Toxic Detection pada Game Dota 2: Klasifikasi Intent dan Slot Filling pada Dataset CONDA

Bagus Cipta Pratama, Kosmas Rio Legowo, Muhammad Akmal Fauzan, Rafid Nur Huda
Department of Computer Sciences and Electronics
Universitas Gadjah Mada, Yogyakarta, Indonesia

Abstract

Perilaku toksik dalam game online merupakan masalah penting yang berdampak pada pengalaman bermain dan kesehatan komunitas. Eksperimen ini mengkaji deteksi bahasa toksik pada chat *in-game* Dota 2 menggunakan dataset CONDA dengan pendekatan joint Natural Language Understanding berbasis Slot-Gated Bi-LSTM untuk klasifikasi intent dan *slot filling* secara bersamaan. Dibandingkan baseline dengan embedding acak, model dengan Word2Vec terlatih, augmentasi sinonim, class weighting, dan weight decay menghasilkan F1 tinggi pada token toksik dan slang, tetapi sedikit menurunkan Joint Semantic Accuracy (JSA) akibat *semantic drift* pada data augmentasi, sehingga menyoroti pentingnya desain augmentasi dan representasi kata yang sesuai dengan konteks bahasa dalam game.

1 Pendahuluan

Seiring dengan meningkatnya popularitas game online multiplayer *toxic behaviour* menjadi masalah serius dalam industri game. Karakteristik ujaran toxic pada chat in-game memiliki perbedaan mencolok dengan platform lain seperti media sosial atau portal berita. Pesan-pesan yang muncul cenderung sangat pendek karena pemain harus mengetik sambil bermain, sementara ujaran yang lebih panjang biasanya hanya muncul pada fase sebelum atau sesudah pertandingan. Dengan sifat pesan dalam game yang seperti ini, pemahaman pada level slot/token kata (Weld et al., 2021b) menjadi sangat penting untuk mendeteksi bahasa toxic di dalam game.

Shared task ini bertujuan untuk memahami karakteristik toksisitas dalam pesan-pesan pada game online. Karena percakapan dalam game online menyerupai percakapan bahasa alami, dibutuhkan pendekatan Natural Language Understanding (NLU) yang mengekstraksi struktur semantik melalui analisis intent dan slot. Dalam kerangka ini, deteksi intent didefinisikan sebagai klasifikasi tujuan atau maksud dari suatu ujaran (atau kalimat), sedangkan slot filling adalah pelabelan setiap kata dalam ujaran sesuai jenis informasi semantik yang dibawanya. Sejumlah literatur menunjukkan bahwa kedua tugas tersebut lebih efektif ketika dilatih secara bersama-sama, karena model dapat memanfaatkan hubungan saling melengkapi antara level intent dan level token; pelatihan bersama ini terbukti menghasilkan kinerja yang lebih baik dibandingkan ketika intent dan slot dipelajari secara terpisah (Zhang et al., 2018). Banyak pendekatan berbasis arsitektur *sequence-to-sequence* dengan RNN dan turunannya (LSTM/GRU) yang dikembangkan untuk memodelkan kedua tugas ini secara bersama, sehingga representasi konteks yang dipelajari untuk slot juga dapat menginformasikan prediksi intent, dan sebaliknya. Pendekatan-pendekatan ini menjadi landasan bagi model yang akan diadaptasi dalam penelitian ini, yaitu Slot-Gated Modeling (Goo et al., 2018) untuk deteksi bahasa toksik *in-game* berbasis intent dan slot.

2 Literature Review

Sebelum *deep learning*, *slot filling* umumnya dimodelkan sebagai *sequence labeling* (misalnya CRF), sedangkan *intent detection* memakai klasifikator terpisah. Pendekatan *pipeline* ini mengabaikan keterkaitan kuat antara intent dan slot, sehingga mendorong munculnya model *joint* berbasis RNN/LSTM.

(Liu and Lane, 2016) mengusulkan model *joint* berbasis RNN/Bi-LSTM dengan mekanisme *attention* untuk memprediksi intent dan slot secara bersamaan. Arsitektur ini menunjukkan bahwa pelatihan bersama dua task dalam satu jaringan meningkatkan akurasi intent dan F1 slot dibanding pelatihan terpisah, dan menjadi salah satu baseline awal untuk SLU berbasis *neural*.

