodelan yang telah didapat sehingga menghasilkan nilai *confusion matrix* yang akan menentukan nilai akurasi, presisi, recall, dan *f1 score*.

1.4 Manfaat Penelitian

Hasil penelitian ini dapat menjadi landasan dalam pengembangan klasifikasi teks menggunakan LSTM dengan fitur *word embedding* secara lebih lanjut. Selain itu manfaat dari penelitian ini secara praktis yaitu sebagai berikut:

1. Pengembangan LSTM dengan fitur GloVe dan Word2vec dapat menghasilkan kinerja metrik dan kurva *loss* dari proses *training* dan *testing* yang lebih optimal.

2. Hasil penelitian ini dapat menjadi referensi untuk meningkatkan nilai performa akurasi, presisi, recall, *f1 score* dalam klasifikasi teks yang menerapkan LSTM dengan fitur GloVe dan Word2vec.
3. Dengan jumlah dokumen yang sangat besar, klasifikasi teks digunakan untuk mengorganisir dokumen agar lebih terstruktur dan memudahkan mencari sebuah dokumen yang telah dikelompokkan sesuai kategorinya masing-masing.

1.5 Batasan Masalah

Batasan masalah dalam pengklasifikasian teks yang dirancang pada penelitian ini adalah:

1. *Dataset* yang digunakan merupakan *Dataset* publik yaitu AGNews.
2. Arsitektur Recurrent Neural Network adalah *Long Short-Term Memory* (LSTM) dan *Gated Recurrent Unit* (GRU), pada penelitian ini menggunakan arsitektur LSTM.

1.6 Metodologi Penulisan

Metodologi penulisan pada penelitian ini terdiri dari tiga bab sebagai berikut:

BAB I : PENDAHULUAN

Bab I berisi tentang latar belakang masalah, rumusan masalah, batasan masalah, tujuan dan manfaat dari topik yang dipilih berupa pengklasifikasian teks dengan LSTM dengan fitur *word embedding* Word2Vec dan GloVe.

BAB II : TINJAUAN PUSTAKA

Bab II berisi kerangka teori dan pustaka yang berhubungan dengan pengklasifikasian teks dengan LSTM dengan fitur *word embedding* Word2Vec dan GloVe berdasarkan beberapa penelitian jurnal publikasi.

BAB III : METODOLOGI PENELITIAN

Bab III berisi metodologi yang menjelaskan secara bertahap dan terperinci tentang langkah-langkah yang digunakan untuk mencari, mengumpulkan dan menganalisa terkait pengklasifikasian teks dengan LSTM dengan fitur *word embedding* Word2Vec dan GloVe, serta model yang digunakan sehingga tujuan dari penulisan dapat tercapai.

BAB IV : HASIL DAN ANALISA SEMENTARA

Bab IV berisi hasil pengujian yang telah dilakukan, data-data yang diambil dari pengujian tersebut akan dianalisa menggunakan berbagai macam teknik, selain itu di bab ini juga membahas kevalidasian dari sistem yang telah dibuat.

BAB V : KESIMPULAN DAN SARAN

BAB V berisi tentang kesimpulan dan saran apa yang diperoleh oleh penulis serta merupakan jawaban dari setiap tujuan dan manfaat yang ingin dicapai.

