Answer Sentence Selection pada Covid-19 Question Answering dengan Metode TandA

Ryan Pramana
Universitas Indonesia
Depok, Jawa Barat, Indonesia
ryan.pramana@ui.ac.id

1 Pendahuluan

Natural Language Processing (NLP) merupakan cabang dari Machine Learning yang mempelajari bagaimana komputer bisa mengerti dan memanipulasi bahasa natural manusia dalam bentuk tulisan maupun lisan untuk dimanfaatkan dalam berbagai tujuan (Chowdhury, 2003). Terdapat beberapa sub-area dalam NLP, yang salah satunya adalah Question Answering (QA). Pengaplikasian QA telah banyak bermanfaat dalam kehidupan sehari-hari, seperti digunakan di Google Assistant, Siri, maupun di search engine seperti Google search.

Dalam QA terdapat 2 task utama yaitu (Garg et al., 2020): pemilihan kalimat yang mengandung jawaban pertanyaan atau yang biasa disebut sebagai Answer Sentence Selection (AS2) dan Machine Reading (MR) atau reading comprehension yang dimana menentukan span jawaban yang tepat dari pertanyaan dan referensi teks yang diberikan. Dalam penelitian ini, akan berfokus pada task Answer Sentence Selection (AS2). Task ini tidaklah hanya mencocokkan/matching kata-kata pada question dan kandidat sentence, melainkan task ini memiliki tantangan untuk mengobservasi relasi semantik yang kompleks dan versatile antara pertanyaan dan jawaban (Bian et al., 2017). Berikut satu contoh kasus dalam AS2 (Wang and Nyberg, 2015) :

- **Question:** ”What sport does Jennifer Capriati play?”
- **Positive Sentence:** “Capriati, 19, who has not played competitive tennis since November 1994, has been given a wild card to take part in the Paris tournament which starts on February 13.”
- **Negative Sentence:** ”Capriati also was playing in the U.S. Open semifinals in ’91, one year be- fore Davenport won the junior title on those same courts”

Walaupun kedua kalimat jawaban memiliki keyword ”Capriati” dan ”play”, namun hanya kalimat pertama yang menjawab pertanyaan.

Metode untuk menyelesaikan permasalahan task AS2 ataupun masalah NLP secara umum dari waktu ke waktu terus berkembang mulai dari rule-based method sampai neural network-based method yang kini menjadi metode terpopuler dalam riset nlp atas keberhasilannya diberbagai task NLP. Metode-metode tersebut mampu menangkap hubungan antara gabungan kata-kata yang membentuk suatu makna tertentu dengan melakukan pretraining neural-network terhadap jumlah besar data (Garg et al., 2020). Menariknya pre-trained model tersebut dapat di fine-tune untuk menyelesaikan berbagai task dalam NLP.

Untuk itu dalam menyelesaikan task AS2, penelitian ini mengajukan metode TandA (Garg et al., 2020) yang merupakan metode berbasis Deep Neural Network yang memanfaatkan pre-trained Transformer model. Metode ini merupakan metode state-of-the-art pada dataset AS2 yang populer yaitu WikiQA dan TREC-QA (acl). Pada penelitian ini akan mencoba dataset yang belum pernah dicoba sebelumnya dengan metode ini, yaitu dataset dengan domain spesifik Covid-19 (COVID-QA) (Möller et al., 2020). Dalam penelitian (Garg et al., 2020) mengklaim bahwa metode ini efektif untuk beradaptasi dengan target domain serta target yang memiliki jumlah data yang kecil.

Hasil pada penelitian ini (yang dapat dilihat pada bagian eksperimen) menunjukkan metode TandA memiliki performa yang lebih baik dari baseline *fine-tunning one shot*. Kode pada penelitian ini dapat diakses pada https://github.com/ryanpram/tanda-for-as2-covidQA.

