MODEL RANGKAIAN NEURAL BAGI PENANDAAN GOLONGAN KATA PADA TEKS MEDIA SOSIAL BAHASA MELAYU

Chew Yee Dhong

Sabrina Tiun

Fakulti Teknologi & Sains Maklumat, Universiti Kebangsaan Malaysia

ABSTRAK

Penandaan golongan kata (PGK) memainkan peranan penting dalam pemprosesan Bahasa Tabii(NLP). Berbanding dengan teks kontekstual biasa, teks dalam media sosial biasanya mengandungi bahasa formal dan tidak formal. Masalah seterusnya ialah prestasi ketepatan PGK bagi teks media sosial Bahasa Melayu pada kajian sedia ada masih rendah bagi set data yang berskala kecil. Oleh itu, tujuan kajian ini adalah untuk membina sebuah model penandaan golongan kata yang baik untuk teks media sosial Bahasa Melayu dan meningkatkan prestasi ketepatan sistem PGK media sosial bahasa Melayu yang sedia ada. Dalam projek ini, domain kajian ialah penandaan golongan kata bagi teks aplikasi *Twitter*. Sebanyak 45 jenis PGK digunakan dalam kajian ini di mana 10 PGK untuk *twitter*. Sebuah model yang bernama PGK Bi-LSTM-CRF dibangunkan dalam kajian ini. PGK Bi-LSTM-CRF menggunakan Rangkaian Memori Jangka Pendek dua arah (Bi-LSTM) dengan Medan Rawak Bersyarat (CRF). Perbandingan antara dua kaedah penyisipan perkataan iaitu (1) Word2Vec pra-latihan dan (2) penyisipan perkataan secara rawak Keras dilaksana dalam kajian ini. Prestasi sistem dinilai dengan ketepatan dan ukuran-f mikro. Model Bi-LSTM-CRF dengan penyisipan Word2Vec pra-latihan menghasilkan prestasi terbaik iaitu sebanyak ukuran-f mikro 94% dan ketepatan 93.81% didapati. Oleh itu dapat disimpulkan penggunaan kaedah pembelajaran mendalam dengan fitur penyisipan yang sesuai boleh mempertingkatkan kecekapan penandaan golongan kata pada teks bahasa Melayu media sosial.

1 PENGENALAN

Kandungan di media sosial berlain-lain tetapi biasanya berkait dengan berita, pendapat, perasaan dan pandangan individu terhadap benda yang berlainan. Data ini dipanggil sebagai data media sosial dan setiap hari data ini dihasilkan dengan jumlah besar. Berbanding dengan teks kontekstual biasa, teks dalam media sosial berbentuk tidak berstruktur. Teks dalam media sosial tidak mengikuti tatabahasa, huruf besar dalam teks digunakan dengan sesuka hati dan banyak ruang antara perkataan. Pengguna juga mencipta bahasa sendiri yang pelik dalam media sosial. Walaupun data media sosial tidak berstruktur, banyak maklumat penting bagi pelbagai aplikasi dapat diperolehi dalam data tersebut.

Setiap perkataan mempunyai kegunaan dan fungsinya dalam sesebuah ayat. Penandaan golongan kata (PGK) adalah untuk mengelaskan perkataan berdasarkan kegunaan dan fungsi. PGK memainkan peranan yang penting dalam pemprosesan Bahasa Tabii. Di era revolusi industri ke-4, PGK juga terlibat dalam teknologi tinggi seperti kereta dan rumah pintar yang dikawal dengan perintah suara manusia (Shamsan, Nazri, Nazlia& Salwani, 2020).

Rangkaian neural (NN) merupakan trend sekarang terutamanya dalam bidang kecerdasan buatan. Rangkaian saraf berulang (RNN) merupakan salah satu contoh NN. RNN mampu mengingati semua input yang terdahulu dan hal ini membantu RNN meramalkan data seterusnya dengan tepat. Oleh itu, algoritma ini sesuai kepada data berurutan seperti pertuturan, teks, cuaca dan sebagainya. Namun begitu, RNN tidak dapat mengingat data yang sangat lama (Niklas, 2019). Kombinasi dengan Memori Jangka Pendek Panjang (LSTM), algoritma ini dapat mengatasi kelemahannya dan mempunyai ingatan yang jangka panjang. LSTM juga mempunyai pintu yang boleh dilatih supaya LSTM boleh memutuskan input menambah ke ingatan dan membuat output. Selain daripada LSTM, LSTM dua arah (Bi-LSTM) bergantung kepada langkah-langkah masa depan dan masa lalu dalam urutan. Bi-LSTM amat sesuai bagi teks sosial media kerana makna perkataan dalam sesebuah ayat tidak hanya bergantung pada nod sebelumnya tetapi juga bergantung kepada nod seterusnya.

RNN telah banyak digunakan dalam pemprosesan bahasa (NLP). Pada tahun 2017, sebuah kajian tentang RNN yang dibuat oleh Kumar, Anand dan Soman dengan

membuat perbandingan antara RNN, GRU, LSTM dan Bi-LSTM bagi mendapat kaedah yang paling sesuai untuk PGK Twitter Malayalam. Dalam kajian mereka, Bi-LSTM mendapat ketepatan yang paling tinggi iaitu 87.39%. Pada tahun 2015, Huang, Wei dan Kai memperkenalkan sebuah model yang bernama Conv-CRF untuk mengkaji ketepatan antara LSTM, Bi-LSTM, LSTM dengan lapisan medan rawak bersyarat (CRF) dan Bi-LSTM dengan lapisan CRF terhadap PGK, pengecaman entiti nama (NER) dan *chunking*. Bi-LSTM-CRF model mendapat ketepatan yang paling tinggi dalam kajian mereka.

