Machine Reading Comprehension pada Unanswerable Questions Characteristic QA dengan Metode Retrospective Reader

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

1 Pendahuluan

Question Answering(QA) merupakan area riset perpaduan dari berbagai bidang berbeda namun saling berkaitan yaitu Information Retrieval (IR), Information Extraction (IE) dan Natural Language Processing (NLP) (Bouziane et al., 2015). Dalam prespektif Information Retrieval, Question Answering merupakan bentuk *sophisticated* dari Information Retrieval yang dicirikan oleh kebutuhan informasi dalam bahasa natural manusia baik sebagian ataupun keseluruhan pernyataan (Kolomiyets and Moens, 2011). Berbeda dengan Information Retrieval Klasik, yang memberikan informasi berupa document yang relevan secara utuh (document retrieval), pada Question Answering membutuhkan bagian spesifik dari informasi untuk diberikan sebagai jawaban (Kolomiyets and Moens, 2011). Pengaplikasian dari QA sendiri 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 Comprehension (MRC) yang dimana menentukan span jawaban yang tepat dari pertanyaan dan referensi teks yang diberikan. Dalam penelitian ini, akan berfokus pada task Machine Reading Comprehension (MRC). Task ini berfungsi untuk mengambil bagian spesifik dari informasi untuk diberikan kepada *user* sebagai jawaban berdasarkan pemahaman dari suatu *passage*. Namun Task MRC memiliki challenge dimana model harus mampu untuk mendeteksi pertanyaan yang tidak memiliki jawaban (*unanswerable question*) untuk menghindari sistem QA memberikan jawaban yang ”masuk akal” tapi salah (*plausible answers*) (Zhang et al., 2020). Berikut satu contoh kasus MRC yang tidak memiliki jawaban (Zhang et al., 2020):

- **Question:** ”What cannot be solved by mechanical application of mathematical steps?”
- **Passage:** “Computational complexity theory is a branch of the the- ory of computation in theoretical computer science that focuses on classifying computational problems according to their inherent difficulty, and relating those classes to each other. A computational problem is understood to be a task that is in principle amenable to being solved by a computer, which is equivalent to stating that the problem may be **solved by mechanical application of mathematical steps, such as an algorithm**.”
- **Gold Answer:** ”(no answer)”
- **Plausible Answer:** ”algorithm”

Untuk menyelesaikan task tersebut, penelitian ini mengajukan metode Retropective Reader yang dalam penelitian (Zhang et al., 2020) meng-klaim metode ini mencapai state-of-the-art pada dataset MRC populer yaitu Squad2.0. Paper (Zhang et al., 2020) telah diterima pada AAAI Conference 2021. Metode ini terinspirasi dari bagaimana manusia menyelesaikan masalah *reading comprehension* dengan memiliki 2 tahap penyelesaian yaitu : (i) *sketchy reading* yang secara singkat mencoba melihat hubungan antara *passage* dan *question*; (ii) *intensive reading* yang memverifikasi jawaban dan memberikan prediksi final. Pada penelitian ini mengajukan penggunaan dataset yang belum di coba dengan metode ini yaitu Question Answering in Context (QuAC). Dasar pemilihan dataset QuAC adalah karena dataset ini juga memiliki pertanyaan dengan tipe *unanswerable question* dengan proporsi cukup.

2 Rumusan Masalah

- Bagaimana mengimplementasikan metode Retrospective Reader untuk task MRC pada dataset QuAC?
- Berapa tingkat akurasi yang dihasilkan metode Retrospective Reader untuk task MRC pada dataset QuAC?

3 Tujuan

- Mengimplementasikan metode Retrospective Reader untuk task MRC pada dataset QuAC
- Mengetahui tingkat akurasi yang dihasilkan metode Retrospective Reader untuk task MRC pada dataset QuAC

4 Penelitian Terkait

Penelitian dibidang Machine Reading Comprehension menjadi populer semenjak kemunculan banyak *benchmark* dataset MRC seperti SQUAD, CoQA, NewsQA dan lain-lain. Dalam periode-periode awal riset task MRC dilakukan dengan pendekatan interaksi attention. Seperti pada penelitian yang dilakukan oleh (Kadlec et al., 2016) yang menggunakan model Attention Sum Reader (AS Reader). Setelah periode tersebut, dengan *booming*nya pre-trained languange models (PrLMs) yang *powerfull* di bidang Natural Language Processing (NLP) membuat riset dibidang MRC didominasi oleh PrLMs. Seperti pada penelitian (Devlin et al., 2019) yang menggunakan pre-trained BERT model untuk fine-tune pada task MRC dengan menambahkan 1 tambahan output layer. PrLMs terbukti memberi boosting pada performa model untuk task MRC.