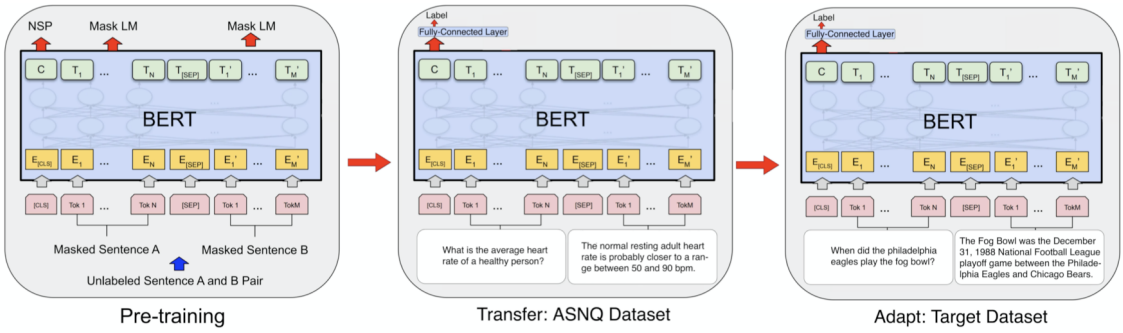

Figure 1: Alur dari TandA untuk task Answer Sentence Selection yang menggunakan BERT (Garg et al., 2020).

2 Rumusan Masalah

- Bagaimana mengimplementasikan metode TandA untuk task answer sentence selection pada spesifik domain Covid-19?
- Berapa tingkat akurasi yang dihasilkan metode TandA untuk task answer sentence selection pada spesifik domain Covid-19?

3 Tujuan

- Mengimplementasikan metode TandA untuk task answer sentence selection pada spesifik domain Covid-19
- Mengetahui tingkat akurasi yang dihasilkan metode TandA untuk task answer sentence selection pada spesifik domain Covid-19

4 Penelitian Terkait

Dalam periode-periode awal riset permasalahan AS2, riset dilakukan dengan pendekatan overlap kata-kata yang ada pada pertanyaan dan jawaban. Seperti pada penelitian yang dilakukan oleh (Wan et al., 2006) yang menggunakan metode bag-of-words dan bag-of-grams. Kelemahan dalam pendekatan di periode awal ini adalah metode-metode tersebut tidak memperhatikan nilai semantik dan fitur linguistik dari kalimat. Perkembangan riset dibidang AS2 dan Question Answering semakin cepat dengan boomingnya Artificial Inteligence. Model berbasis Deep Neural Network(DNN) menjadi populer untuk task AS2, dengan memberikan peningkatan performa yang signifikan. Arsitektur dari DNN mempelajari pattern dari answer sentence yang relevan menggunakan intra-pair dan cross-pair similarities (question-to-question dan answer-to-answer similarities) (Garg et al., 2020). Baru-baru ini (Devlin et al., 2019) dalam penelitiannya menggunakan model pre-trained Transformer untuk berbagai task di NLP, yang salah satunya adalah Question Answering (MR Task). Dalam penelitian tersebut berhasil mencapai state-of-the-art pada dataset SQUADv2.0.

Paling baru untuk task AS2, (Garg et al., 2020) menggunakan metode TandA (Transfer and Adapt) yang berhasil mencapai state-of-the-art pada dataset populer TREC-QA dan WikiQA. Metode ini berbasis Transformer dengan melatih model Tranformer melalui 2 langkah fine-tuning, yaitu Transfer dan Adapt. Transfer merupakan langkah men-transfer language model dari Transformer ke task AS2. Lalu diikuti dengan langkah Adapt untuk beradaptasi ke spesifik domain dari target dataset. Dengan 2 langkah fine-tunning yang dilakukan, *transffered* model yang dihasilkan menjadi lebih efektif dalam adaptasi ke target domain dan juga efektif walaupun target dataset yang dimiliki tidak besar.

5 Metodelogi Penelitian

5.1 Metode

Pada penelitian ini metode yang akan digunakan adalah metode TandA. TandA (Transfer dan Adapt) merupakan metode dalam melatih Transformer Model untuk task AS2 dengan melakukan 2 step fine-tunning. Proses fine-tunning tersebut tergambar pada Figure 1. Fine-tunning ditujukan untuk melakukan Transfer dari model language Transformer ke task spesifik nlp yaitu AS2. Fine-tunning dilakukan dengan menggunakan general dataset yang besar untuk task AS2. Dataset yang akan digunakan untuk proses Transfer adalah dataset Answer Sentence Natural Questions (ASNQ). ASNQ merupakan dataset yang dibuat oleh (Garg et al.,

2020) dalam penelitiannya yang ditujukan untuk memenuhi kebutuhan dataset AS2 dalam *size* besar. ASNQ dibangun dengan mentranformasikan dataset Natural Question (NQ) ke format yang sesuai task AS2.