DAFTAR PUSTAKA

- Acharya, V., Engle, R., & Richardson, M. (2012). Capital shortfall: A new approach to ranking and regulating systemic risks. *American Economic Review*, 102(3), 59–64. <https://doi.org/10.1257/aer.102.3.59>.
- Arora, S., Liang, Y., & Ma, T. (2017). A Simple but Tough-to-Beat Baseline for Sentence Embeddings. *Iclr*, 15, 416–424. Retrieved from <https://openreview.net/forum?id=SyK00v5xx%0Ahttps://github.com/PrincetonML/SIF>.
- Azam, M., Ahmed, T., Sabah, F., & Hussain, M. I. (2018). Feature Extraction based Text Classification using K-Nearest Neighbor Algorithm. *IJCSNS International Journal of Computer Science and Network Security*, 18(12), 95–101. http://paper.ijcsns.org/07_book/201812/20181213.pdf.
- Bai, X., Shi, B., Zhang, C., Cai, X., & Qi, L. (2017). Text/non-text image classification in the wild with convolutional neural networks. *Pattern Recognition*, 66, 437–446. <https://doi.org/10.1016/j.patcog.2016.12.005>.
- Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3(4–5), 993–1022. <https://doi.org/10.1016/b978-0-12-411519-4.00006-9>.
- Carrera-trejo, V., Sidorov, G., Miranda-jiménez, S., Ibarra, M. M., & Martínez, R. C. (2015). Latent Dirichlet Allocation complement in the vector space model for Multi-Label Text Classification. *International Journal of Combinatorial Optimization Problems and Informatics*, 6(1), 7–19.
- Chen, K., Zhang, Z., Long, J., & Zhang, H. (2016). Turning from TF-IDF to TF-IGM for term weighting in text classification. *Expert Systems with Applications*, 66, 245–260. <https://doi.org/10.1016/j.eswa.2016.09.009>.
- Chiu, C., Sainath, T. N., Wu, Y., Prabhavalkar, R., Nguyen, P., Chen, Z., ... Bacchiani, M. (2018). State-Of-The-Art Speech Recognition With Sequence-To-Sequence Models. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 4774–4778. <https://doi.org/10.1109/ICASSP.2018.8462105>.
- Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. *EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, 1724–1734. <https://doi.org/10.3115/v1/d14-1179>.
- Choi, K., Fazekas, G., & Sandler, M. K. C. (2017). Convolutional recurrent neural networks for music classification. *IEEE International Conference on*

- Acoustics, Speech and Signal Processing (ICASSP).
<https://doi.org/10.1109/ICASSP.2017.7952585>.
- Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. 1–9. Retrieved from <http://arxiv.org/abs/1412.3555>.
- Conneau, A., Schwenk, H., Barrault, L., & Lecun, Y. (2016). Very Deep Convolutional Neural Networks for Text Classification. ArXiv Preprint ArXiv:1606.01781 (2016)., 11727 LNCS, 193–207. https://doi.org/10.1007/978-3-030-30487-4_16.
- Diaz, F., Mitra, B., & Craswell, N. (2016). Query expansion with locally-trained word embeddings. 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016 - Long Papers, 1, 367–377. <https://doi.org/10.18653/v1/p16-1035>.
- Fanjin, M., Ling, H., Jing, T., & Wang, X. (2017). The research of semantic kernel in SVM for Chinese text classification. ACM International Conference Proceeding Series, Part F1318(319). <https://doi.org/10.1145/3144789.3144801>.
- Gao, M., Li, T., & Huang, P. (2019). Text Classification Research Based on Improved Word2vec and CNN. International Conference on Service-Oriented Computing, 1(645751), 367–373. <https://doi.org/10.1007/978-3-030-17642-6>.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. Pp. 429. MIT press.
- Goudjil, M., Koudil, M., Bedda, M., & Ghoggali, N. (2018). A Novel Active Learning Method Using SVM for Text Classification. International Journal of Automation and Computing, 15(3), 290–298. <https://doi.org/10.1007/s11633-015-0912-z>.
- Graves, A. (2013). Generating Sequences With Recurrent Neural Networks. Neural and Evolutionary Computing, 1–43. Retrieved from <http://arxiv.org/abs/1308.0850>.
- Heidarysafa, M., Kowsari, K., Brown, D. E., Meimandi, K. J., & Barnes, L. E. (2018). An Improvement of Data Classification Using Random Multimodel *Deep learning* (RMDL). International Journal of *Machine learning* and Computing, 8(4), 298–310. <https://doi.org/10.18178/ijmlc.2018.8.4.703>
- Hingmire, S., Chougule, S., Palshikar, G. K., & Chakraborti, S. (2013). Document classification by topic labeling. SIGIR 2013 - Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval, 877–880. <https://doi.org/10.1145/2484028.2484140>