Penelitian oleh (Wang et al., 2018) juga mem-

perkenalkan Bi-model BLSTM yang memakai dua jaringan BLSTM terpisah untuk intent dan slot, namun saling berbagi representasi tersembunyi. Pertukaran informasi ini terbukti memberi peningkatan kinerja di atas model *attention*-based sebelumnya, dan menegaskan bahwa eksploitasi hubungan intent–slot penting bagi pemahaman semantik.

(Goo et al., 2018) kemudian mengusulkan *Slot-Gated Modeling* yang secara eksplisit memodelkan interaksi antara intent dan slot melalui mekanisme *gate* di atas BLSTM dengan *attention*. Model ini menghitung representasi intent tingkat kalimat dan representasi slot per token, lalu menggabungkannya lewat *slot gate* untuk mengontrol seberapa besar informasi intent memengaruhi prediksi slot.

Optimasi dilakukan dengan memaksimalkan probabilitas bersama label intent dan urutan slot, sehingga model benar-benar belajar keterkaitan keduanya. Pada dataset ATIS dan Snips, Slot-Gated Modeling mencapai peningkatan konsisten pada akurasi intent, F1 slot, maupun *sentence-level semantic frame accuracy* dibanding Attention BiRNN (Goo et al., 2018). Hal ini menunjukkan bahwa pemodelan eksplisit hubungan intent–slot melalui *gating* lebih efektif daripada sekadar berbagi *attention* atau *hidden state*.

3 Dataset

Dataset yang digunakan adalah CONDA (CONtextual Dual-Annotated dataset for in-game toxicity understanding and detection) , yaitu korpus chat *in-game* dari gim *Dota 2* yang dirancang untuk pemodelan toksisitas berbasis intent dan slot (Weld et al., 2021a). Dataset terdapat tiga subset, yaitu conda-train (26086 ujaran), conda-validation (8706 ujaran), dan conda-test (8703 ujaran), dengan rata-rata panjang ujaran yang sangat pendek (sekitar 3–6 token), mencerminkan sifat komunikasi cepat saat bermain.

Setiap contoh data terdiri dari teks ujaran, daftar token yang telah diproses, label slot untuk tiap token, serta label intent pada level ujaran. CONDA juga menyediakan metadata konteks seperti ID pertandingan, ID percakapan, dan waktu chat.

3.1 Label Intent (Level Ujaran)

CONDA menggunakan empat kelas intent yang merepresentasikan fungsi ujaran sekaligus tingkat toksisitasnya:

- **Explicit Toxicity (E)**: ujaran dengan kata-kata kasar atau hinaan langsung.
- **Implicit Toxicity (I)**: ujaran bernada sarkastik atau merendahkan tetapi tidak eksplisit.
- **Action (A)**: ujaran yang berkaitan dengan tindakan atau koordinasi di dalam permainan.
- **Other (O)**: ujaran umum yang tidak bersifat toksik.

Table 1: Distribusi Label Intent

Label Intent	Count	Persentase (%)
Explicit	4709	13.53
Implicit	2271	6.53
Action	2299	6.61
Other	25513	73.33

Distribusinya didominasi oleh kelas O (73.33%), sedangkan toksisitas eksplisit mencakup 13.53% dan implisit mencakup sekitar 6.53% dari train dan validation dataset.

3.2 Label Slot (Level Token)

Setiap token dalam ujaran diberi label salah satu dari enam kategori:

- **T** (Toxicity): kata bernada ofensif.
- **C** (Character): nama atau singkatan hero dalam game.
- **D** (Dota-specific): istilah teknis dalam game.
- **S** (Slang): istilah slang khas komunitas.
- **P** (Pronoun): pronomina.
- **O** (Other): token lain yang tidak termasuk kategori sebelumnya.

Table 2: Distribusi Label Slot

Label Slot	Count	Persentase (%)
Toxicity	5764	4.33
Character	6422	4.83
Dota-spesific	1672	1.26
Slang	13358	10.05
Pronoun	15935	11.98
Other	75806	57.01
SEPA	14021	10.54

Mayoritas token berada pada kategori O (57.01%), diikuti P (11.98%), dan S (10.5%) dari train dan validation dataset.