Hasil dari model Transfer tersebut kurang optimal terhadap data pada target domain. Untuk itu dilakukan fine-tunning kedua yaitu step adapt untuk beradaptasi dengan pertanyaan-pertanyaan target domain. Contohnya, pada step Transfer banyak bertemu dengan pertanyaan general seperti "What is the average heart rate of a healthy person?", dimana nantinya pada step Adapt akan bertemu pertanyaan spesifik sesuai target domain (contohnya Covid-19) seperti "What is the incubation period of the coronavirus disease?" (Garg et al., 2020).

Pada penelitian ini diajukan menggunakan pretained Transformer model berbasis RobertTa. RoberTa dipilih karena dalam paper (Garg et al., 2020), mendapat hasil lebih baik dibandingkan pretained Transformer model berbasis Bert. RobertA merupakan *improvement* dari Bert dengan melakukan modifikasi pelatihan seperti lebih lama melatih model dengan batch dan data yang lebih besar dan menggunakan sequence yang lebih panjang (Liu et al., 2020).

5.2 Pelatihan

Untuk proses step Transfer, di penelitian ini akan langsung menggunakan model RobertTa-ASNQ hasil dari penelitian (Garg et al., 2020).Untuk panjang sequence maksimal akan diambil 128 token mengikuti paper (Garg et al., 2020). Proses pelatihan akan dioptimasi menggunakan Adam Optimizer.

Pelatihan pada penelitian ini dilakukan beberapa percobaan epoch yaitu 3, 7, dan 9. Setelah dilakukan percobaan, ternyata epoch dengan nilai 3 yang memiliki performa terbaik. Learning rate yang dipakai juga dicoba beberapa nilai yaitu 1e-6, 2e-6, 1e-5, 2e-5 pada ukuran batch 15. Dalam percobaan ditemukan nilai learning-rate yang besar menunjukkan kecenderungan performa yang lebih baik.

5.3 Dataset

Dataset yang dipakai pada penelitian ini adalah dataset dengan spesifik domain Covid-19 yaitu COVID-QA. COVID-QA dibentuk oleh 15 ahli (minimal memiliki master's degree dibidang bidomedical science) dengan menganotasi 147 artikel *scientific* terkait Covid-19 yang diambil dari CORD-19 (Möller et al., 2020). Hasil anotasi tersebut menghasilkan pasangan question/answer.

Dalam pengamatan penulis, dataset COVID-QA memiliki format yang mirip dengan SQUAD yaitu pertanyaan — konteks — posisi dari short answer. Hal ini dikarenakan memang dataset COVID-QA diutamakan untuk riset pada task Machine Reading. Oleh karena itu, dataset ini direncanakan untuk di *pre-processing* terlebih dahulu untuk diubah formatnya menyerupai dataset ASNQ. Preprocessing dilakukan dengan mentokenisasi konteks paragraf menjadi kalimat-kalimat.

5.4 Metrics

Metrics yang digunakan sering digunakan untuk mengukur performa sistem adalah Akurasi dan F1.

6 Eksperimen

Eksperimen pada penelitian ini dilakukan dengan 2 skema. Perbedaan kedua skema yang ada terletak pada cara pembagian(*split*) dataset menjadi data latih, *validation*, dan uji.

6.1 Spesifikasi Sistem

Eksperimen/percobaan pada penelitian dilakukan pada *notebook* google colab. Pelatihan dijalankan dengan GPU Tesla K80 dengan versi CUDA 11.2. Beberapa spesifikasi software lainnya yang membantu penelitian:

- Python 3.7
- datasets 1.8.0
- boto3 1.17.97
- sacremoses 0.0.45
- sentencepiece-0.1.95
- nltk 3.2.5

6.2 Skema Pertama

Pada skema pertama ini dilakukan pembagian dataset secara acak menjadi data latih, validasi, dan uji . Pembagian secara acak yang dimaksud disini adalah pembagian tidak memperhatikan adanya pertanyaan yang sama atau tidak (overlap) di ketiga set data.

6.2.1 Data

Dataset COVID-QA dilakukan pra-pemrosesan terlebih dahulu dengan melakukan tokenisasi kalimat pada teks konteks. Kalimat-kalimat hasil tokenisasi inilah yang akan menjadi kandidat jawaban dari suatu pertanyaan. Kalimat jawaban yang benar akan diberi label 1 dan apabila sebaliknya akan diberi label 0. Proses tokenisasi ini dilakukan dengan bantuan fungsi PunktSentenceTokenizer dari *library* nltk.