- Jiang, L., Li, C., Wang, S., & Zhang, L. (2016). Deep feature weighting for naive Bayes and its application to text classification. *Engineering Applications of Artificial Intelligence*, 52, 26–39. <https://doi.org/10.1016/j.engappai.2016.02.002>
- Joulin, A., Grave, E., Bojanowski, P., & Mikolov, T. (2017). Bag of tricks for efficient text classification. 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017 - Proceedings of Conference, 2, 427–431. <https://doi.org/10.18653/v1/e17-2068>
- Kim, H. K., Kim, H., & Cho, S. (2017). Bag-of-concepts: Comprehending document representation through clustering words in distributed representation. *Neurocomputing*, 266, 336–352. <https://doi.org/10.1016/j.neucom.2017.05.046>
- Kim, Y. (2014). Convolutional neural networks for sentence classification. EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference, 1746–1751. <https://doi.org/10.3115/v1/d14-1181>
- Kowsari, K., Brown, D. E., Heidarysafa, M., Jafari Meimandi, K., Gerber, M. S., & Barnes, L. E. (2017). HDLTex: Hierarchical *Deep learning* for Text Classification. Proceedings - 16th IEEE International Conference on *Machine learning* and Applications, ICMLA 2017, 2017-Decem, 364–371. <https://doi.org/10.1109/ICMLA.2017.0-134>
- Kumar, A., & Khemchandani, R. R. (nee.(2019). Attentional Recurrent Neural Networks for Sentence Classification. In *Advances in Intelligent Systems and Computing* (Vol. 757). <https://doi.org/10.1007/978-981-13-1966-2>
- Le, Q., & Mikolov, T. (2014). Distributed representations of sentences and documents. International Conference on *Machine learning*, 1188–1196.
- Li, L., Xiao, L., Jin, W., Zhu, H., & Yang, G. (2018). Text Classification Based on Word2vec and Convolutional Neural Network. International Conference on Neural Information Processing, 3, 450–460. <https://doi.org/10.1007/978-3-030-04221-9>
- Li, W., Zhu, X., & Gong, S. (2017). Person re-identification by deep joint learning of multi-loss classification. IJCAI International Joint Conference on Artificial Intelligence, 2194–2200. <https://doi.org/10.24963/ijcai.2017/305>
- Lilleberg, J., Zhu, Y., & Zhang, Y. (2015). Support vector machines and Word2vec for text classification with semantic features. Proceedings of 2015 IEEE 14th International Conference on Cognitive Informatics and Cognitive Computing, ICCI*CC 2015, 136–140. <https://doi.org/10.1109/ICCI-CC.2015.7259377>

- Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Dean, J. (2013). Distributed Representations of Words and Phrases and their Compositionality. In *Advances in Neural Information Processing Systems*, 3111–3119. <https://doi.org/10.18653/v1/d16-1146>
- Novaković, J., Strbac, P., & Bulatović, D. (2011). Toward optimal feature selection using ranking methods and classification algorithms. *Yugoslav Journal of Operations Research*, 21(1), 119–135. <https://doi.org/10.2298/YJOR1101119N>
- Onan, A. (2018). An ensemble scheme based on language function analysis and feature engineering for text genre classification. *Journal of Information Science*, 44(1), 28–47. <https://doi.org/10.1177/0165551516677911>
- Pei, K., Chen, Y., Ma, J., & Nie, W. (2018). Short Text Classification Research Based on TW-CNN. *PACIS 2018 Proceedings*. Retrieved from <https://aisel.aisnet.org/pacis2018/41>
- Pennington, J., & Socher, R. (1971). Glove: Global vectors for word representation. *AES: Journal of the Audio Engineering Society*, 19(5), 417–425.
- Pota, M., Marulli, F., Esposito, M., De Pietro, G., & Fujita, H. (2019). Multilingual POS tagging by a composite deep architecture based on character-level features and on-the-fly enriched Word Embeddings. *Knowledge-Based Systems*, 164(xxxx), 309–323. <https://doi.org/10.1016/j.knosys.2018.11.003>
- Pratama, B. Y., & Sarno, R. (2016). Personality classification based on Twitter text using Naive Bayes, KNN and SVM. *Proceedings of 2015 International Conference on Data and Software Engineering, ICODSE 2015*, 170–174. <https://doi.org/10.1109/ICODSE.2015.7436992>
- Qu, Z., Song, X., Zheng, S., Wang, X., Song, X., & Li, Z. (2018). Improved Bayes Method Based on TF-IDF Feature and Grade Factor Feature for Chinese Information Classification. *Proceedings - 2018 IEEE International Conference on Big Data and Smart Computing, BigComp 2018*, 677–680. <https://doi.org/10.1109/BigComp.2018.00124>
- Rahhal, M. M. A., Bazi, Y., Alhichri, H., Alajlan, N., Melgani, F., & Yager, R. R. (2016). *Deep learning* approach for active classification of electrocardiogram signals. *Information Sciences*, 345, 340–354. <https://doi.org/10.1016/j.ins.2016.01.082>
- Rossi, R. G., Lopes, A. D. A., & Rezende, S. O. (2016). Optimization and label propagation in bipartite heterogeneous networks to improve transductive classification of texts. *Information Processing and Management*, 52(2), 217–257. <https://doi.org/10.1016/j.ipm.2015.07.004>