Objektif kajian ini adalah untuk membina sebuah model PGK yang sesuai digunakan pada teks media sosial Melayu. Set data dalam kajian ini dilatih dengan kaedah RNN kerana urutan penting dalam permasalahan PGK teks media sosial. Apabila kita memahami makna sesebuah teks, tidak cukup hanya memahami perkataan secara bersaingan. Kita perlu menangani keseluruhan urutan kata-kata dalam ayat tersebut. Perkataan sebelumnya mempunyai pengaruh besar terhadap ramalan PGK perkataan semasa. Bagi memproses data urutan seperti teks media sosial Melayu, RNN merupakan pilihan yang baik.

Terdapat banyak sistem PGK yang dibina berdasarkan Bi-LSTM dengan CRF, tetapi kaedah ini bagi PGK Bahasa Melayu teks media sosial setakat ini belum lagi dibangunkan. Bi-LSTM menggunakan tag masa lalu dan masa depan untuk meramalkan tag semasa dengan cekap. Namun begitu, hubungan antara label bersebelahan amat penting dalam pelabelan urutan. Oleh itu, penggunaan teknik CRF juga digunakan dalam kajian ini.

2 PENYATAAN MASALAH

Walaupun terdapat kajian PGK bagi teks media sosial Melayu sekarang seperti kajian yang dibuat oleh Siti dan Sabrina pada tahun 2018, PGK Bahasa Melayu untuk teks media sosial masih mempunyai ruang untuk diperbaiki.

QTAG Bahasa Melayu (Siti dan Sabrina, 2018) iaitu PGK teks media sosial Bahasa Melayu mempunyai ketepatan masih rendah bagi skala data yang kecil. Walaupun QTAG Melayu mendapat ketepatan keseluruhan sebanyak 88.8%, namun hanya kejituan dicapai sebanyak 72.7% bagi set data yang berskala kecil.

3 OBJEKTIF KAJIAN

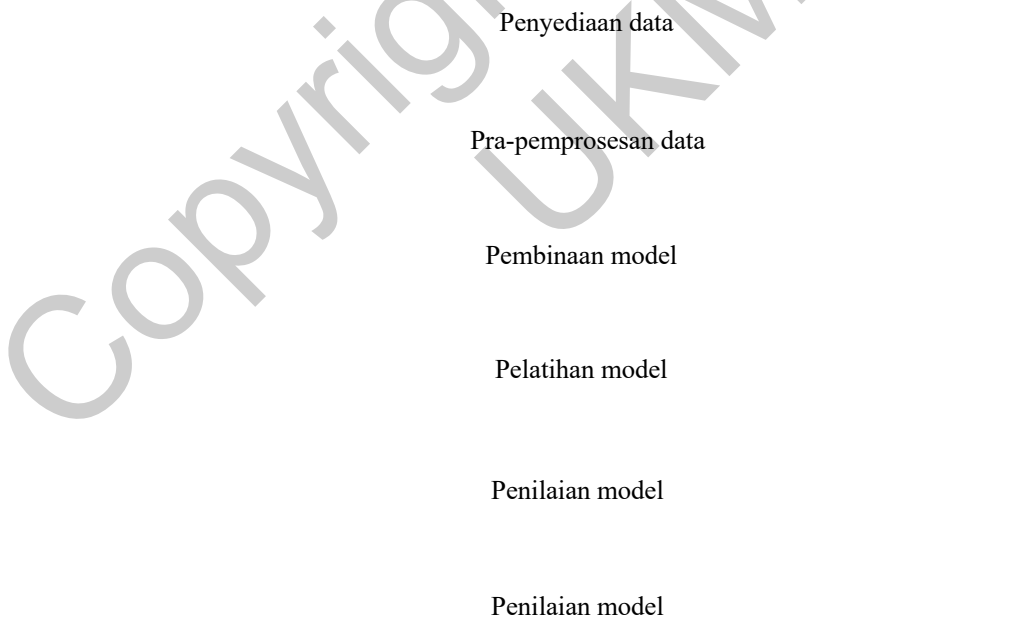
Berdasarkan cadangan penyelesaian bagi kajian ini, beberapa objektif kajian yang akan dicapai adalah:

I. Membangunkan sebuah sistem PGK teks media sosial Bahasa Melayu yang lebih baik dengan menggunakan kaedah RNN dengan CRF.

II. Membandingkan prestasi sistem PGK teks media sosial Bahasa Melayu yang dibangunkan dengan sistem PGK yang terdahulu.

4 METOD KAJIAN

Fasa ini bertujuan untuk menyatakan rangka kerja dan seni bina yang akan digunakan dalam pembangunan sistem model yang bernama PGK BI-LSTM-CRF. Fasa pembangunan termasuk fasa penyediaan data, pra-pemprosesan data, pembinaan model, pelatihan model, penilaian model dan penilaian model. Rajah 1 menunjuk carta aliran sistem PGK.



Rajah 1 Carta aliran sistem PGK Bi-LSTM-CRF

4.1 PENYEDIAAN DATA

Korpus kajian ini adalah *Twitter* dan sebanyak 500 *tweet* digunakan. Kajian ini menggunakan set data daripada kajian lepas Siti dan Sabrina (2018). Set data ini telah ditandakan dengan PGK secara manual menggunakan PGK dari karya kajian sebelumnya. Data ini akan dibahagikan menjadi dua iaitu 80% bagi data latih yang untuk pembuatan model dan 20% bagi data kajian untuk menguji prestasi model. Jadual 1 menunjukkan contoh *tweet* yang telah ditandakan PGK. Sebanyak 45 PGK diguna dalam kajian ini di mana 10 PGK untuk *tweet*. Penandaan golongan kata yang diguna dalam kajian ini ditunjuk dalam jadual 2.