Setelah dilakukan pra-pemrosesan, distribusi dataset yang didapat untuk data latih adalah 81154 dan 1607 untuk label 0 dan label 1 secara berurutan. Kemudian untuk data validasi/uji sendiri mendapatkan 10142 dan 211 untuk label 0 dan label 1 secara berurutan. Distribusi dataset yang didapat disini terlihat sangat tidak berimbang (*imbalance*) dan ketidakseimbangan data ini dapat mempengaruhi performa model. Terbukti ketika dicoba data ini didapat akurasi yang sangat tinggi namun akurasi F1 yang didapat sangat rendah.

Untuk mengatasi hal tersebut penulis menggunakan metode *undersampling* terhadap data label 0. *Undersampling* yang dilakukan menghasilkan total data latih sebesar 8607, dan data validasi serta uji masing-masing sebesar 961. Statistik pada dataset akhir untuk skema pertama ini dapat dilihat pada Table 1 .

Data	**Label 1**	**Label 0**
Latih	1607	7000
Validasi	211	750
Uji	211	750

Table 1: Statistik Data Skema Pertama.

6.2.2 Hasil

Untuk optimasi *hyperparameter* dilakukan beberapa percobaan. Untuk menentukan *learning rate* terbaik dicoba beberapa nilai yaitu 1e-6, 2e-6, 1e-5 (percobaan dilakukan dengan nilai *epoch* 7). Hasil menunjukkan learning rate 1e-5 memiliki performa terbaik dengan nilai akurasi 0.89906 dan skor f1 0.84660 (lihat pada Table 2).

Setelah mendapatkan learning rate terbaik yang akan digunakan, selanjutnya dilakukan eksperimen untuk mendapatkan nilai *epoch* terbaik dengan percobaan nilai *epoch* yaitu 3, 7, 9. Percobaan ini menghasilkan nilai epoch terbaik yaitu 3 dengan akurasi 0.90218 dan skor f1 0.85216 (lihat pada Table 3).

Learning rate	**Akurasi**	**F1**
1e-6	0.89282	0.83207
2e-6	0.88762	0.82695
1e-5	**0.89906**	**0.84660**

Table 2: Ekperimen dengan nilai *learning rate* yang berbeda-beda pada skema pertama.

Epoch	***Lr***	**Akurasi**	**F1**
3	1e-5	**0.90218**	**0.85216**
7	1e-5	0.89906	0.84660
9	1e-5	0.90010	0.84790

Table 3: Ekperimen nilai *epoch* yang berbeda-beda pada skema pertama.

Hasil TandA dengan learning rate 1e-5 , ukuran batch 15, epoch 3 inilah yang menjadi hasil terbaik dari data pada eksperimen skema pertama. Sebagai perbandingan, dilakukan percobaan fine-tunning dengan teknik *one shot* yaitu *fine-tunning* untuk mempelajari target dan domain sekaligus dalam satu kali pelatihan. *One shot* dilakukan dengan model roberta-large. Ketika dibandingkan hasil metode TandA melampaui/lebih baik dari pelatihan dengan *one shot*. Hasil ini sesuai ekspektasi dimana (Garg et al., 2020) dalam papernya mengklaim metode *fine-tunning* TandA lebih baik untuk dataset dengan domain spesifik yang *low resource* daripada *fine-tunning* dengan *one shot*. Perbandingan ini dapat dilihat pada tabel Table 4.

Metode	**Akurasi**	**F1**
TandA	**0.90218**	**0.85216**
One Shot (Baseline)	0.89906	0.84368

Table 4: Perbadingan hasil metode TandA dan One shot.

6.3 Skema Kedua

Pada skema kedua ini dilakukan pembagian dataset yang berbeda dari skema pertama sehingga tidak ada pertanyaan sama yang di ketiga dataset yang ada yaiut latih, validasi dan uji. Hal ini dilakukan untuk menguji apakah model hasil *fine-tunning* bisa dengan baik memprediksi kalimat jawaban dari pertanyaan yang belum pernah ditemukan pada saat pelatihan.

6.3.1 Data

Sama halnya dengan data di skema pertama, data di skema kedua ini juga dilakukan pra-pemrosesan

dengan tokenisasi kalimat pada teks paragraf konteks. Yang berbeda disini adalah pada skema ini kalimat kandidat yang diambil dari untuk setiap pertanyaan sebanyak 3 kalimat (2 kalimat dengan label 0 dan 1 kalimat dengan label 1). Dengan begitu, data pada skema kedua ini tidak terjadi fenomena ketidakseimbangan data. Perbandingan antara data label 0 dan label 1 menjadi 2:1 dengan total pertanyaan sebanyak 1963. Statistik data pada skema kedua ini dapat terlihat pada tabel Table 5.

