

BAB II

LANDASAN TEORI

2.1 Penelitian Terkait

Tabel 2.1 merupakan beberapa penelitian-penelitian terdahulu yang dipilih dan dijadikan sebagai referensi oleh peneliti di dalam penulisan maupun penelitian ini, yakni :

Tabel 2.1 Penelitian Sebelumnya

Judul Penelitian	Metode	Rangkuman
<i>Aspect-based sentiment analysis to review products using Naïve Bayes</i> (Mubarok, Adiwijaya, & Aldhi, 2017)	Naïve Bayes	Mereka melakukan ekstraksi informasi terhadap <i>review</i> produk online. Data-data tersebut kemudian melalui <i>preprocessing</i> , pemilihan fitur, dan klasifikasi aspek beserta sentimennya. Proses pemilihan fitur menggunakan <i>Chi Square</i> untuk memilih subset dengan kata yang relevan untuk digunakan pada model Naïve Bayes model. Langkah terakhir yaitu menggunakan Naïve Bayes sebagai metode klasifikasi pada sentimen dan aspek. Pengklasifikasian <i>Aspect-based Sentiment Analysis</i> dengan Naïve Bayes menghasilkan F1-Measure sebesar 78.12%. F1-Measure terbaik untuk klasifikasi aspek sebesar 88.13%, sedangkan klasifikasi sentimen sebesar 75%.
<i>ELECTRONIC PRODUCT FEATURE-BASED SENTIMENT ANALYSIS USING NU-SVM</i>	NU-SVM	Penelitian ini bertujuan untuk mendapatkan sentimen pada ulasan ponsel berdasarkan aspeknya. Data ulasan ponsel berbahasa Inggris diambil dari www.cnet.com . Pada tahap ekstraksi fitur digunakan template dependency relation dengan frasa yang sesuai, yang kemudian diikuti dengan penyaringan fitur. Pada tahap evaluasi didapatkan F1 Measure sebesar 86.25% untuk kelas positif dan 77.71% pada kelas negatif. Sedangkan untuk akurasi identifikasi aspek sebesar 82%.

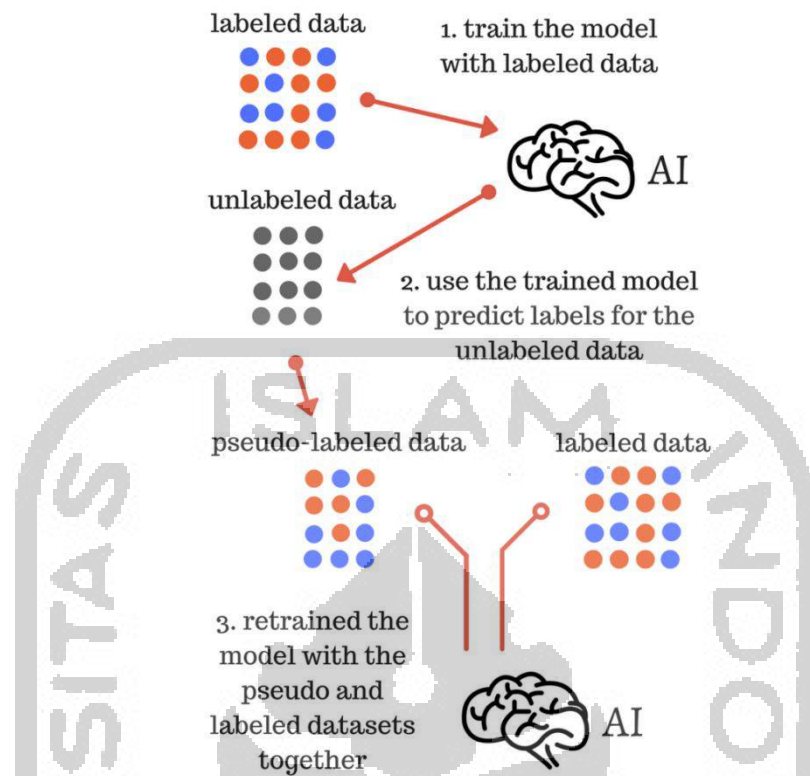
METHOD (Gozali, 2017)		
<i>Aspect Based Sentiment Analysis with Gated Convolutional Networks</i> (Xue & Li, 2018)	Gated Convolutional Networks	Penelitian ini merangkum pendekatan ini ke dalam dua subtugas: sentimen kategori-aspek analisis (ACSA) dan analisis sentimen istilah-aspek (ATSA). Dua subtugas tersebut menggunakan model <i>Gated Convolutional network</i> with Aspect Embedding (GCAE). <i>Gated Tanh-ReLU Units</i> dapat memilih output fitur sentimen bergantung pada aspek ataupun entitas yang diberikan. Dengan digunakannya GTRU dapat secara efektif mengontrol aliran sentimen bergantung pada aspek yang diberikan. Model ini dibuat berdasarkan metode CNN dan mekanisme penggerbangan (<i>gating</i>) yang lebih akurat dan efisien. Model yang diaplikasikan pada dataset SemEval ini terbukti mampu menaikkan performa dibandingkan dengan metode neural network lainnya.
<i>Medical Text Classification using Convolutional Neural Networks</i> (Hughes, Li, Kotoulas, & Suzumura, 2017)	CNN	Mereka memanfaatkan CNN sebagai pendekatan untuk melakukan klasifikasi teks pada level kalimat, berdasarkan semantik yang sering muncul dari ekstraksi korpus teks medis. Dilakukan perbandingan menggunakan 3 metode lainnya, yaitu : <i>Sentence Embeddings</i> , <i>Mean Word Embeddings</i> , dan <i>Word Embeddings</i> dengan BOW (<i>bag-of-words</i>). Dari evaluasi yang dilakukan secara mendetil pada setiap metode, metode CNN yang digunakan pada penelitian ini lebih baik dalam melkaukan klasifikasi dari pendekatan NLP dengan metode lain sebanyak 15%.
<i>A Pseudo Label based Dataless Naive Bayes Algorithm for Text Classification</i>	Pseudo-Label Based Dataless Naive Bayes PL-DNB	Li membangun sebuah <i>Pseudo-Label Based Dataless Naive Bayes</i> PL-DNB sebagai algoritma klasifikasi, yang bisa digunakan untuk melatih data yang belum terlabeli menggunakan <i>seed words</i> , daripada menggunakan data berlabel. <i>Pseudo-label</i> terbarukan secara berulang, and informasi dari tetangga terdekat dokumen digunakan