Jadual 1 Perbandingan antara parameter yang berlainan

No.	***Tweet***	**Hasil**
1	ah sakit pula tiba	ah/KSR sakit/KA pula/KAD tiba/KAD
2	ada seekor ular di dalam perigi	ada/KK seekor/KBIL ular/KN di/KS dalam/KS perigi/KN
3	Aku nak main bola jangan bising	Aku/GN1 nak/KEP main/KK bola/KN jangan/KPE bising/KA

Jadual 2 Jenis label PGK dengan perkataan contoh

PGK	**Nama PGK**	**Contoh**
AWL	Kata Awalan	Anti, semi
BY	Bunyi	Huhu, hehe
GN1	Kata Ganti Nama Diri Pertama	Saya, aku, kami
GN1-LD	Kata Ganti Nama Diri Pertama- Bahasa Tempatan	Den
GN2	Kata Ganti Nama Diri Kedua	Kamu, kau
GN2-LD	Kata Ganti Nama Diri Kedua- Bahasa Tempatan	hangpa
GN3	Kata Ganti Nama Diri Ketiga	Dia, beliau
GN3-LD	Kata Ganti Nama Diri Ketiga- Bahasa Tempatan	dema
GT	Kata Ganti Nama Tunjuk	Itu, ini, sini
GT-KEP	Kata Ganti Nama Tunjuk- Kata Singkat	Tu, ni
GDT	Kata Ganti Nama Tunjuk Tempat	Apakah, siapakah
GDT-KTY	Kata Ganti Nama Tunjuk Tempat- Kata Tanya	mana
GL	Kata Ganti Nama Diri Laras Bahasa Istana	Hang, baginda
KN	Kata Nama	Ali, Abu, sungai
KN-LD	Kata Nama- Bahasa Tempatan	Ambo, okemo
KN-KEP	Kata Nama-Kata Singkat	laki
KK	Kata Kerja	Jalan, lari, ada
KA	Kata Adjektif	Cantik, sabar
KA-KEP	Kata Adjektif – Kata Singkat	kat
KH	Kata Hubung	Sebab, yang, kalau
KH-KEP	Kata Hubung- Kata Singkat	tapi
KB	Kata Bantu	Akan, belum, nak
KB-KEP	Kata Bantu-Kata Singkat	Kan
KBIL	Kata Bilangan	Semua, empat

KS	Kata Sendi Nama	Pada, dengan, untuk, dari
KP	Kata Penguat	Amat, sangat, sekali
KAD	Kata Adverba Jati	Tadi, esok, dahulu
KAD-KEP	Kata Adverba Jati-Kata Singkat	dulu
KAR	Kata Arah	Luar, sudut
KTY	Kata Tanya	Mana, bila, kenapa
KNF	Kata Nafi	Tidak, entah
KNF-KEP	Kata Nafi- Kata Singkat	tak
KEP	Kata Singkat	Ni, nak
KPB	Kata Pembenar	Ya, benar
KPE	Kata Perintah	Tolong, usah
KPM	Kata Pemeri	Ialah, adalah
KPN	Kata Pemeri	Saja, pun, lagi
KPN-KEP	Kata Pemeri-Kata Singkat	dri
KSR	Kata Seru	Wah, ah
FOR	Bahasa Asing	Lead, operation
FOR-KEP	Bahasa Asing- Bahasa Singkat	wf
FOR-NEG	Bahasa Asing-Bahasa Buruk	bitch
SL	Bahasa Slanga	La, ba, kerek, dok
LD	Bahasa Tempatan	Ko, sepang, uting, mengyu, sapa, jebe
MW	Kata Berkait dengan Malaysia	Ringgit, Malaysia
NEG	Bahasa Terlarang	Bana, sakai, bodoh

4.2 PRA-PEMPROSESAN DATA

Set data yang telah ditandakan dengan PGK akan ditunjuk dalam *dataframe*. Seterusnya, perkataan ditukar sebagai input X dan PGK sebagai output Y.

Model hanya menerima input yang dalam bilangan tetap, maka input haruslah diproseskan dengan *padding* kerana panjang setiap input tidak sama pada peringkat ini. list yang sudah menjadi indek akan diberi *padding*. Panjang ditetapkan dalam 74 token setiap kumpulan. Contohnya, kalau sesebuah input mempunyai 44 token, maka '0' *padding* diberikan sebanyak 30 kali. *Padding* diperlukan untuk mengelakkan input terpotong.

Hasil data daripada pra-pemprosesan yang dalam keadaan *array* dibahagikan kepada 2 set iaitu data untuk melatih array x_train dan data untuk menguji array x_text.

Langkah-langkah tersebut adalah seperti berikut.

1. Masuk data.
2. Token dan label diberi indeks.

3. Setiap token digabungkan dengan labelnya ke dalam sebuah urutan yang sama. Contohnya "Siti makan nasi" diubah menjadi [[("Siti", "KN"), ("makan", "KK"), ("nasi", "KN")], ……]

4. Token dan label kemudian dipisah dalam bentuk masukan X dan luaran Y.

5. Masukan X perlu memiliki panjang urutan yang sama. *Padding* dilakukan.

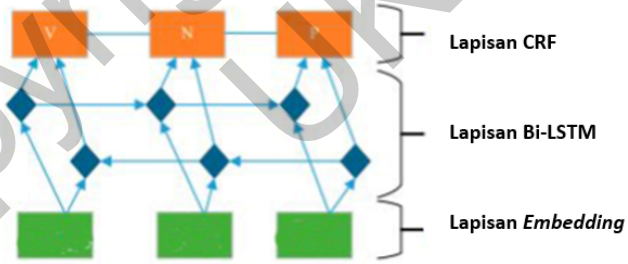
6. Luaran Y diubah ke dalam *one-hot-embedding*.

