



دانشکده مهندسی کامپیوتر

تولید زبان طبیعی با استفاده از مدل پیش آموخته به روش یادگیری با نمونه های محدود

در رشته مهندسی کامپیوتر گرایش هوش مصنوعی

نام دانشجو:

محمد مصطفی رستم خانی - 401722235

غزل زمانی نژاد - 401722244

سید محمد عرفان موسوی منزله - 401722199

نام دستیاران:

ایمان براتی

استاد راهنما:

دکتر محمدرضا محمدی

خرداد ماه 1402

چکیده

یکی از چالش‌های تحقیقاتی مهم در کاربردهای مختلف پردازش زبان طبیعی تولید زبان طبیعی از داده‌های ساخت یافته است. پیشرفت‌های اخیر در سیستم‌های تولید زبان طبیعی مبتنی بر شبکه‌های عصبی عمیق، نیاز به مهندسی ویژگی‌ها به صورت دستی را کاهش داده و انسجام و روان بودن متن را بهبود بخشیده است. اما این شبکه‌های عمیق عموماً به مقدار انبوهی نمونه برای یادگیری وابسته هستند. ما سه مقاله پر استناد را بررسی کردیم که هر کدام با روش متفاوتی به رفع چالش کمبود داده پاسخ می‌دهند.

نویسندگان مقاله Few-Shot NLG with Pre-Trained Language Model، با پیشنهاد یک وظیفه (Task) جدید به نام تولید زبان طبیعی با نمونه‌های کم (Few-Shot)، به رفع چالش گرسنگی داده (Data Hunger) در توسعه مدل‌های تولید زبان طبیعی (NLG) می‌پردازند. آنها از چگونگی خلاصه کردن داده‌های جدولی توسط انسان‌ها الهام می‌گیرند و یک رویکرد ساده و در عین حال مؤثر ارائه می‌دهند که در عین نمایش عملکرد قوی، توانایی تعمیم به دامنه‌های گوناگون را نیز دارد. معماری این مدل بر دو جنبه کلیدی استوار است: (۱) انتخاب محتوا از داده‌های ورودی. (۲) مدل زبانی (Language Model) برای تولید جملات منسجم (Coherent). مدل زبانی با ترکیب دانش زبانی قبلی که آموخته است و محتوای انتخاب شده از داده‌ها به تولید جملات منسجم می‌پردازد. نویسندگان این مقاله، آزمایش‌هایی را با تنها 200 نمونه آموزشی از دامنه‌های مختلف انجام دادند و نتیجه آزمایش‌ها را با یک مدل مبنای قوی مقایسه کردند. نتایج آن‌ها نشان می‌دهد که روش آن‌ها با بهبود متوسط بیش از 8٪ در معیار ارزیابی BLEU از مدل مبنای پیشی می‌گیرد.

تولید داده به متن اخیراً به دلیل کاربردهای گسترده آن مورد توجه قرار گرفته است. روش‌های موجود عملکرد چشمگیری را در مجموعه‌ای از وظایف نشان داده‌اند. با این حال، آنها به مقدار قابل توجهی از داده‌های برچسب‌گذاری شده برای هر وظیفه نیاز دارند که به دست آوردن آن هزینه‌بر است و در نتیجه کاربرد آنها را در وظایف و موضوعات جدید محدود می‌کند. در مقاله Knowledge-Grounded Pre-Training سعی شده با روش شبکه پیش آموخته و یادگیری انتقالی بر این مسئله غلبه شود. KGPT شامل دو قسمت است: (1) یک مدل تولیدکننده بر پایه دانش برای تولید متن غنی شده با دانش. (2) یک الگو برای شبکه پیش آموخته بر روی حجم زیادی از متن غنی شده با دانش که از خزش وب بدست آمده. مدل ارائه شده در سه آزمایش با ناظر، بدون نمونه و با تعداد نمونه محدود برای ارزیابی میزان موثر بودن مورد بررسی قرار گرفته است. در ارزیابی تحت یادگیری با نمونه محدود، مدل می‌تواند با حدود ۱/۱۵ تعداد نمونه برچسب خورده، به همان میزان عملکرد مدل‌های مبنای دست یابد.