- Ruangkanokmas, P., Achalakul, T., & Akkarajitsakul, K. (2017). Deep Belief Networks with Feature Selection for Sentiment Classification. Proceedings - International Conference on Intelligent Systems, Modelling and Simulation, ISMS, 0, 9–14. <https://doi.org/10.1109/ISMS.2016.9>
- Shao, Y., Taylor, S., Marshall, N., Morioka, C., & Zeng-Treitler, Q. (2019). Clinical Text Classification with Word *Embedding* Features vs. Bag-of-Words Features. Proceedings - 2018 IEEE International Conference on Big Data, Big Data 2018, 2874–2878. <https://doi.org/10.1109/BigData.2018.8622345>
- Shen, D., Wang, G., Wang, W., Min, M. R., Su, Q., Zhang, Y., ... Carin, L. (2018). Baseline Needs More Love: On Simple Word-Embedding-Based Models and Associated Pooling Mechanisms. ACL 2018 - 56th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers), 1, 440–450. <https://doi.org/10.18653/v1/p18-1041>
- Soares, R. G. F. (2018). Effort Estimation via Text Classification and Autoencoders. Proceedings of the International Joint Conference on Neural Networks, 2018-July, 1–8. <https://doi.org/10.1109/IJCNN.2018.8489030>
- Socher, R., Perelygin, A., Wu, J. Y., Chuang, J., Manning, C. D., Andrew Y. Ng, A., & Potts, C. (2013). Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank. Empirical Methods in Natural Language Processing, (October), 1631–1642. <https://doi.org/10.1371/journal.pone.0073791>
- Stein, R. A., Jaques, P. A., & Valiati, J. F. (2019). An analysis of hierarchical text classification using word embeddings. Information Sciences, 471, 216–232. <https://doi.org/10.1016/j.ins.2018.09.001>
- Sun, F., & Chen, H. (2018). Feature extension for Chinese short text classification based on LDA and Word2vec. Proceedings of the 13th IEEE Conference on Industrial Electronics and Applications, ICIEA 2018, (1), 1189–1194. <https://doi.org/10.1109/ICIEA.2018.8397890>
- Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. Advances in Neural Information Processing Systems, 4(January), 3104–3112.
- Tang, B., He, H., Baggenstoss, P. M., & Kay, S. (2016). A Bayesian Classification Approach Using Class-Specific Features for Text Categorization. IEEE Transactions on Knowledge and Data Engineering, 28(6), 1602–1606. <https://doi.org/10.1109/TKDE.2016.2522427>
- Tang, D., Qin, B., & Liu, T. (2015). Learning semantic representations of users and products for document level sentiment classification. ACL-IJCNLP 2015 - 53rd Annual Meeting of the Association for Computational

- Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, Proceedings of the Conference, 1, 1014–1023. <https://doi.org/10.3115/v1/p15-1098>
- Tang, D., Wei, F., Yang, N., Zhou, M., Liu, T., & Qin, B. (2014). Learning Sentiment-Specific Word Embedding. *Acl*, 1555–1565. <https://doi.org/10.3115/1220575.1220648>
- Tseng, K. K., Ou, C., Huang, A., Lin, R. F. Y., & Guo, X. (2019). Genetic and Evolutionary Computing. *Advances in Intelligent Systems and Computing*, 834, 545–552. https://doi.org/10.1007/978-981-13-5841-8_57
- Uysal, A. K. (2016). An improved global feature selection scheme for text classification. *Expert Systems with Applications*, 43, 82–92. <https://doi.org/10.1016/j.eswa.2015.08.050>
- Wang, G., Li, C., Wang, W., Zhang, Y., Shen, D., Zhang, X., ... Carin, L. (2018). Joint *embedding* of words and labels for text classification. *ACL 2018 - 56th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers)*, 1, 2321–2331. <https://doi.org/10.18653/v1/p18-1216>
- Wiatowski, T., & Bolcskei, H. (2018). A Mathematical Theory of Deep Convolutional Neural Networks for Feature Extraction. *IEEE Transactions on Information Theory*, 64(3), 1845–1866. <https://doi.org/10.1109/TIT.2017.2776228>
- Wieting, J., Bansal, M., Gimpel, K., & Livescu, K. (2016). Charagram: *Embedding* words and sentences via character n-grams. *EMNLP 2016 - Conference on Empirical Methods in Natural Language Processing, Proceedings*, 1504–1515. <https://doi.org/10.18653/v1/d16-1157>
- Xing, C., Wang, D., Zhang, X., & Liu, C. (2014). Document classification with distributions of word vectors. *2014 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, APSIPA 2014*. <https://doi.org/10.1109/APSIPA.2014.7041633>
- Xu, H., Kotov, A., Dong, M., Carcone, A. I., Zhu, D., & Naar-King, S. (2016). Text classification with topic-based word *embedding* and Convolutional Neural Networks. *ACM-BCB 2016 - 7th ACM Conference on Bioinformatics, Computational Biology, and Health Informatics*, 88–97. <https://doi.org/10.1145/2975167.2975176>
- Xu, S. (2018). Bayesian Naïve Bayes classifiers to text classification. *Journal of Information Science*, 44(1), 48–59. <https://doi.org/10.1177/0165551516677946>