7. Dimensi X adalah jumlah data, panjang urutan dan dimensi y adalah jumlah data, panjang urutan dan jumlah kelas.

8. Langkah terakhir ialah membahagikan data ke dalam data latih x_train dan dan data uji array x_text.

4.3 PEMBINAAN MODEL

Reka bentuk model dalam kajian ini ialah berdasarkan Memori Jangka Pendek Panjang Dua Arah (Bi-LSTM) dengan Medan Rawak Bersyarat (CRF). Model ini pertama kali dicadangkan oleh Huang et al pada tahun 2015. Reka bentuk Model PGK BI-LSTM-CRF ditunjuk dalam Rajah 2.



Rajah 2 Gambaran keseluruhan Sistem Bi-LSTM-CRF
Sumber: Maihemuti, Aishan, Keheerjiang, Tuergan 2017

4.3.1 LAPISAN PENYISIPAN

Menurut kajian Mikolov et al pada tahun 2013, vektor ini menangkap maklumat semantik dan sintaksis perkataan. Contohnya, jika terdapat token "terima" dalam input, program akan mencari perkataan tersebut dalam model *Word2Vec*. Program akan mengecek apa indeks yang mewakil perkataan tersebut. Setiap perkataan memiliki

indeks wakili yang istimewa. Jika indeks 2 mewakil token “terima”, maka indeks 2 tidak akan digunakan untuk mewakil token lain.

Penyisipan perkataan adalah lapisan pertama dalam Keras dan lapisan ini memberikan matrik wakilan kepada setiap perkataan yang dapat menentukan ciri-ciri perkataan tersebut. Penyisipan perkataan yang diguna dalam kajian ini adalah dengan (1)Word2Vec pra-latihan dan (2) penyisipan perkataan secara rawak Keras. Perbandingan antara kedua-dua kaedah dilakukan untuk memilih kaedah yang paling sesuai digunakan untuk menguji data. Word2Vec pra-latihan ini didapati dari sumber awam *github* (Kyubyong, 2017), dilatih dengan korpus Wikipedia yang mempunyai 10010 kosa kata. Perkataan dalam Word2Vec pra-latihan adalah berdasarkan dari Wikipedia manakala data kajian mengandungi perkataan tidak formal. Oleh itu, model Word2Vec yang dilatih dengan korpus PGK Siti dan Sabrina (2018) ditambah untuk meninggikan ketepatan sistem. Model Word2Vec dilatih dengan pustaka *Gensim*. Gabungan kedua-dua model ini dinamakan sebagai Gabungan Word2Vec. Penyisipan perkataan dengan model Gabungan Word2Vec dan secara rawak mempunyai fungsi yang sama iaitu menukar semua perkataan yang unik kepada vektor $w \in R^n$.

4.3.2 LAPISAN RANGKAIAN NEURAL

Lapisan rangkaian neural yang digunakan dalam kajian ini ialah Bi-LSTM. Bi-LSTM sebagai alat pengekstrakan ciri perkataan. Bi-LSTM mengambil maklumat konteks dan mengecam potensi-potensi penandaan golongan bagi setiap input. Formula Deni, Moch, Ibnu (2019) menunjukkan keadaan tersembunyi (*hidden states*) dalam Bi-LSTM di tempat *i*. $\vec{h}i$ mewakili keadaan tersembunyi bagi input *forward* dan $\overleftarrow{h}i$ bagi input *backward*.

Formula Deni, Moch, Ibnu (2019) bagi Bi-LSTM adalah seperti berikut:

$$h_i = \vec{h}_i \oplus \overleftarrow{h_i}$$

4.3.3 LAPISAN MEDAN RAWAK BERSYARAT (CRF)

Lapisan ini digunakan untuk mengklasifikasikan penandaan golongan kata. Fungsi skor CRF digunakan dalam mencari urutan PGK yang mempunyai skor tertinggi dan hitung taburan kebarangkalian pada semua urutan PGK.

Lapisan CRF mempunyai faedah untuk memberi beberapa batasan kepada perkataan seperti sesebuah perkataan perlu diikuti dengan apa perkataan. Ciri-ciri tersebut tidak dapat diperagakan secara jelas dalam lapisan rangkaian neural terutamanya bagi skala data set yang kecil (Luisa, Dietrich & Benjamin, 2019) tetapi lapisan CRF mempunyai kekuatan ini. Dalam pelabelan urutan, hubungan antara label bersebelahan amat penting. (Deni, Moch & Ibnu, 2019). Contohnya dalam perkataan 'Negeri Sembilan', PGK bagi 'Negeri' ialah Kata Nama (KN) dan 'Sembilan' ialah Kata Bilangan(KB). "Negeri" yang mempunyai label KN harus membantu sistem untuk membuat keputusan bahawa "Sembilan" sesuai untuk diikutinya. CRF bertanggungjawab dalam tugasan ini dan memberi label KN semasa perkataan "Negeri" diikuti dengan "Sembilan".

4.4 PELATIHAN MODEL

Setelah model BI-LSTM-CRF ditubuhkan, x_train dimasukkan ke dalam model untuk melatih dan model akan mempelajari dari data tersebut. Jumlah *epoch* dan *batch size* diberikan dalam proses ini. *Epoch* dan *batch size* diubah setiap latihan untuk mendapat hasil yang terbaik. *Batch size* ialah jumlah sampel dari data yang akan dimasukkan ke dalam model manakala *epoch* ialah bilangan lelaran. Dalam setiap kumpulan yang mempunyai batch size yang berlainan, skor output dikeluarkan melalui lapisan Bi-LSTM bagi semua PGK di semua kedudukan. Output ini akan melalui lapisan CRF untuk mengira kecerunan output dan peralihan keadaan hujung (*transition edges*). Selama proses pelatihan, *crf loss* sebagai *loss function* pada model. Terdapat 4 parameter yang akan diubah iaitu *learning algorithm (optimizer*), *epoch* , jumlah Bi-LSTM units, *batch size* dan nilai *dropout rate*.