در سیستم‌های گفتگوی مبتنی بر وظیفه، تولید زبان طبیعی، نقش مهمی بر عهده دارد و باید کنش‌های گفتگو (dialog act) را به یک پاسخ در قالب زبان طبیعی در بیاورد. روش‌های گذشته شامل دو نوع است: (1) مبتنی بر قالب که در آن یک قالب از پیش تعیین شده که توسط متخصصین آن حوزه استخراج شده است و مدل قرار است تنها جاهایی از آن قالب را پر

کند 2) مدل‌های آماری، نیاز به تعداد زیادی داده دارای برچسب بودند و این موضوع برای موضوع‌های جدید در دنیای واقعی که داده‌های زیادی برای آنها موجود نیستند، کاربردی نیست. پس مدل باید بتواند با تعداد نمونه کم به تعمیم پذیری خوبی برسد (اهمیت یادگیری با تعداد داده‌های محدود). در مقاله Few-shot Natural Language Generation for Task-Oriented Dialog، دو چیز برای دیالوگ‌های مبتنی بر وظیفه ارائه شده است: 1) یک معیار ارزیابی به اسم FEWSHOTWOZ که یک مجموعه داده با تعداد نمونه کم برای سیستم‌های مبتنی بر وظیفه است 2) یک مدل پیش‌آمورده بر روی متن‌های زیاد برچسب گذاری شده برای تولید زبان طبیعی به صورت کنترل‌شده به اسم SC-GPT که کافی است این مدل بر روی داده‌های جدید با حوزه‌های جدید، تنظیم دقیق شود تا به نتایج بسیار خوبی دست پیدا کنیم.

واژه‌های کلیدی: یادگیری با تعداد نمونه محدود - مدل زبانی - شبکه پیش‌آمورده - تولید زبان طبیعی - تولید داده به متن

فهرست مطالب

ا	چکیده
1	فصل 1: مقدمه
4	فصل 2: مروری بر ادبیات موضوع
8	فصل 3: روش انتخابی حل مسئله
9 Few-Shot NLG with Pre-Trained Language Model
9 KGPT: Knowledge-Grounded Pre-Training
11 Few-shot Natural Language Generation for Task-Oriented Dialog
14	فصل 4: نتایج و تفسیر آنها
15 Few-Shot NLG with Pre-Trained Language Model
15 KGPT: Knowledge-Grounded Pre-Training
17 Few-shot Natural Language Generation for Task-Oriented Dialog
21	فصل 5: جمع‌بندی و پیشنهادها
22 جمع‌بندی:
22 پیشنهادها:
23	مراجع

فهرست اشکال

- شکل 1: بررسی اجمالی روش مطرح شده با سیاست سوییچ..... 9
- شکل 2: معیار مکالمه برای یکپارچه کردن ورودی داده ساختار یافته به فرم عمومی 10
- شکل 3: کدگذار گراف انتشار سلسله مراتبی (از پایین به بالا) 10
- شکل 4: کدگذاری گراف دانش به عنوان یک دنباله با استفاده از جانمایی ویژه 11
- شکل 5: تصویر مازول تولید زبان طبیعی – مثالی از کنش گفتگو..... 12
- شکل 6: تصویر SC-GPT 12

فهرست جداول

15	جدول 1 : نتایج BLEU-4 بر روی 3 دامنه
16	جدول 2 : نتایج یادگیری با نمونه محدود روی مجموعه آزمون WebNLG
16	جدول 3 : نتایج یادگیری با نمونه محدود روی مجموعه آزمون E2ENLG
16	جدول 4 : نتایج یادگیری با نمونه محدود روی مجموعه آزمون WIKIBIO
17	جدول 5 : تعداد نمونه آموزشی مورد نیاز برای رسیدن به امتیاز BLEU مشخص در مجموعه داده های مختلف ...
17	جدول 6 : مقایسه دیتاست های موجود برای NLG
18	جدول 7 : مقایسه روش های موجود بر روی FEWSHOTWOZ
18	جدول 8 : مقایسه آماری روش های موجود توسط ارزیابی انسانی
18	جدول 9 : عملکرد روی MultiWOZ
18	جدول 10 : امتیاز BLEU مدل های مختلف بر روی MultiWOZ با مقدار داده های مختلف
18	جدول 11 : ارزیابی انسانی MultiWOZ