4 Metodologi

Pada bagian ini akan dibahas slot-gated RNN model yang diusulkan oleh Goo et al., 2018. Arsitektur model yang dipakai adalah slot-gated

model hanya dengan intent attention ditunjukkan pada gambar 1. Selain itu, akan dibahas terkait representasi Word2Vec dan teknik augmentasi data yang digunakan.

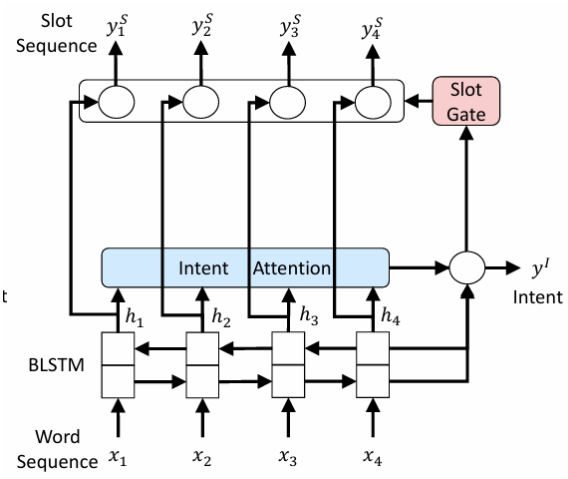

Figure 1: Slot-gated model dengan intent attention (Goo et al., 2018)

4.1 Slot-gated Bi-LSTM

Model ini terdiri dari Bi-LSTM yang memproses urutan kata $\mathbf{x} = (x_1, ..., x_T)$ sebagai input dan menghasilkan *forward hidden state* $\overrightarrow{h_i}$ dan backward hidden state $\overleftarrow{h_i}$. Hidden state akhir h_i adalah representasi konteks kata ke-i, yaitu $h_i = [\overrightarrow{h_i}, \overleftarrow{h_i}]$ (Mesnil et al., 2015).

Selanjutnya, model ini akan menghitung *context vector* untuk intent dari seluruh hidden state. Untuk setiap hidden state h_i dapat dihitung context vektor c^I sebagai penjumlahan hidden state LSTM dengan bobot attention α_j^I untuk tiap kata.

$$c^I = \sum_j^T \alpha_j^I h_j \tag{1}$$

Perhitungan vektor konteks slot c^S dilakukan dengan menjumlahkan hidden states LSTM secara berbobot menggunakan bobot attention, dan proses ini juga diterapkan dengan cara yang kurang lebih sama untuk memperoleh vektor konteks intent c^I

Setelah memperoleh hidden state h_i dan vektor konteks slot c_i^S, keduanya digunakan untuk melakukan *slot filling*. Prediksi label slot untuk kata ke-i dihitung sebagai berikut:

$$y_i^S = softmax\left(W_{hy}^S(h_i + c_i^S)\right), \tag{2}$$

di mana y_i^S merupakan label slot dari kata ke-i pada input, dan W_{hy}^S adalah matriks bobot yang dipelajari model.

Untuk prediksi intent, vektor konteks intent c^I dihitung dengan cara yang sama seperti c^S, tetapi bagian pendeteksian intent hanya menggunakan hidden state terakhir dari BLSTM, yaitu h_T. Prediksi intent dimodelkan sebagai berikut:

$$y^I = softmax\left(W_{hy}^I(h_T + c^I)\right). \tag{3}$$

4.2 Word2Vec

Pada eksperimen kedua, representasi kata dibangun menggunakan model Word2Vec yang dilatih ulang (*trained from scratch*) pada seluruh token yang berasal dari data latih dan validasi. Proses pelatihan dilakukan menggunakan *skip-gram* dengan dimensi embedding sebesar 200. Pendekatan ini memastikan bahwa vektor kata yang dihasilkan merepresentasikan konteks linguistik yang spesifik terhadap domain percakapan pada dataset CONDA, sehingga lebih efektif dibandingkan penggunaan embedding umum. Hasil pelatihan Word2Vec kemudian dimuat ke dalam layer embedding pada arsitektur Bi-LSTM sehingga model memperoleh informasi semantik awal yang lebih kaya dan stabil.