Model akan melakukan ramalan setelah menerima input. Output daripada ramalan ialah skor PGK yang sesuai bagi setiap token. Proses optimasi juga dilakukan.

Pencarian kombinasi nilai hiper-parameter yang menghasilkan prestasi paling tinggi dilakukan.

4.5 PENILAIAN MODEL

Pengujian diperlukan untuk menentukan ketepatan model semasa memberi PGK kepada data teks. Pengujian ini akan memberi gambaran kemampuan model Bi-LSTM-CRF dalam penandaan golongan kata bagi teks media sosial. Pengujian dilakukan ketika pelatihan model dan juga ketika model mempredikasi data teks. Menunjuk kepada kajian Derczynski (2017), ukuran-f dapat digunakan untuk menilai prestasi model. Prestasi model diuji dengan menggunakan metrik ketepatan ukuran-f ketika proses pelatihan dan pengujian untuk setiap label. Prestasi keseluruhan model diukur dengan rataan mikro. Rekod data uji dengan nilai hiper-parameter iaitu parameter yang boleh dipelajari (*Trainable Params*), masa mempelajari setiap *Epoch* dan ukuran-f disimpankan.

Jason (2019, 2020) menunjukkan bahawa prestasi model berdasarkan dengan ketepatan, *trainning loss*, *validation loss*. dan ukuran-f. Ketepatan adalah pecahan ramalan antara PGK benar yang dihasil oleh sistem dengan jumlah ramalan.

Formula Jason (2019) digunakan untuk menilai ketepatan sistem adalah seperti berikut:

$$Ketepatan = \frac{Jumlah\ PGK\ yang\ betul\ dihasil}{Jumlah\ ramalan}$$

Selain itu, *trainning loss* dan *validation loss* juga amat penting untuk mengenal pasti prestasi model Bi-LSTM-CRF sama ada *underfit*, *overfit* atau sesuai digunakan. Dalam tahun 2019, Jason menerangkan bahawa sebuah model yang *underfit, trainning loss* tidak akan menurun atau *trainning loss* akan terus menurun sampai tamat latihan. Jika model mengalami masalah *overfit* bermakna model tersebut belajar terlalu baik dengan data uji mengakibatnya model tersebut susah bernilai data baru. Model yang *overfit* mempunyai *validation loss* yang menurun dan meninggikan lagi. Model yang sesuai mempunyai *trainning loss* dan *validation loss* yang terus menurun sampai kestabilan dan jurang antara mereka kecil.

Jason (2020) juga menerangkan tentang ketepatan sistem mengikut 6 jenis parameter iaitu TP (tepat), FP (separa tepat), FN (tidak tepat), dapatan, kejituan dan

ukuran-f. Seperti yang dirujukkan (Jason, 2020), TP adalah jumlah sistem berjaya menghasil PGK dan tag tersebut benar, FP adalah jumlah sistem salah meramal tag benar sebagai tag salah, FN adalah jumlah sistem meramal tag salah sebagai tag benar. Dapatan adalah peratusan antara TP dengan jumlah TP dan FP. Kejituan adalah peratusan antara TP dengan jumlah TP dan FN. Ukuran-f adalah untuk mengukur ketepatan data yang dihasil oleh sistem. Prestasi model secara keseluruhan diukur dengan ukuran-f rataan mikro kerana pengedaran PGK pada data yang digunakan tidak seimbang.

Formula Jason (2020) untuk menilai skor dapatan adalah seperti berikut:

$$dapatan = \frac{TP}{TP + FP}$$

Formula Jason (2020) untuk menilai kejituan adalah seperti berikut:

$$kejituan = \frac{TP}{TP + FN}$$

Formula Jason (2020) untuk menilai F1 adalah seperti berikut:

$$ukuran\ f = \frac{dapatan * kejituan}{2 * (dapatan + kejituan)}$$

4.6 PEMILIHAN MODEL TERBAIK

Setiap model yang telah latih akan disimpan untuk kegunaan penilaian. Model terbaik akan dipilih dan dimuat untuk kegunaan ramalan dan pengujian. Pemilihan nilai uji merupakan bahagian yang terpenting pada proses hiper-parameter agar parameter yang terbaik dipilih sebagai model yang paling sesuai. Metrik penilaian yang digunakan adalah ketepatan dan ukuran-f untuk mengukur prestasi model.

5 HASIL KAJIAN

5.1 PENGUJIAN 1: JENIS PENYISIPAN PERKATAAN

Selepas model Bi-LSTM dengan CRF dibangunkan, pengujian model dijalankan. Pelatihan model menggunakan dua jenis penyisipan perkataan iaitu (1)penyisipan perkataan secara rawak Keras dan (2) Gabungan Word2Vec. Semua

model mengikut parameter awal sama iaitu 300 *embedding dimension, 30 batch size,* 0 *dropout point* dan 6 *epochs.* Rajah 3 dan rajah 4 menunjukkan *training loss* dan *validation loss* kedua-dua model dalam latihan. Rajah 5 dan rajah 6 menunjukkan ketepatan daripada hasil ujian.