- Xu, W., Sun, H., Deng, C., & Tan, Y. (2017). Variational autoencoder for semi-supervised text classification. 31st AAAI Conference on Artificial Intelligence, AAAI 2017, 3358–3364.
- Xu, Y. M., Wang, C. D., & Lai, J. H. (2016). Weighted Multi-view Clustering with Feature Selection. *Pattern Recognition*, 53, 25–35. <https://doi.org/10.1016/j.patcog.2015.12.007>
- Xue, B., Fu, C., & Shaobin, Z. (2014). A study on sentiment computing and classification of sina weibo with Word2vec. *Proceedings - 2014 IEEE International Congress on Big Data, BigData Congress 2014*, (2013), 358–363. <https://doi.org/10.1109/BigData.Congress.2014.59>
- Yan, Y., Wang, Y., Gao, W. C., Zhang, B. W., Yang, C., & Yin, X. C. (2018). LSTM2: Multi-Label Ranking for Document Classification. *Neural Processing Letters*, 47(1), 117–138. <https://doi.org/10.1007/s11063-017-9636-0>
- Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., & Hovy, E. (2016). Hierarchical Attention Networks for Document Classification. *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 1480–1489. Retrieved from <http://arxiv.org/abs/1606.02393>
- Yogatama, D., Dyer, C., Ling, W., & Blunsom, P. (2017). Generative and Discriminative Text Classification with Recurrent Neural Networks. *arxiv:1703.01898*.
- Yuan, H., Wang, Y., Feng, X., & Sun, S. (2018). Sentiment analysis based on weighted word2vec and ATT-LSTM. *ACM International Conference Proceeding Series*, 420–424. <https://doi.org/10.1145/3297156.3297228>
- Zen, H., & Sak, H. (2015). Unidirectional long short-term memory recurrent neural network with recurrent output layer for low-latency speech synthesis. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2015-Augus, 4470–4474. <https://doi.org/10.1109/ICASSP.2015.7178816>
- Zhang, D., Xu, H., Su, Z., & Xu, Y. (2015). Chinese comments sentiment classification based on word2vec and SVMperf. *Expert Systems with Applications*, 42(4), 1857–1863. <https://doi.org/10.1016/j.eswa.2014.09.011>
- Zhang, X., & LeCun, Y. (2015). Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems*, 1–9. Retrieved from <http://arxiv.org/abs/1502.01710>
- Zhang, Y., Jin, R., & Zhou, Z. H. (2010). Understanding bag-of-words model: A statistical framework. *International Journal of Machine Learning and Cybernetics*, 1(1–4), 43–52. <https://doi.org/10.1007/s13042-010-0001-0>

- Zheng, X., Chen, H., & Xu, T. (2013). *Deep learning* for Chinese word segmentation and POS tagging. EMNLP 2013 - 2013 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference, (October), 647–657.
- Zou, Q., Ni, L., Zhang, T., & Wang, Q. (2015). *Deep learning* Based Feature Selection for Remote Sensing Scene Classification. IEEE Transactions on Geoscience and Remote Sensing Letters, 12(11), 2321–2325. <https://doi.org/10.1109/LGRS.2015.2475299>.