<p><i>with</i> <i>Words</i> 2018)</p>	<p><i>Seed</i> (Li, 2018)</p>	<p>untuk menghindari <i>noisy</i> pseudo-label. Pada penelitian ini PL-DNB diimplementasikan pada 2 dataset yang sering digunakan, dimana satu data memiliki data yang seimbang, sedangkan satu lagi tidak. Selain itu, PL-DNB juga dibandingkan dengan 4 algoritma lain yaitu <i>Seed Word Based Naive Bayes</i> (SNB), <i>Seed-guided Topic Model</i> (STM), Naive Bayes (NB), Support Vector Machines (SVMs) untuk mendapatkan algoritma yang efektif. Li menunjukkan secara empiris bahwa PL-DNB mengungguli algoritma klasifikasi dengan teks berlabel, PL-DNB juga memiliki performa baik pada klasifikasi dataset yang tidak seimbang.</p>
---	---------------------------------------	---

Berbeda dengan penelitian sebelumnya, penelitian yang saya lakukan yaitu *Aspect Based Sentiment Analysis* (ABSA) pada bidang pariwisata dengan menggunakan metode 1D CNN. ABSA pada penelitian ini memanfaatkan pelabelan dengan menggunakan metode *pseudo labeling* CNN untuk mendapatkan label. Ekstraksi fitur yang digunakan yaitu POS Tagging dan *negation handling* yang kemudian diolah menggunakan 1D CNN untuk mendapatkan aspek kategori dan sentimen suatu ulasan.

2.2 Pseudo Labeling

Tahap penggunaan *pseudo-labeling* cukup mudah untuk dilakukan seperti yang ditunjukkan Gambar 2.1, yaitu dengan melakukan pelatihan model untuk memprediksi label pada data yang belum terlabeli, sehingga tercipta sebuah *pseudo-label*. Kemudian, mengkombinasikan data yang sudah terlabeli dengan data *pseudo-label* dalam sebuah dataset baru untuk digunakan sebagai pelatihan data. Sebuah jaringan dilatih secara terawasi (*supervised*) menggunakan dataset baru tersebut secara bersamaan. Data yang belum terlabeli (*pseudo-label*) memilih kelas yang memiliki probabilitas prediksi tertinggi di setiap pembaruan bobot, kemudian hasil kelas tertinggi akan digunakan layaknya label aslinya. Pada prinsipnya, metode ini dapat dikombinasikan hampir di setiap model jaringan syaraf dan metode pelatihan (Lee, 2013).