Rajah 3 Bentuk learning curve model Gabungan Word2Vec

Rajah 4 Bentuk learning curve model penyisipan perkataan secara rawak Keras

micro avg	0.94	0.94	0.94
macro avg	0.85	0.76	0.79
weighted avg	0.94	0.94	0.94

Rajah 5 Hasil pengujian model Gabungan Word2Vec

micro avg	0.92	0.92	0.92
macro avg	0.83	0.81	0.82
weighted avg	0.93	0.92	0.92

Rajah 6 Hasil pengujian model penyisipan perkataan secara rawak Keras

Walaupun kedua-dua model mempunyai hasil pengujian yang baik, model yang menggunakan penyisipan perkataan dengan Gabungan Word2Vec mempunyai jurang yang kecil antara *validation loss* dengan *training loss*. Model yang menggunakan penyisipan perkataan secara rawak mengalami masalah *overfit.* Oleh itu, model penyisipan perkataan yang sesuai ialah model Gabungan Word2vec. Masalah *overfit* bagi penggunaan penyisipan perkataan secara rawak Keras adalah kerana jumlah *trainable parameter* model penyisipan secara rawak sangat besar iaitu 3091116 manakala *trainable parameter* model Gabungan Word2Vec hanya 1474416 sahaja. Kerumitan ciri dalam lapisan menyebabkan terjadinya *overfit* model. Selain itu, *overfit* berlaku kerana set data latih yang digunakan untuk melatih model sangat kecil. Sebahagian besar perkataan dalam data uji tidak wujud dalam set data latih. Varian

yang besar menyebabkan *overfit* berlaku. Gabungan Word2Vec dapat mengatasi masalah varian ini kerana ia telah dilatih dengan set data yang lagi besar.

Oleh itu, Gabungan Word2Vec akan dipilih sebagai kaedah penyisipan perkataan dalam model Bi-LSTM-CRF.

5.2 PENGUJIAN 2: PERUBAHAN PARAMETER

Parameter yang menghasilkan prestasi terbaik adalah *learning algorithm Adam*, *dropout point* 0, *batch size* 6 dan 30 *epoch*. Perbandingan antara *learning algorithm* Adam dan *RMSprop menunjukkan* Adam mendapat ukuran-f lebih tinggi iaitu 94%. Batch size 6 dan epoch 30 digunakan kerana *learning curve* mula meningkat selepas nilai 6. Model ini tidak memerlukan *dropout point* kerana ia akan menyebabkan *training loss* kurang daripada *validation loss*. Hal ini kerana *dropout point* adalah untuk mengelakkan masalah *overfit* dan model ini tidak mengalami masalah ini atas bantuan lapisan penyisipan perkataan Gabungan Word2Vec. Jadual 3 menunjukkan nilai uji dipilih yang menghasilkan prestasi model terbaik.

Jadual 3 Perbandingan antara parameter yang berlainan

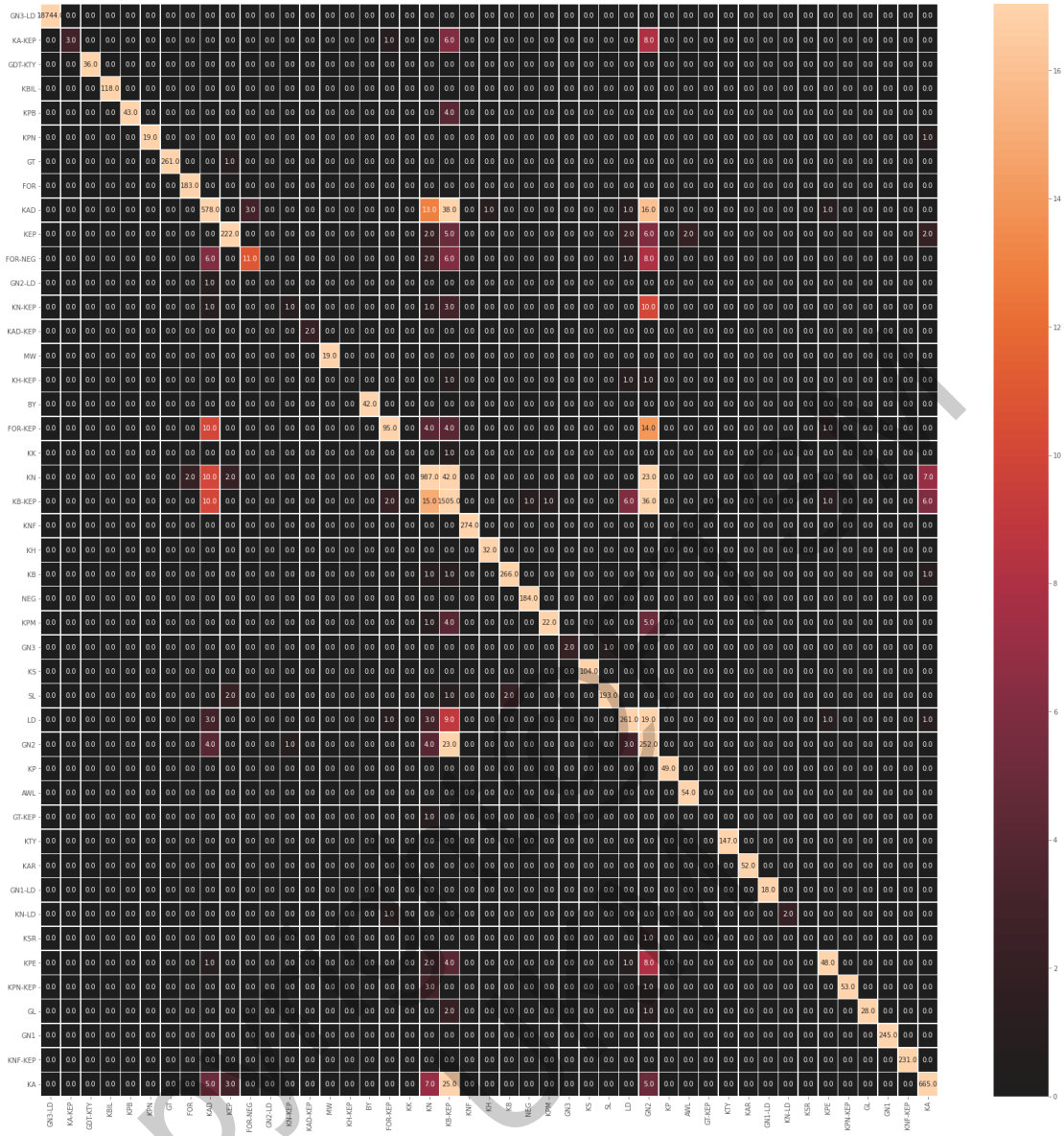
Parameter	**Nilai Uji**	**Nilai yang menghasilkan prestasi model terbaik**
Learning algorithm	Adam, RMSprop	Adam
Dropout	0, 0.5	0
Batch size	6, 7, 10	6
Epoch	20, 30, 32	30