فصل 1:

مقدمه

یکی از چالش های تحقیقاتی مهم در کاربردهای مختلف پردازش زبان طبیعی از جمله گفت و گو های وظیفه محور (Peng ۲۰۲۰, Baolin)، تبدیل جدول به متن (Chen Zhiyu, Harini Eavani, Wenhui Chen, Yinyin Liu, William Yang Wang ۲۰۱۹) و تبدیل گراف به متن (Chen Wenhui, Yu Su, Xifeng Yan, William Yang Wang ۲۰۲۰) تولید زبان طبیعی از داده های ساختار یافته است (Gatt Albert, Emiel Krahmer ۲۰۱۸). پیشرفت های اخیر در سیستم های تولید زبان طبیعی مبتنی بر شبکه های عصبی عمیق، نیاز به مهندسی ویژگی ها به صورت دستی را کاهش داده و انسجام و روان بودن متن را بهبود بخشیده است. اما این شبکه های عمیق عموماً به مقدار انبوهی نمونه برای یادگیری وابسته هستند. برای مثال مجموعه داده WIKIBIO دارای ۵۰۰ هزار نمونه آموزشی فقط برای یک دامنه است (Lebret ۲۰۱۶, Remi, David Grangier, Michael Auli). وابستگی شدید به دیتای آموزشی در این مدل ها، ما را در جهت مطرح کردن دو سوال تحقیقاتی بنیادی هدایت می کند: (۱) آیا میتوان با حفظ عملکرد در سطح معقول، میزان داده های جمع آوری شده برای آموزش را کاهش داد؟ (۲) آیا میتوان از مدل های پیش آموخته برای تولید خودکار متن زبان طبیعی از داده های ساختار یافته به نحوی استفاده کرد که با کمترین تعداد نمونه های ممکن در دامنه های جدید قابل استفاده باشد؟ برای پاسخ گویی به این دو سوال بنیادی، ما سه مقاله پر استناد را بررسی کردیم که هر کدام با روش متفاوتی به این سوالات پاسخ می دهند.

۱-۱) Few-Shot NLG with Pre-Trained Language Model (Chen Zhiyu, Harini Eavani, Wenhui Chen, Yinyin Liu, William Yang Wang 2019)

این مقاله با ایده برداری از روشی که انسان داده های درون جدول را به متن تبدیل می کند، یک سامانه خودکار دو مرحله ای برای تبدیل جدول به متن طبیعی پیشنهاد می دهد: (۱) انتخاب و رونوشت محتوا از جدول. (۲) نوشتن جملات منسجم و روان بر پایه محتوای انتخاب شده از جدول. مرحله اول این سامانه وابسته به دامنه مسئله است و باید برای هر دامنه به طور جداگانه یادگرفته شود. در مرحله دوم میتوان از یک مدل زبانی از پیش آموخته برای ایجاد جملات منسجم و روان استفاده کرد. این مدل زبانی دانش قبلی قوی در مورد ساخت جملات روان و منسجم ارائه می دهد که میتوان آن را با یک عملکرد انتخاب از میان دانش مدل یا محتوای رو نوشت شده ترکیب کرد تا با یادگیری از تنها چند نمونه آموزشی، مدل شبکه عصبی تولید متن زبان طبیعی را از وابستگی به تعداد نمونه زیاد رها کنیم. نتایج آزمایش های این مقاله نشان می دهد که روش های موجود که به مجموعه داده های آموزشی بزرگ متکی هستند در تنظیمات با تعداد نمونه اندک در مقایسه با روش پیشنهاد شده ضعیف عمل می کنند. از طرف دیگر روش پیشنهادی با تعداد نمونه های اندک می تواند در دامنه های جدید استفاده شود.