4.3 Augmentasi

Selain itu, untuk menangani ketidakseimbangan kelas serta meningkatkan keragaman data, diterapkan teknik augmentasi berbasis *synonym replacement* yang memanfaatkan WordNet. Augmentasi dilakukan secara selektif pada contoh yang memiliki intent `E`, `I`, atau `A`, serta contoh yang mengandung slot `T`, `D`, atau `S`. Hanya token dengan label `O` yang dipertimbangkan sebagai kandidat penggantian agar struktur anotasi slot tidak terganggu. Pada setiap kalimat, sebagian token tersebut diganti dengan sinonimnya untuk menghasilkan variasi kalimat baru yang secara semantik masih konsisten. Selain augmentasi, proses pelatihan juga menggunakan *weight decay* sebagai bentuk regularisasi untuk menekan overfitting dan meningkatkan generalitas model secara keseluruhan.

5 Tantangan

Dataset CONDA memiliki distribusi kelas yang sangat tidak seimbang, di mana label `O` mendominasi secara signifikan dibandingkan label target seperti `T`, `D`, dan yang lainnya. Hal ini menyulitkan model untuk menangkap pola pada kelas minoritas tanpa bantuan teknik re-weighting loss atau augmentasi.

Kemudian, penerapan teknik Data Augmentation berbasis sinonim pada code kedua menghadapi

tantangan perubahan makna semantik. Karena dataset bersifat domain-spesific (Game Slang), penggunaan thesaurus umum seperti WordNet sering kali menghasilkan kalimat sintetik yang tidak koheren dengan konteks permainan sehingga sedikit menurunkan performa model pada data validasi asli.

Terakhir, tantangan utama dalam Joint Learning adalah menyeimbangkan optimasi antara Intent Classification dan Slot Filling. Dengan metrik Joint Semantic Accuracy (JSA) yang ketat, kesalahan prediksi pada satu token slot saja akan menggugurkan prediksi intent yang sudah benar, sehingga model memerlukan konvergensi yang presisi di kedua task secara simultan.

6 Hasil dan Pembahasan

Pada bagian ini dipaparkan kinerja model yang diusulkan serta analisis trainingnya, dengan menyoroti perbedaan antara konfigurasi baseline dan model yang telah dioptimasi. Selain itu, juga akan dibahas analisis terkait terjadinya penurunan metrik tertentu pada model yang sudah dioptimasi.

6.1 Kinerja Model

Untuk mengevaluasi efektivitas arsitektur Slot-Gated Joint NLU dalam menangani tugas klasifikasi intent dan pengisian slot (ner) pada dataset CONDA, kami membandingkan dua skenario eksperimen utama:

1. Eksperimen 1 (Baseline): Menggunakan model dasar Slot-Gated Bi-LSTM dengan inisialisasi embedding acak, tanpa penanganan ketidakseimbangan kelas, tanpa augmentasi data, dan tanpa regularisasi weight decay.

2. Eksperimen 2 (Optimized): Menggunakan arsitektur yang sama, tetapi ditambah dengan empat strategi optimasi, yaitu:

   (a) Pre-trained Embeddings: Inisialisasi menggunakan Word2Vec yang dilatih pada domain-spesific
   (b) Data Augmentation: Augmentasi berbasis sinonim (WordNet) untuk memperkaya variasi data latih
   (c) Weighted Loss: Penerapan Class Weighting pada fungsi Cross Entropy untuk menangani ketidakseimbangan kelas
   (d) Regularization: Penambahan Weight Decay (1e-3) pada optimizer untuk mencegah overfitting

Tabel 3 memperlihatkan rangkuman perbandingan performa kedua model

Table 3: Perbandingan Eksperimen 1 dan Eksperimen 2

Metric	Eksperimen 1	Eksperimen 2
JSA	0.8726	0.8523
F1(E)	0.83	0.83
F1(I)	0.72	0.65
F1(T)	0.96	0.96
F1(D)	0.93	0.94
F1(S)	0.99	0.99