Sumber : datawhatnow.com/pseudo-labeling-semi-supervised-learning/

Gambar 2.1 Langkah-langkah pada pseudo-label

2.3 Analisis Sentimen Berbasis Fitur

Sentimen analisis adalah salah satu penelitian yang populer dalam Pengolahan Bahasa Alami (PBA). Salah satu tujuan dengan adanya sentimen analisis yaitu mendapatkan nilai polaritas sentimen setiap kalimat atau ulasan. Analisis sentimen berbasis fitur merupakan perluasan dari sentimen analisis yang berfokus dalam ekstraksi kalimat atau ulasan bersamaan dengan aspek dan nilai polaritas sentimennya (Al-ayyoub et al., 2017).

2.4 Oversampling

Pada pendekatan eksternal (tingkatan data) untuk mengatasi data tidak seimbang, pertama, dataset harus diseimbangkan dan algoritma klasifikasi konvensional kemudian diterapkan sehingga kinerja pengklasifikasi tidak bias terhadap kelas mayoritas. Menyeimbangkan dataset dapat dilakukan dengan *undersampling* yaitu dengan memangkas kelas mayoritas atau *oversampling* yaitu dengan menambah kelas minoritas untuk dataset (Chawla, Japkowicz, & Drive, 2004; Monard, n.d.). Gambar 2.2 di bawah ini merupakan gambaran teknik *oversampling* data.



Sumber : towardsdatascience.com

Gambar 2.2 Teknik *oversampling* data

2.5 Ekstraksi Fitur

Ekstraksi fitur menciptakan variabel baru dari kombinasi-kombinasi yang diciptakan untuk mengurangi dimensionalitas (Hira & Gillies, 2015). Tujuan dari dilakukan ekstraksi fitur yaitu untuk mencari set minimal dari fitur baru melalui beberapa transformasi (Motoda & Liu, 2002). Ekstraksi fitur memiliki banyak kegunaan dalam mengurangi dimensionalitas untuk proses selanjutnya (Motoda & Liu, 2002), fitur yang didapatkan dapat membantu meningkatkan perolehan hasil untuk *data mining* (H. Liu & Motoda, 1998). Pada kasus ini, ada 2 ekstraksi fitur yang dilakukan yaitu POS Tag dan *negation handling*.

2.6 Negation Handling

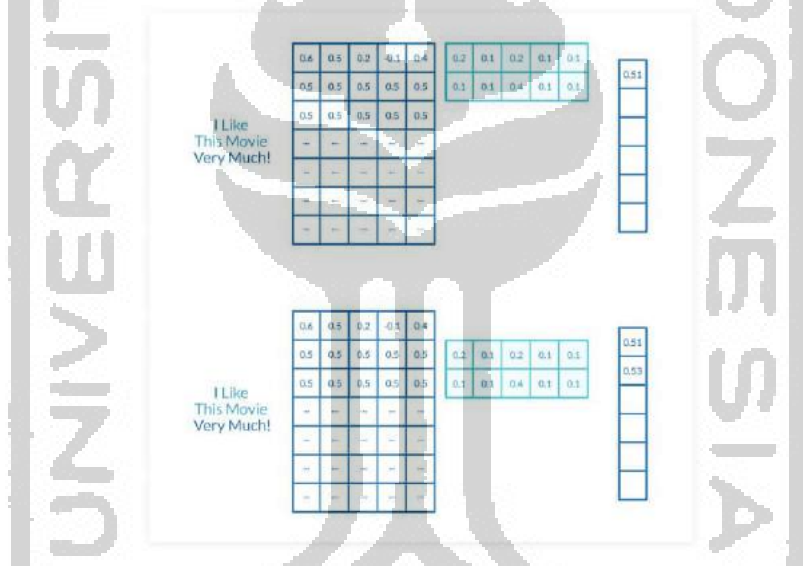
Negasi merupakan linguistik yang sangat penting karena mempengaruhi polaritas dari suatu kata. Tujuan dengan adanya *negation handling* yaitu untuk meningkatkan performa dari sentimen analisis dengan melihat adanya kata negasi yang digunakan (Farooq, Mansoor, Nongailard, Ouzrout, & Abdul, 2016). Beberapa kata negasi yang digunakan pada kasus ini antara lain, “tidak”, “bukan”, “jangan”, dan “kurang”.