Walaupun sudah banyak penelitian dibidang MRC, namun model yang berfokus pada solusi MRC task dengan *unanswerable question* belum banyak memberi solusi langsung. Umumnya pendekatan yang ada adalah meng-adopsi layer answer verification tambahan, prediksi answer span, dan train secara bersamaan answer verification dengan multi-task learning (Zhang et al., 2020). Seperti pada penelitian (Liu et al., 2018) yang menambahkan token empty word ke konteks dan menambahkan simple classification layer pada reader.

Namun pada penelitian (Zhang et al., 2020) berbeda dengan pendekatan-pendekatan yang ada yang hanya men-*stack* verifier model dengan simpel. Pendekatan (Zhang et al., 2020) terinspirasi dari bagaimana manusia menyelasaikan Machine Reading Comprehension task.

5 Metode dan Detail Penelitian yang diajukan

5.1 Metode

Metode yang akan diajukan untuk dipakai adalah Retrospective Reader. Retrospective Reader memiliki arsitektur dengan 2 modul yang paralel yaitu ”Sketchy Reading Module” dan ”Intensive Reading Module”. Secara intuisi, *sketchy reading* membuat *coarse judgement*/penilaian sementara tentang *answerability* sebuah pertanyaan dan kemudian *intensive reading* bersama-sama memprediksi kandidat jawaban dan menggabungkan *answerability confidence* dengan skor penilaian sketchy untuk mendapatkan jawaban final (rear verification) (Zhang et al., 2020).

5.1.1 Sketchy Reading Module

Embedding Pada bagian embedding, *question* dan *passage* akan digabungkan sebagai input (embedding vector) untuk dijadikan input ke encoder. Encoder yang dipakai disini adalah PrLM. Output kan berupa encoded sequence (X).

Interaction Pada bagian ini, output dari encoder (X) akan di proses ke multi layer Transformer untuk mempelajari contextual representation (H).

External Front Verification Setelah melakukan ”reading” *sketchy* reader akan membuat sebuah penilaian awal dengan modul External Front Verifier (E-FV). Pada modul ini token [CLS] akan dilempar ke fully connected layer untuk mendapatkan classification logits atau regression score. Skor itu lah yang nantinya akan dipakai di rear verification untuk menentukan skor akhir.

5.1.2 Intensive Reading Module

Modul ini bertujuan untuk memverifikasi *answerability* dari suatu pertanyaan , memproduksi kandidat answer span dan memberikan prediksi jawaban final (Zhang et al., 2020).

Question-aware Matching merupakan bagian untuk memverifikasi *answerbility* dari sebuah pertanyaan. Disini akan diinverstigasi question-aware matching mechanisms dengan Cross Attention dan Matching Attention. Investigasi akan dilakukan dengan membagi representasi (H) menjadi representasi question (HQ) dan representasi passage (HP). Hasil dari bagian ini akan menghasilkan (H’).

Span Prediction merupakan bagian untuk mendapatkan answer span. Hal ini dilakukan dengan

employ layer linear dengan operasi SoftMax dan memberikan (H') sebagai Input.

Internal Front Verification Internal Front Verifier (I-FV) mirip prosesnya dengan E-FV hanya kali ini yang dilempar sebagai input adalah (H').

Rear Verification Pada bagian ini akan dihasilkan hasil akhir prediksi jawaban dengan mengkombinasikan prediksi probabilitas antara E-FV dan I-FV. Model akan mengembalikan sebuah span jawaban apabila nilai kombinasi tersebut melebihi threshold yang telah ditentukan, namun apabila sebaliknya maka model akan mengembalikan string null (no answer).

5.2 Pelatihan

Untuk proses encoder akan digunakan Pre-Language Models ALBERT-base-v2. Untuk melakukan fine-tunning akan mengikuti nilai-nilai hyperparameter yang dicoba pada paper (Zhang et al., 2020), yaitu learning-rate {2e-5,3e-5}; batch size {32,48}; epochs{2}; threshold {3.2}. Dalam eksperimen ditemukan bahwa di modul *Intensive Reading* epoch senilai 2 belum mendapatkan hasil yang memuaskan, oleh karena itu nilai epoch yang dicoba ditambah untuk modul ini yaitu {3,5,7}.