Data	Label 1	Label 0
Latih	1570	3140
Validasi	197	394
Uji	196	392

Table 5: Statistik Data Skema Kedua.

6.3.2 Hasil

Dalam percobaan pada skema kedua dengan TandA ditemukan bahwa performa terbaik berada di nilai *learning rate* 2e-5. Hasil perbandingan antara metode TandA dan One shot terlihat pada tabel Table 6.

Metode	Akurasi	F1
TandA	**0.90306**	**0.89582**
One Shot (Baseline)	0.87755	0.87019

Table 6: Perbadingan hasil metode TandA dan One shot Skema Kedua

Terlihat dari Table 6 bahwa TandA sekali lagi mengungguli performa dari metode *one shot*. Dari eksperimen skema kedua ini, keunggulan metode TandA dibandingkan *one shot* terlihat lebih jelas.

Untuk memperkuat uji hasil eksperimen skema kedua, penulis kemudian mencoba pengambilan data dengan label 0 yang berbeda.

Dapat terlihat di tabel Table 7 bahwa TandA sekali lagi mengungguli performa dari baseline *One shot*. Hasil dari eksperimen kedua pada skema kedua ini turun dari sebelumnya dikarenakan eksperimen ini tidak mengambil kalimat pertama pada paragraf konteks. Dimana kalimat pertama pada paragraf konteks merupakan kalimat judul yang memiliki ciri khas khusus dan hampir semuanya berlabel 0. Kehadiran kalimat dengan ciri khas khusus berlabel 0 inilah yang menurut penulis mempermudah model dalam mengidentifikasi kalimat non jawaban.

Metode	Akurasi	F1
TandA	**0.84183**	**0.83459**
One Shot (Baseline)	0.76190	0.75886

Table 7: Perbadingan hasil metode TandA dan *One shot* Skema Kedua dengan pengambilan data label 0 yang berbeda

7 Kesimpulan dan Saran Penelitian Kedepan

Dapat disimpulkan pada penelitian ini bahwa sesuai klaim dari paper (Garg et al., 2020), TandA terbukti cukup efektif untuk dataset kecil dengan domain spesifik. Terbukti juga bahwa metode *fine-tunning* TandA dapat lebih efektif daripada *fine-tunning* untuk mempelajari target dan domain secara sekaligus dalam satu pelatihan (*One shot*).

Saran kepada penelitian kedepannya :

- Mencoba metode tokenisasi kalimat yang lebih baik karena masih banyak kalimat dalam penelitian ini yang tidak tertokenisasi dengan baik.
- Mencoba beberapa penanganan lain terhadap ketidakseimbangan data seperti *oversampling* dan *weighted training*.

8 Kode penelitian

Kode pemograman pada penelitian ini dapat diakses pada https://github.com/ryanpram/tanda-for-as2-covidQA.

References

aclweb.org question answering (state of theart). https://aclweb.org/aclwiki/Question_Answering_(State_of_the_art). Accessed: 2020-04-20.

Weijie Bian, Si Li, Zhao Yang, Guang Chen, and Zhiqing Lin. 2017. A compare-aggregate model with dynamic-clip attention for answer selection. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, CIKM '17, page 1987–1990, New York, NY, USA. Association for Computing Machinery.

G. Chowdhury. 2003. Natural language processing. annual review of information science and technology. *Annual Review of Information Science and Technology*, 37:51–89.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of

deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Siddhant Garg, Thuy Vu, and Alessandro Moschitti. 2020. Tanda: Transfer and adapt pre-trained transformer models for answer sentence selection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 7780–7788.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Ro{bert}a: A robustly optimized {bert} pretraining approach.

Timo Möller, Anthony Reina, Raghavan Jayakumar, and Malte Pietsch. 2020. COVID-QA: A question answering dataset for COVID-19. In *Proceedings of the 1st Workshop on NLP for COVID-19 at ACL 2020*, Online. Association for Computational Linguistics.

Stephen Wan, Mark Dras, Robert Dale, and Cécile Paris. 2006. Using dependency-based features to take the 'para-farce' out of paraphrase. In *Proceedings of the Australasian Language Technology Workshop 2006*, pages 131–138, Sydney, Australia.

Di Wang and Eric Nyberg. 2015. A long short-term memory model for answer sentence selection in question answering. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 707–712, Beijing, China. Association for Computational Linguistics.