2.7 POS Tagging

Part of speech (POS) adalah proses pemberian label untuk setiap kata dalam sebuah kalimat. POS Tagging ini dapat mengkategorikan kelas kata menjadi kata benda, kata sifat, kata kerja, dll (Kumawat, 2015).

2.8 Convolutional Neural Network (CNN)

Convolutional Neural Networks memiliki kemampuan baik dalam menyelesaikan masalah visi komputer karena dapat beroperasi secara konvolusional, yaitu melakukan ekstraksi fitur dari *patch* masukan lokal yang memungkinkan modularitas representasi dan efisiensi data. CNN sangat signifikan untuk pemrosesan berurutan. Lapisan konvolusi 1D juga tidak mengalami perubahan translasi karena masukan transformasi dilakukan pada setiap *patch*, sebuah pola yang dipelajari pada posisi tertentu dalam kalimat kemudian dapat dikenali pada posisi yang berbeda. Mirip dengan CNN 2D, *patch* 1D dapat diekstraksi dari masukan dan memberikan keluaran nilai maksimum atau nilai rata-rata, proses ini disebut dengan *Max Pooling* dan *Average Pooling*. Penggunaannya pun sama dengan CNN 2D, yaitu untuk mengurangi panjang dari masukan 1D (secara teknik disebut dengan *subsampling*) (Chollet, 2017). Gambar 2.3 merupakan contoh operasi konvolusi 1D.



Sumber : missinglink.ai/guides/convolutional-neural-networks

Gambar 2.3 Operasi Konvolusi 1D

CNN adalah sebuah konstruk matematika yang biasanya disusun oleh 3 tipe layer, yaitu *convolution*, *pooling*, dan lapisan yang sepenuhnya terhubung. Dua layer pertama, *convolution* dan *pooling* melakukan ekstraksi fitur, sedangkan layer ketiga, menempatkan fitur yang terekstraksi menjadi hasil akhir keluaran, seperti klasifikasi (Yamashita, Nishio, Kin, Do, & Togashi, 2018).

Konvolusi adalah sebuah tipe operasi linier khusus yang digunakan untuk ekstraksi fitur, dimana sebuah array angka, yang disebut kernel diimplementasikan pada masukan, yang merupakan array angka yang disebut tensor. Kalkulasi yang dilakukan pada masing-masing

elemen matriks kernel dan masukan tensor dan dijumlahkan untuk mendapatkan keluaran yang disebut dengan *feature map*. Tahap ini dilakukan berulang hingga didapatkan *feature map* yang merepresentasikan karakteristik dari masukan tensor. Oleh karena itu, kernel dengan ukuran lain bisa memiliki hasil ekstraksi yang berbeda pula. Perhitungan propagasi konvolusi ditunjukkan pada persamaan (2.1).

$$a_{ij}^{(k)} = \sum_{s=0}^{m-1} \sum_{t=0}^{n-1} W_{st}^{(k)} x_{(i+s)(j+t)} + b^{(k)} \quad (2.1)$$

Keterangan :

x : masukan

a^k : setelah konvolusi

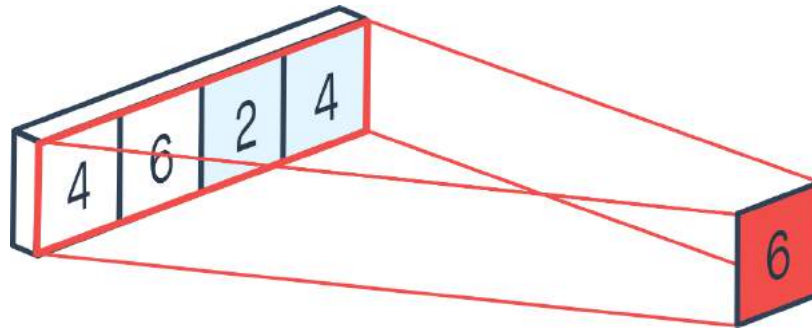
k : indeks kernel (*weight filter*)

W : kernel (*weight filter*)

b : bias

Pooling disebut juga dengan penurunan sample (*downsampling*) yang mengurangi dimensionalitas bidang namun tetap menjaga informasi yang penting. Operasi *pooling* yang sering digunakan yaitu *max pooling*, dimana operasi ini mampu melakukan ekstraksi dari filter masukan *feature maps*, kemudian memilih nilai tertinggi di setiap filter dan membuang nilai lainnya.