5.3 Dataset

Dataset yang dipakai pada penelitian ini adalah dataset belum dicoba sebelumnya menggunakan metode Restopective Reader yaitu QuAC (Choi et al., 2018). QuAC memiliki karekteristik yang sama dengan SQUAD2.0 yaitu mempunyai pertanyaan yang tidak tersedia jawabannya pada passage yang diberikan. Selain itu QuAC mempunyai karekteristik khusus dimana lebih banyak memiliki pertanyaan open-ended.

QuaC sedikit memiliki perbedaan format dengan SQUAD2.0, oleh karena itu harus dilakukan konversi agar model dari metode Restropective Reader dapat bekerja maksimal. Konversi rencananya akan menggunakan tool yang disediakan oleh (Yatskar, 2019)

5.4 Metrik

Untuk mengukur akurasi dari system yang akan dibangun akan digunakan metrik yang sering digunakan untuk mengukur performa sistem pada task MRC yaitu Exact Match (EM) dan F1 score.

6 Eksperimen dan Hasil

6.1 Spesifikasi Sistem

Eksperimen/percobaan pada penelitian dilakukan pada GPU NVIDIA Tesla V100 SXM2 dengan versi CUDA 10.1 Beberapa spesifikasi software lainnya yang membantu penelitian:

- tensorboard 1.14.0
- torch 1.7.0
- sacremoses 0.0.45
- boto3 1.17.99

6.2 Statistik dari Dataset

Pada penelitian ini, menggunakan data QuAC yang dapat diakses pada https://quac.ai/. Dataset QuAC hanya menyediakan 1 set data latih dan 1 set data validasi. Oleh karena itu, dalam penelitian ini akan menggunakan data validasi dari QuAC sebagai data uji dalam penelitian ini. Data validasi/uji ini tidak akan diikutsertakan dalam proses pelatihan. Pada Table 1 dibawah terlihat statistik yang digunakan dalam penelitian ini. Dalam statistik tersebut juga ditunjukkan proporsi dari pertanyaan yang tidak memiliki jawaban (*unanswerable question*) dan yang memiliki jawaban (*answerable question*). Dalam statistik tersebut terlihat bahwa setiap set dari data memiliki sekitar 20% pertanyaan yang tidak memiliki jawaban.

Dataset	***UQ***	***AQ***	**Total**
Data latih	14459	69109	83568
Data uji	780	2897	3677

Table 1: Statistik Dataset. (*UQ* merupakan *unanswerable question* dan *AQ* merupakan *answerable question*

6.3 Implementasi Eksperimen

Implementasi dilakukan dengan melakukan pelatihan 2 modul yaitu Sketchy Reading dan Intesive Reading secara paralel. Untuk pelatihan Sketchy Reading menggunakan epoch yang disarankan oleh (Zhang et al., 2020) yaitu 2. Dalam pelatihan Sketchy Reading ini berhasil menjalankan 4 eksperimen dan menghasilkan hasil seperti pada tabel Table 2

Secara bersamaan juga dilakukan pelatihan terhadap modul Intensive Reading. Karena limitasi waktu dan sumber daya gpu yang dimiliki hanya beberapa eksperimen yang dapat dilakukan

Epoch	Jumlah Batch	*Lr*	Akurasi
2	6	2e-5	79.79
2	32	2e-5	82.68
2	32	3e-5	82.32
2	64	3e-5	**82.78**

Table 2: Hasil Eksperimen Sketchy Reader

Model	F1
Bert-FlowDelta (Yeh and Chen, 2019)	65.5
HAM (Qu et al., 2019)	65.4
FlowQA (Huang et al., 2019)	64.1
Retro (penelitian ini)	40.502

Table 5: Perbandingan dengan penelitian lain

pada modul ini. Pada modul Intensive Reading memakan waktu pelatihan yang jauh lebih lama dibanding modul Sketchy Reader. Hasil dari eksperimen Intensive Reading dapat dilihat pada tabel Table 3

Epoch	Jumlah Batch	*Lr*	F1	EM
2	32	2e-5	**40.745**	**26.679**
2	32	3e-5	40.002	26.244