5.3 HASIL PENGUJIAN

Daripada kajian atas, model yang dipilih ialah model yang mempunyai lapisan penyisipan perkataan dengan Word2Vec pra-latihan, *embedding_dim* 300, *batch_size* 30, *epioch* 6 dengan *optimizer* Adam. Hasil ujian daripada data uji mendapat ketepatan sebanyak 93.81% dan ukuran-f mikro mendapat 94%. Rajah 7 menunjuk ukuran-f bagi setiap PGK dan rajah 8 menunjuk matriks keliru sistem PGK Bi-LSTM-CRF.

	precision	recall	f1-score	support
GN3-LD	0.80	0.25	0.38	16
KA-KEP	1.00	1.00	1.00	33
GDT-KTY	1.00	1.00	1.00	133
KBIL	1.00	0.95	0.98	65
KPB	1.00	0.94	0.97	18
KPN	1.00	0.99	1.00	281
GT	1.00	1.00	1.00	193
FOR	0.92	0.91	0.92	656
KAD	0.96	0.94	0.95	233
KEP	0.81	0.49	0.61	35
FOR-NEG	0.00	0.00	0.00	3
GN2-LD	0.67	0.17	0.27	12
KN-KEP	1.00	1.00	1.00	14
KAD-KEP	1.00	1.00	1.00	13
MW	1.00	0.73	0.84	11
KH-KEP	1.00	1.00	1.00	56
BY	0.96	0.80	0.87	128
FOR-KEP	0.00	0.00	0.00	3
KK	0.94	0.92	0.93	1060
KN	0.90	0.96	0.93	1525
KB-KEP	1.00	1.00	1.00	299
KNF	1.00	1.00	1.00	37
KH	0.97	0.99	0.98	274
KB	0.99	1.00	1.00	191
NEG	0.94	0.59	0.73	27
KPM	1.00	0.50	0.67	2
GN3	1.00	1.00	1.00	128
KS	0.99	0.97	0.98	197
SL	0.95	0.89	0.92	339
LD	0.57	0.83	0.67	242
GN2	1.00	1.00	1.00	69
KP	0.94	0.98	0.96	51
AWL	0.00	0.00	0.00	1
GT-KEP	1.00	1.00	1.00	132
KTY	1.00	1.00	1.00	58
KAR	1.00	0.90	0.95	10
GN1-LD	0.00	0.00	0.00	0
KN-LD	1.00	0.29	0.44	7
KSR	0.98	0.71	0.82	69
KPE	0.98	0.92	0.95	50
KPN-KEP	0.00	0.00	0.00	0
GL	1.00	0.89	0.94	28
GN1	1.00	1.00	1.00	294
KNF-KEP	1.00	1.00	1.00	262
KA	0.97	0.91	0.94	759
micro avg	0.94	0.94	0.94	8014
macro avg	0.85	0.76	0.79	8014
weighted avg	0.94	0.94	0.94	8014

Rajah 7 Ukuran-f bagi semua tag



Rajah 8 Matriks keliru sistem PGK Bi-LSTM-CRF

Kebanyakan PGK mendapat ukuran-f yang tinggi iaitu melebihi 0.6. Namun begitu, terdapat 5 PGK iaitu FOR-NEG, FOR-KEP, AWL, GN1-LD dan KPN-KEP mendapat ukuran-f 0.00 kerana jumlah bilangan data tag ini sangat rendah dalam set data. Hanya 5 perkataan FOR-NEG, 8 perkataan FOR-KEP, 4 perkataan AWL dan 7 perkataan GN1-LD dalam kajian ini. PGK bagi perkataan tempatan seperti GN3-LD, GN2-LD dan KN-LD mendapat ukuran-f yang kurang daripada 0.5. Berdasarkan laporan ukuran-f dalam rajah 7, skor dapatan GN3-LD, GN2-LD dan KN-LD agak bagus dan skor kejituan agak teruk menjejaskan prestasi ukuran-f. Hal ini mungkin

kerana ciri perkataan ketiga-tiga tag sangat serupa kerana ketiga-tiga tag mewakili perkataan tempatan.

Daripada 10 PGK Bahasa Melayu media sosial iaitu BY (Bunyi), KN-LD (Kata Nama, Bahasa Singkatan), KN-KEP (Kata Nama-Kata Singkat), KA-KEP (Kata Adjektif, Kata Singkat), FOR (Bahasa Asing), FOR-KEP (bahasa Asing, Bahasa Singkat), FOR-NEG (Bahasa Asing, Bahasa Buruk), SL (Bahasa Slanga), LD (Bahasa Tempatan), dan NEG(Bahasa Terlarang), hanya 3 PGK yang mendapat ketepatan yang kurang daripada 0.5.