Pooling memiliki beberapa jenis, pada kasus ini digunakan 1D *global max pooling*. 1D *Global max pooling* mengambil sebuah vektor dan menghitung nilai maksimum dari semua nilai untuk masing-masing saluran input. Gambar 2.4 menampilkan pengambilan vektor dengan menggunakan 1D *global max pooling*.



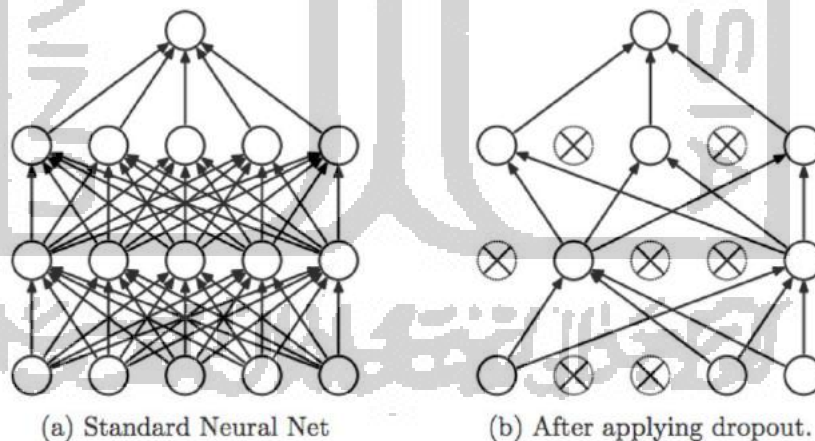
Sumber : peltarion.com/

Gambar 2.4 Contoh pengambilan vektor 1D global max pooling

Layer FC yaitu lapisan neuron yang terhubung sepenuhnya. Seluruh neuron pada lapisan *FC* memiliki hubungan dengan seluruh aktivasi pada lapisan sebelumnya. Lapisan ini ditempatkan sebelum hasil klasifikasi CNN.

Layer Dropout juga diaplikasikan pada CNN untuk mengatasi *overfitting*. Teknik *dropout* mengatasi *overfitting* dengan cara menghapus sementara kontribusi terhadap aktivasi neuron ketika *forward pass*, dan setiap pembaruan bobot tidak diterapkan pada neuron ketika *backward pass*. Sumber : towardsdatascience.com

Gambar 2.5 menunjukkan bagaimana perbedaan ketika tidak menggunakan teknik dropout dan menggunakan teknik dropout.

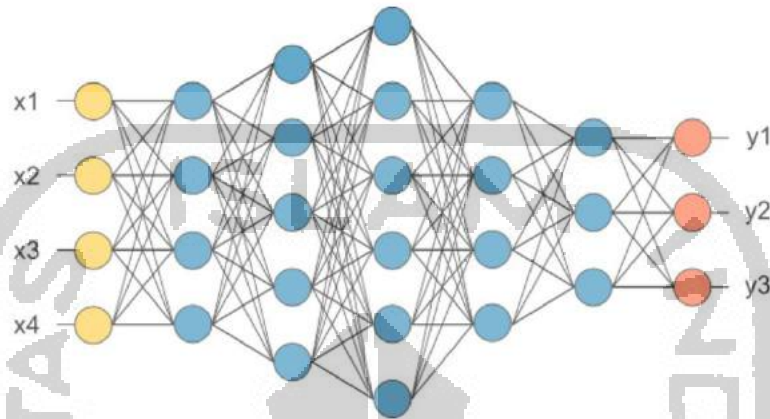


Sumber : towardsdatascience.com

Gambar 2.5 Perbedaan penggunaan *dropout layer*

Keluaran *feature maps* dari konvolusi akhir atau *layer pooling* biasanya *flattened* seperti diubah menjadi satu dimensi angka/*vector array* dan dihubungkan ke satu/lebih layer yang terhubung sempurna (*FC*) atau disebut juga *dense layer* yang ditunjukkan Gambar 2.6, dimana setiap masukan yang terhubung ke setiap keluaran memiliki bobot. Setelah melalui ekstraksi

fitur pada layer konvolusi dan dilakukan penurunan sample pada *layer pooling*, lalu keluaran dipetakan oleh *FC layer* menjadi keluaran akhir jaringan, seperti probabilitas setiap kelas pada tugas klasifikasi. Hasil akhir *FC layer* biasanya memiliki jumlah *node* yang sama dengan jumlah kelasnya.