Table 3: Hasil Eksperimen Sketchy Reader

Hasil terbaik dari kedua modul kemudian digabungkan untuk mendapatkan hasil akhir jawaban. Langkah ini disebut dengan Rear Verification. Ambang batas yang dipakai senilai 3.2 (mengikuti konfigurasi dari penelitian (Zhang et al., 2020). Detail hasil akhirnya menjadi seperti yang terlihat pada Table 4

Metrik	Nilai
EM	26.788142507478923
F1	40.50202285051372
HasAns EM	12.840869865377977
HasAns F1	30.247130832357126
NoAns EM	78.58974358974359
NoAns F1	78.58974358974359

Table 4: Detail Hasil Akhir

Hasil ini masih sangat jauh dari harapan. Perbandingan dari hasil pada penelitian ini dengan penelitian lain dalam papan peringkat QuAC, dapat dilihat pada Table 5

7 Kesimpulan dan Saran Penelitian Kedepan

Kesimpulannya hasil dari penelitian ini dengan menggunakan metode Retropective Reader tidak berhasil menghasilkan hasil yang memuaskan dengan hanya menghasilkan F1 sebesar 40.50202285051372 dan EM sebesar 26.788142507478923. Hal ini kemungkinan disebabkan metode ini tidak mampu dengan baik menyelesaikan permasalahan MRC di dataset percakapan seperti QuAC. Walaupun gagal dengan baik menangkap jawaban pada pertanyaan yang memiliki jawaban, model dengan metode ini dapat menghasilkan hasil yang cukup baik pada pertanyaan yang tidak memiliki jawaban (*unanswerable question*) dengan nilai F1 sebesar 78.58974358974359.

Saran kepada penelitian kedepannya :

- Menambah epoch pada modul Intensive Reader karena penulis memiliki masalah keterbatasan sumber daya dan waktu untuk melakukannya. Hal ini mungkin akan memperbaiki lebih baik pada nilai EM dari model yang dihasilkan
- Mencoba lebih banyak hyperparameter (fine-tuning) pada modul Sketchy Reader untuk memperbaiki hasil yang diperoleh pada modul ini

8 Kode penelitian

Kode pemograman pada penelitian ini dapat diakses pada https://github.com/ryanpram/AwesomeMRC-QuACQA.

References

Abdelghani Bouziane, Djelloul Bouchiha, Noureddine Doumi, and Mimoun Malki. 2015. Question Answering Systems: Survey and Trends. *Procedia Computer Science*, 73:366–375.

Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wentau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. 2018. QuAC: Question answering in context. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2174–2184, Brussels, Belgium. Association for Computational Linguistics.

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.

Hsin-Yuan Huang, Eunsol Choi, and Wen-tau Yih. 2019. Flowqa: Grasping flow in history for conversational machine comprehension. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Rudolf Kadlec, Martin Schmid, Ondrej Bajgar, and Jan Kleindienst. 2016. Text understanding with the attention sum reader network. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 908–918, Berlin, Germany. Association for Computational Linguistics.

Oleksandr Kolomiyets and Marie-Francine Moens. 2011. A survey on question answering technology from an information retrieval perspective. *Information Sciences*, 181(24):5412–5434.

Xiaodong Liu, Yelong Shen, Kevin Duh, and Jianfeng Gao. 2018. Stochastic answer networks for machine reading comprehension. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1694–1704, Melbourne, Australia. Association for Computational Linguistics.

Chen Qu, Liu Yang, Minghui Qiu, Yongfeng Zhang, Cen Chen, W. Bruce Croft, and Mohit Iyyer. 2019. Attentive history selection for conversational question answering. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, CIKM '19, page 1391–1400, New York, NY, USA. Association for Computing Machinery.

Mark Yatskar. 2019. A qualitative comparison of CoQA, SQuAD 2.0 and QuAC. 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 2318–2323, Minneapolis, Minnesota. Association for Computational Linguistics.

Yi Ting Yeh and Yun-Nung Chen. 2019. Flowdelta: Modeling flow information gain in reasoning for conversational machine comprehension. *CoRR*, abs/1908.05117.

Zhuosheng Zhang, J. Yang, and Hai Zhao. 2020. Retrospective reader for machine reading comprehension. *ArXiv*, abs/2001.09694.