6 KESIMPULAN

Kesimpulannya, pembangunan sistem PGK Bi-LSTM-CRF telah berjaya dibangunkan dan mencapai objektif yang telah tetapkan. Gabungan Word2Vec sebagai *embedding_matrix* lebih baik berbanding dengan penyisipan perkataan Keras kerana masalah *overfit* tidak berlaku. Ketepatan sistem PGK model Bi-LSTM-CRF sebanyak 93.81% dan ukuran-f 94% juga lebih tinggi daripada kajian lepas QTAG Bahasa Melayu (Siti dan Sabrina, 2018). Oleh itu dapat disimpulkan penggunaan kaedah pembelajaran mendalam dengan fitur penyisipan yang sesuai boleh mempertingkatkan lagi kecekapan penandaan golongan kata pada teks bahasa Melayu media sosial. Walau bagaimanapun, penambahbaikan seperti melatih model dengan data *tweet* yang lebih besar dengan data semasa dan BERT Bahasa Melayu digunakan sebagai matrik penyisipan perkataan boleh dijalankan bagi meningkatkan lagi ketepatan kajian. Kajian ini juga memberi informasi berguna bahawa *overfit* yang berlaku pada set data kecil dapat diatasi dengan menggunakan model pra-latihan untuk penyisipan perkataan. Selain itu, sistem PGK yang dibangunkan juga dapat digunakan dalam aplikasi mesin yang memerlukan pemahaman Bahasa Tabii dalam teks media sosial Bahasa Melayu.

7 RUJUKAN

Deni Cahya Wintaka, Moch Arif Bijaksana, Ibnu Asror. 2019. Named-entity Recognition on Indonesian tweets using Biderectional LSTM-CRF. Procedia Computer Science 157: 221-228. https://doi.org/10.1016/j.procs.2019.08.161.

Shamsan Gaber, Mohd Zakree Ahmad Nazri, Nazlia Omar& Salwani Abdullah. 2020. Part-of-Speech (POS) Tagger for Malay Language using Naïve Bayes and K-Nearest Neighbor Model. https://www.researchgate.net/publication/342211317_Part-of-Speech_POS_Tagger_for_Malay_Language_using_Naive_Bayes_and_K-Nearest_Neighbor_Model.

Siti Noor Allia Noor Ariffin, Sabrina. 2018. Part-of-Speech Tagger for Malay Social Media Texts. Gema Online Journal of Language Studies 18(4).https://www.researchgate.net/project/Part-of-Speech-Tagger-for-Malay-Social-Media-Texts.

Jason Brownlee. 2017. How to Develop Word Embeddings in Python with Gensim. https://machinelearningmastery.com/develop-word-embeddings-python-gensim/.

Jason Brownlee. 2019. Classification Accuracy is Not Enough: More Performance Measures You can Use. https://machinelearningmastery.com/classification-accuracy-is-not-enough-more-performance-measures-you-can-use/.

Jason Brownlee. 2019. How to use Learning Curves to Diagnose Machine Learning Model Performance. https://machinelearningmastery.com/learning-curves-for-diagnosing-machine-learning-model-performance/.

Jason Brownlee. 2021. How to Use Embedding Layers for Deep Learning with Keras. https://blog.keras.io/using-pre-trained-word-embeddings-in-a-keras-model.html.

Kalaiarasi Sonai Muthu Anbananthen, Jaya Kumar Krishnan, Mohd. Shohel Sayeed and Praviny Muniapan, 2017. Comparison of Stochastic and Rule-Based POS Tagging on Malay Online Text. American Journal of Applied Science,14(9), 843-851. https://doi.org/10.3844/ajassp.2017.843.851.

Luisa Marz, Dietrich Trautmann, Benjamin Roth. 2019. Domain adaptation for part-of-speech tagging of noisy user-generated text. https://www.aclweb.org/anthology/N19-1345.

Maihemuti Maimaiti , Aishan Wumaier , Kahaerjiang Abiderexiti & Tuergen Yibulayin. 2017. Bidirectional Long Short-Term Memory Network with a Conditional Random Field Layer for Uyghur Part-Of-Speech Tagging. semanticscholar.org/paper/Bidirectional-Long-Short-Term-Memory-Network-with-a-Maimaiti-Wumaier/f27e1c036cdf5bbec542ac2a9ef6ce32c6cdfcde.

Meghdad Farahmand. 2019. Pre-trained Word Embeddings or Embedding Layer? – A Dilemma. https://towardsdatascience.com/pre-trained-word-embeddings-or-embedding-layer-a-dilemma-8406959fd76c.

Niklas Donges, 2019. A Guide to Run RNN: Understanding Recurrent Neural Network and LSTM. Builtin. https://builtin.com/data-science/recurrent-neural-networks-and-lstm.

Y.Xia dan Q.Wang. 2018. Incorporating Dictionaries into Deep Neural Networks for the Chinese Clinical Named Entity Recognition. https://www.researchgate.net/publication/324536347_Incorporating_Dictionaries_into_Deep_Neural_Networks_for_the_Chinese_Clinical_Named_Entity_Recognition/citations.

Yuda Munarko, Yufis Azhar, Maulina Balqis & Ekawati. 2017. POS Tagger Tweet Bahasa Indonesia. https://www.researchgate.net/publication/315437571_POS_Tagger_Tweet_Bahasa_Indonesia.

Zeineb Ghrib. 2020. Use Pre-trained Word Embedding to detect real disaster tweet. https://towardsdatascience.com/pre-trained-word-embedding-for-text-classification-end2end-approach-5fbf5cd8aead.

Zhiheng Huang, Wei Xu, Kai Yu. 2015. Bidirectional LSTM-CRF Models for Sequence Tagging. https://www.groundai.com/project/bidirectional-lstm-crf-models-for-sequence-tagging/1.