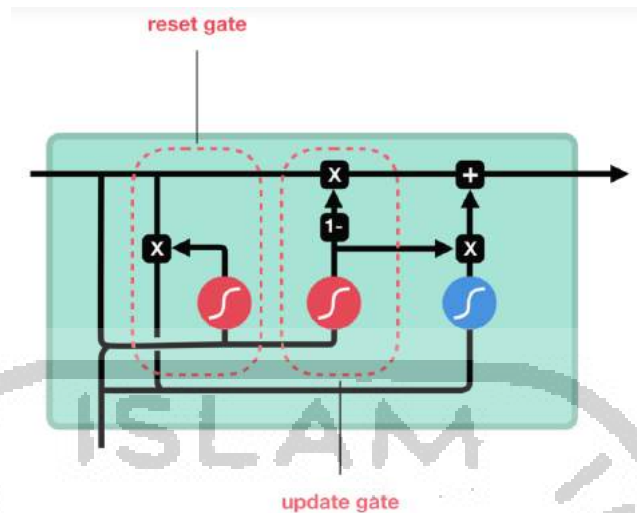


Sumber : *medium.com*

Gambar 2.6 Fully connected (FC) Layer

2.9 GRU (*Gated Recurrent Units*)

Pada GRU, hanya terdapat 2 gate yang digunakan, *reset gate* (gerbang pengulang) dan *update gate* (gerbang pembaharuan). Karena GRU merupakan generasi terbaru dari LSTM sehingga metode ini juga tidak terlalu berbeda dengan LSTM. Pada gerbang pengulang, terjadi proses yang sama seperti yang terjadi pada *forget gate* dan juga *input gate*. Gerbang ini akan memilih informasi mana yang harus disimpan atau harus dibuang. Sedangkan pada gerbang pembaharuan merupakan gerbang lain yang digunakan untuk menentukan seberapa banyak data yang perlu dilupakan/dibuang. Gambar 2.7 di bawah ini merupakan gambaran dari sel dan *gate* yang ada pada metode GRU (Nguyen, 2018).

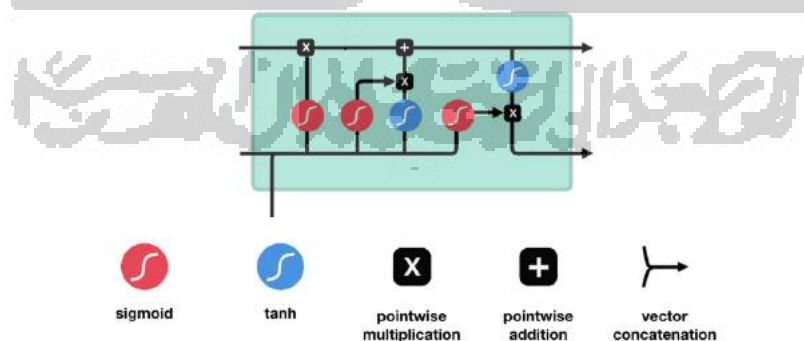


Sumber : towardsdatascience.com

Gambar 2.7 sel dan gate GRU

2.10 LSTM (Long Short-Term Memory)

RNN bersamaan dengan LSTM telah dianggap sebagai model yang efektif untuk beberapa masalah pembelajaran terkait dengan data sekuensial (Greff, Srivastava, Koutn, & Steunebrink, n.d.). Unit LSTM umumnya terdiri dari sel, gerbang input, gerbang keluaran dan gerbang lupa. Sel akan mengingat nilai selama interval waktu yang berubah-ubah dan tiga gerbang lupa lainnya mengatur aliran informasi masuk dan keluar dari sel. Gerbang lupa menentukan informasi mana yang harus dibuang atau disimpan. Untuk melakukan pembaruan sel digunakan gerbang input. Gerbang keluaran akan menentukan *hidden state* berikutnya (Nguyen, 2018). Gambar 2.8 merupakan gambaran dari LSTM sel.



Sumber : towardsdatascience.com

Gambar 2.8 LSTM Sel