۱-۲) KGPT: Knowledge-Grounded Pre-Training for Data-to-Text Generation (Chen Wenhui, Yu Su, Xifeng Yan, William Yang Wang 2020)

ایده اصلی این مقاله پیش آموزش یک مدل زبانی بر روی داده های جمع آوری شده از سطح وب است. سپس مدل زبانی پیش آموخته با توجه به دانشی که از این داده ها بدست آورده است در وظایف پایین دستی در تنظیمات با داده های محدود مورد استفاده قرار می گیرد. نویسندگان مقاله با جمع آوری داده های ویکی پدیا و ایجاد یک گراف دانش از این داده ها، یک مجموعه داده بدون برچسب با ۱.۸ میلیون زوج داده ایجاد کرده اند. در نهایت یک مدل زبانی را بر اساس داده های گراف دانش آموزش داده اند. مدل نهایی KGPT نام دارد و توانایی این را دارد که به گستره وسیعی از وظایف و دامنه ها تعمیم داده شود. نویسندگان این مقاله مدل خود را در هر یک از سه تنظیمات زیر امتحان کردند: (۱) تعداد زیادی داده برچسب گذاری شده (۲) بدون هیچ داده برچسب گذاری شده (۲) تعداد اندکی داده برچسب گذاری شده. آزمایش های این گروه نشان داده است که

مدل ایشان می تواند تا ۱۵ برابر نیاز به برچسب گذاری داده ها را کاهش دهد و کماکان به امتیاز ۳۰ درصدی در معیار ارزیابی BLEU برسد.

(۱-۳) Few-shot Natural Language Generation for Task-Oriented Dialog (Peng Baolin 2020)

در یک سامانه گفت و گو محور مبتنی بر وظیفه، مازول تولید زبان طبیعی نقش بسیار مهمی ایفا می کند. این مازول مسئولیت تبدیل یک کنش سیستمی به یک پاسخ نهایی در زبان طبیعی را دارد. این پاسخ ها باید منسجم باشند تا کنش های معنایی را برساند و همچنین روان باشد تا توجه مخاطب را برانگیزد. روش های تولید زبان طبیعی به دو دسته تقسیم بندی می شوند: (۱) روش های مبتنی بر قالب که نیازمند متخصص های مربوط به آن زمینه است تا قالب های مربوط به هر زمینه را به صورت دست ساز بسازند و سیستم جاهای خالی را پر می کند، بنابراین این روش شامل اطلاعات لازم هست ولی ممکن است روان نباشد که این برای سیستم مناسب نیست. (۲) مدل های زبانی آماری مانند شبکه های عصبی که با داده های برچسب زده، یاد می گیرند به صورت روان پاسخ دهند. یکی از این مدل ها Tsung-Hsien Wen, Milica Gasic, Dongho Kim, Nikola Mrksić, Pei-Hao Su, David Vandyke, Steve Young 2015) است که کنش های دیالوگ را به صورت یک بردار one-hot در آورده و از آن به عنوان یک ویژگی اضافه استفاده می کند. علیرغم کارایی خوب این روش در دامنه های ساده، این روش نیاز به داده های برچسب گذاری شده زیادی در زمینه مربوطه دارد که در دنیای واقعی و همچنین در دامنه های جدید امکان پذیر نیست. برای ارزیابی مدل ها در تنظیمات نمونه های اندک، آن ها دیتاست FEWSHOTWOZ را معرفی کرده اند که یک مجموعه داده معیار جدید است که بر اساس دو مجموعه داده MultiWOZ و Cambridge NLG ساخته شده است. این مجموعه داده شامل دیالوگ هایی در ۷ زمینه است و برای هر زمینه کمتر از ۵۰ داده برچسب گذاری شده برای fine-tuning وجود دارد. مدل SC-GPT را نیز برای چالش مطرح شده ارائه دادند، که طی سه مرحله آموزش دیده بود:

- (۱) پیش آموزش بر روی متن های ساده مانند GPT-۲
- (۲) پیش آموزش بر روی مقدار زیادی داده برچسب گذاری شده برای dialog-act تا مدل بتواند توانایی تولید متن کنترل شده را به دست بیاورد.
- (۳) تنظیم دقیق در زمینه مورد نظر با تعداد محدودی داده. این مدل بر خلاف SC-LSTM می تواند پاسخ هایی مشروط به یک فرم معنایی تولید کند.

