

## **BAB II**

### **Tinjauan Pustaka**

Pada tahun 1950-an, ilmuwan Alan Turing mencetuskan tes Turing, yaitu tes kemampuan komputer untuk mengidentifikasi dengan benar apakah seseorang sedang berbicara dengan manusia atau komputer. Joseph Weizenbaum kemudian menciptakan model percakapan pertama dengan nama ELIZA dan punya kapabilitas untuk melakukan tes Turing pada masa itu. Akan tetapi, sampai saat ini belum ada satupun komputer yang lolos tes Turing semenjak tes itu diperkenalkan, sehingga menjadi pemicu bagi peneliti untuk mengembangkan model percakapan yang lebih baik.

Perkembangan model percakapan dalam beberapa tahun terakhir ditandai dengan banyaknya penelitian jaringan saraf. Model percakapan dengan jaringan saraf yang populer saat ini adalah *sequence-to-sequence* (SEQ2SEQ). Model ini dibentuk dari dua jaringan RNN (2-RNN) dan juga dikenal sebagai model *encoder-decoder*. Kegunaannya dipopulerkan oleh Sutskever (2014) untuk tugas terjemahan mesin dalam menerjemahkan Bahasa Inggris ke Bahasa Prancis. Sutskever (2014) memakai *multilayer LSTM* pada model SEQ2SEQ dan membalik urutan kata pada kalimat sumber sehingga terjadi banyak dependensi jangka pendek untuk mengatasi masalah pengoptimalan yang jauh lebih sederhana. Setelah itu, terdapat penelitian SEQ2SEQ dengan *Gated Recurrent Unit* (GRU). GRU diketahui mampu mempelajari representasi frasa bahasa secara semantik dan sintaksis yang diperkenalkan pada tugas terjemahan mesin (Cho *et al.*, 2014).

Mekanisme gerbang LSTM dan GRU telah berhasil menangani masalah gradien yang menghilang atau meledak selama pelatihan jangka panjang. Akan tetapi, butuh beberapa bukti empiris yang menunjukkan unit gerbang mana yang paling baik untuk masalah optimasi model percakapan. Chung (2014) menemukan bahwa GRU menghasilkan model yang setara dengan LSTM, tetapi GRU secara komputasi lebih efisien. Hal itu disebabkan karena GRU disusun atas dua gerbang: *reset* dan *update*; sedangkan LSTM disusun atas tiga gerbang yaitu *input*, *output*, dan *forget* serta terdapat unit memori yang dikontrol oleh gerbang tersebut (Chung

*et al.*, 2014). GRU mempunyai sedikit parameter sehingga pelatihan dan konvergensinya lebih cepat daripada LSTM. Akan tetapi, LSTM pada prinsipnya dapat menggunakan unit memorinya untuk mengingat informasi yang jaraknya jauh dan melacak berbagai atribut yang sedang diproses (Karpathy *et al.*, 2016). LSTM mempertahankan pemahaman informasi di setiap langkah waktu secara lebih lama, sehingga untuk pelatihan jangka panjang dengan data yang besar, secara teori LSTM lebih baik daripada GRU.

Arsitektur SEQ2SEQ itu sendiri juga memiliki keterbatasan. Model SEQ2SEQ memerlukan seluruh input yang dikodekan ke dalam vektor tunggal dengan panjang yang berukuran tetap, sehingga model tidak dapat menggeneralisasi input jauh lebih lama daripada kapasitas tetapnya. Salah satu cara untuk mengatasi masalah ini adalah dengan memakai mekanisme perhatian (Bahdanau *et al.*, 2014). Mekanisme perhatian mampu memperluas vektor ukuran tetap dengan memungkinkan model secara otomatis mencari bagian dari sumber kalimat yang relevan untuk memprediksi kata target, tanpa harus membentuk bagian-bagian itu sebagai segmen padat secara eksplisit.

Metode optimasi yang juga sering digunakan pada *deep learning* adalah *gradient descent* (GD). Ide dasar GD adalah menyesuaikan bobot model dengan mencari turunan dari fungsi biaya (*error*) dan secara iteratif bergerak menuju nilai parameter yang meminimalisasi nilai dari fungsi biaya (*error*) tersebut. Contoh ekstensi GD adalah AdaGrad, RMSProp, dan Adam. AdaGrad bekerja dengan baik pada masalah gradien yang tersebar, sedangkan RMSProp cocok untuk masalah pembaruan bobot secara online dan non-stasioner. Adam lalu menggabungkan keuntungan dari dua ekstensi itu untuk mempertahankan tingkat pembelajaran per-parameter dari perkiraan momen pertama dan kedua (Kingma *et al.*, 2014).

Teknik-teknik optimasi yang sudah dijelaskan tadi telah tersedia secara resmi pada *framework deep learning* dan dirilis secara terbuka (*opensource*). Beberapa penelitian telah menunjukkan detail-detail pekerjaan mereka seperti bahasa pemrograman maupun *framework* yang digunakan pada penelitian mereka. Perbandingan mengenai teknik optimasi model SEQ2SEQ tersebut dapat dilihat pada Tabel 2.1. berikut ini.

Tabel 2.1. Perbandingan Teknik Optimasi Model SEQ2SEQ

Item Pemandangan	Sutskever (2014)	Cho (2014)	Bahdanau (2014)	Vinyals (2015)	Made (2019)*
Mekanisme Gerbang	LSTM	GRU	GRU dengan <i>Attention</i>	LSTM	Kombinasi LSTM dan GRU dengan <i>Attention</i>
<i>Gradient Descent</i>	SGD	AdaDelta, SGD	SGD, AdaDelta	SGD, AdaGrad	Adam
Bahasa Pemrograman	C++	Python	C++	C++	Python
<i>Deep Learning Framework</i>	-	Theano	-	-	Keras dan Tensorflow
Hasil	Terjemahan Mesin Bahasa Inggris - Prancis	Terjemahan Mesin Bahasa Inggris - Prancis	Terjemahan Mesin Bahasa Inggris - Prancis	Model Percakapan Bahasa Inggris	Model Percakapan Bahasa Indonesia

\* Dalam penelitian ini.

Penelitian ini memakai *OpenSubtitle 2018* Bahasa Indonesia selayaknya yang dikerjakan oleh Vinyals (2015) dengan *OpenSubtitle 2016* Bahasa Inggris. Penelitian yang ada sebagian besar memakai korpus Bahasa Inggris seperti yang ditunjukkan oleh Tabel 2.1. Itu artinya masih ada ruang untuk melakukan penelitian dengan *dataset* Bahasa Indonesia. Tabel 2.1 juga menunjukkan optimasi model SEQ2SEQ bisa dilakukan dengan bantuan kerangka kerja *deep learning*. Penulis memakai *framework* Keras dan Tensorflow untuk implementasi kombinasi gerbang LSTM dan GRU dengan penambahan mekanisme perhatian dan *gradient descent*. Diharapkan hasil performa setelah optimasi akan menghasilkan model yang sepadan atau lebih dari implementasi-implementasi sebelumnya.