Berdasarkan tabel tersebut, terlihat adanya karakterisitk kinerja yang kontras. Model Baseline menunjukkan performa yang lebih seimbang secara keseluruhan, menghasilkan JSA tertinggi sebesar 87.26%. Sebaliknya, Model Eksperimen 2 mengalami penurunan pada JSA menjadi 85.23%, tetapi mencatat peningkatan konsisten pada seluruh metrik Slot Filling (T, D, dan S)

6.2 Analisis Grafik Loss dan Overfitting

Perbedaan perilaku pembelajaran kedua model terlihat jelas melalui grafik Loss selama proses pelatihan

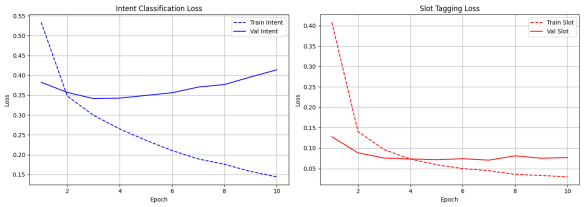

Figure 2: Kurva Loss Eksperimen 1

Pada model baseline, terlihat fenomena overfitting pada task Intent Classification. Kurva loss intent pada gambar 2 menunjukkan validation loss stagnan dan cenderung meningkat, sementara loss train terus turun mendekati nol. Hal ini mengindikasikan bahwa model terlalu menghafal data latih, tetapi model kesulitan menggeneralisasi intent pada data validasi yang belum pernah dilihat. Sementara itu, pada kurva loss slot tagging, jarak antara kurva train dan validation untuk slot relatif kecil dan tidak menunjukkan kenaikan drastis, sehingga kemampuan generalisasi untuk slot tagging masih cukup baik, meskipun model mulai overfit pada sisi intent.

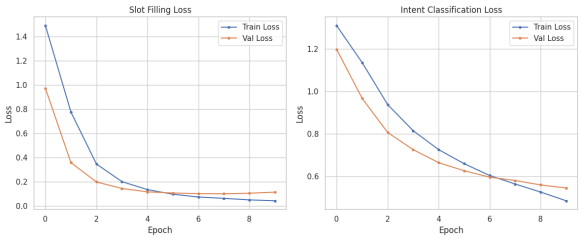

Figure 3: Kurva Loss Eksperimen 2

Sebaliknya, pada eksperimen kedua yang kami lakukan beberapa optimasi, terlihat pada gambar 3 bahwa kurva intent dan slot cenderung stabil melandai pada akhir epoch. Celah antara Train Loss dan Validation Loss relatif kecil, menandakan model yang dilakukan optimasi memiliki kemampuan generalisasi yang cukup baik pada tugas klasifikasi ini.

6.3 Analisis Penurunan Kinerja pada Eksperimen 2

Meskipun Eksperimen 2 dirancang untuk mengoptimalkan model melalui Word2Vec, augmentasi data, class weighting, dan weight decay, hasil pada Tabel 3 menunjukkan bahwa Joint Semantic Accuracy (JSA) justru sedikit menurun dibandingkan baseline. Penurunan ini dapat dijelaskan oleh beberapa faktor.

Pertama, augmentasi berbasis sinonim pada eksperimen kedua menghadapi tantangan *semantic drift*. Karena dataset bersifat sangat domain-spesifik (slang dan istilah teknis *Dota 2*), penggunaan *thesaurus* umum seperti WordNet sering kali menghasilkan kalimat sintetik yang tidak koheren dengan konteks game. Sebagai contoh, kata slang yang sangat khas untuk komunitas game dapat diganti dengan sinonim formal yang secara statistik valid tetapi tidak pernah muncul dalam percakapan *in-game*. Hal ini membuat distribusi data latih bergeser dari distribusi data validasi asli, sehingga model belajar pola yang kurang relevan dan sedikit menurunkan performa pada saat evaluasi.

Kedua, pelatihan Word2Vec dari awal pada korpus yang relatif terbatas menyebabkan kualitas embedding sangat bergantung pada frekuensi kemunculan kata. Token yang jarang muncul akan memiliki representasi vektor yang kurang stabil, sehingga tidak selalu memberikan keuntungan yang konsisten dibandingkan embedding acak yang kemudian diupdate penuh selama pelatihan. Kombinasi embedding yang tidak sepenuhnya matang dengan data hasil augmentasi yang mengalami *semantic drift* membuat model cenderung lebih hati-hati (terregularisasi) namun tidak selalu lebih tepat dalam menangkap nuansa toksisitas implisit, yang tercermin pada penurunan F1 untuk intent I dan skor JSA.

7 Kesimpulan

Penelitian ini mengkaji penggunaan Slot-Gated Bi-LSTM untuk klasifikasi intent dan *slot filling* pada dataset CONDA, korpus chat *in-game* Dota 2 yang singkat dan domain-spesifik. Pendekatan *joint learning* terbukti efektif, dengan F1 tinggi pada token toksik dan slang (misalnya F1(T) dan F1(S) ≥ 0.96) serta akurasi intent yang baik, sehingga pemodelan eksplisit hubungan intent–slot penting untuk memahami ujaran toksik dalam game.

Eksperimen kedua menambahkan embedding Word2Vec terlatih di korpus CONDA, augmentasi sinonim berbasis WordNet, *class weighting*, dan *weight decay*. Meskipun konfigurasi ini menstabilkan kurva loss dan mengurangi indikasi overfitting, Joint Semantic Accuracy (JSA) sedikit menurun akibat *semantic drift* dari kalimat sintetik yang tidak sepenuhnya selaras dengan slang dan konteks game, serta keterbatasan korpus untuk melatih embedding. Hal ini menunjukkan adanya *trade-off* antara stabilitas pembelajaran dan kinerja semantik gabungan, serta membuka ruang untuk eksplorasi teknik augmentasi yang lebih kontekstual dan pemanfaatan embedding pra-latih yang lebih kuat di masa depan.

References

Chih-Wen Goo, Guang-Lai Gao, Yun-Kai Hsu, Chih-Li Huo, Tsung-Chieh Chen, Keng-Wei Hsu, and Yun-Nung (Vivian) Chen. 2018. Slot-gated modeling for joint slot filling and intent prediction. In *North American Chapter of the Association for Computational Linguistics*.

Bing Liu and Ian Lane. 2016. Attention-based recurrent neural network models for joint intent detection and slot filling. *ArXiv*, abs/1609.01454.

Grégoire Mesnil, Yann Dauphin, Kaisheng Yao, Yoshua Bengio, Li Deng, Dilek Hakkani-Tur, Xiaodong He, Larry Heck, Gokhan Tur, Dong Yu, and Geoffrey Zweig. 2015. Using recurrent neural networks for slot filling in spoken language understanding. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, 23(3):530–539.

Yu Wang, Yilin Shen, and Hongxia Jin. 2018. A bi-model based rnn semantic frame parsing model for intent detection and slot filling. In *North American*

Chapter of the Association for Computational Linguistics.

Henry Weld, Guanghao Huang, Jean Lee, T. Zhang, Kunze Wang, Xinghong Guo, Siqu Long, Josiah Poon, and Soyeon Caren Han. 2021a. Conda: a contextual dual-annotated dataset for in-game toxicity understanding and detection. *ArXiv*, abs/2106.06213.

Henry Weld, Xiaoqi Huang, Siqi Long, Josiah Poon, and Soyeon Caren Han. 2021b. A survey of joint intent detection and slot-filling models in natural language understanding. *CoRR*, abs/2101.08091.

Chenwei Zhang, Yaliang Li, Nan Du, Wei Fan, and Philip S. Yu. 2018. Joint slot filling and intent detection via capsule neural networks. *ArXiv*, abs/1812.09471.

Lampiran

Link Google Colab

Eksperimen 1 (baseline):
https://colab.research.google.com/drive/1B7kl2bVcdPLmKSTWipfa74JmZD2QCugt?usp=sharing

Eksperimen 2 (optimized):
https://colab.research.google.com/drive/1zNjNuYSheP8Kv8jjMw62-JxjlnYdjCDX?usp=sharing

Link Shared Task

CONDA Competition:
https://codalab.lisn.upsaclay.fr/competitions/7827