فصل 2:

مروری بر ادبیات موضوع

تولید زبان طبیعی از داده های ساختار یافته

از آنجا که تولید زبان طبیعی یکی از اهداف اصلی در بسیاری از بخش های NLP است، تولید متن از روی داده ها و یا دانش ساختاریافته برای سال ها مورد مطالعه قرار گرفته است. سیستم های سنتی از الگوی pipeline پیروی می کنند که به صورت صریح تولید متن را به انتخاب محتوا، برنامه ریزی کلان و خرد و تحقق سطحی تقسیم می کند (Kukich 1983, Ehud (Reiter, Robert Dale 1997). چنین روشی تا حد زیادی به قالب ها و ویژگی هایی که توسط انسان مهندسی شده متکی است.

کارهای زیادی برای مقابله با ماژول های جداگانه پیشنهاد شده است، مانند (Percy Liang, Michael I. Jordan, Dan (Wei Lu 2009) (Marilyn A. Walker, Owen Rambow, Monica Rogati 2001) (Klein 2009) اخیراً با پیشرفت یادگیری عمیق، از شبکه های عصبی انتها به انتها استفاده می شود، (Tianyu Liu, Kexiang Wang, Lei Sha, Baobao Chang, Zhifang Sui 2018) (Sam Wiseman, Stuart M. Shieber, and Alexander M. Rush (Ratish Puduppully, Li Dong, Mirella Lapata n.d.) (2018). این شبکه ها توانسته اند به عملکرد قابل توجهی

بر روی مجموعه داده های بزرگ مانند

E2E challenge (Ondřej Dusek, Jekaterina Novikova, Verena Rieser 2019), WebNLG challenge (Claire Gardent, Anastasia Shimorina, Shashi Narayan, Laura Perez-Beltrachini 2017) and WIKIBIO (Remi Lebret, David Grangier, Michael Auli 2016), ROTOWIRE (Sam Wiseman, Stuart Shieber, Alexander Rush 2017), TOTTO (Ankur P Parikh, Xuezhi Wang, Sebastian Gehrmann, Manaal Faruqui, Bhuwan Dhingra, Diyi Yang, Dipanjan Das n.d.) Log-icNLG (Wenhu Chen, Jianshu Chen, Yu Su, Zhiyu Chen, William Yang Wang 2020) دست یابند. با این وجود، روش یادگیری با ناظر نیازمند تعداد زیادی داده برچسب خورده برای وظیفه مشخص است. در مقاله (Hady ElSahar, Christophe Gravier, and Fred' erique ' Laforest 2018) یادگیری با نمونه های محدود برای تولید سوال از گراف های دانش پیشنهاد می شود. در مقاله (Shuming Ma, Pengcheng Yang, Tianyu Liu, Peng Li, Jie Zhou, Xu Sun 2019) تولید متن از روی جدول با 1000 نمونه جفت شده و نمونه های هدف در مقیاس بزرگ پیشنهاد می شود.

مدل های زبانی پیش آموخته

بسیاری از بهترین روش های فعلی، ترکیبی از پیش آموزش و به دنبال آن تنظیم دقیق نظارت شده، با استفاده از داده های مخصوص وظیفه را اتخاذ می کنند. سطوح مختلف پیش آموزش عبارتند از جانمایی کلمه (Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean 2013) (Jeffrey Pennington, Richard Socher, and Christopher D. Manning 2014) (Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, Luke Zettlemoyer 2018) جانمایی جمله (Quoc V. Le, Tomas Mikolov 2014) (Ryan Kiros, Yukun Zhu, Ruslan Salakhutdinov, Richard S. Zemel, Raquel Urtasun, Antonio Torralba, Sanja Fidler 2015) و اخیراً، پیش آموزش مبتنی بر مدل سازی زبان مانند (Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova

GPT-2 (Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, Ilya و 2018) BERT
 CTRL (Nitish Shirish Keskar, Bryan McCann, Lav R Varshney, Caiming و Sutskever 2019)
 Grover (Rowan Zellers, Ari Holtzman, Hannah Rashkin, و Xiong, Richard Socher 2019)
 Yonatan Bisk, Ali Farhadi, Franziska Roesner, Yejin Choi. 2019). یکی از معایبی که GPT-2 دارد این
 است که توانایی کنترل معنایی سطح بالا در تولید متن ندارد. این قضیه در CTRL کمتر شده و به این صورت بود که بر
 اساس نوع نگارش و توصیف محتوا میتوان متن تولید کرد. Grover برای تولید متن مشروط به نویسنده و تاریخ است.
 چنین مدل هایی از قبل بر روی مجموعه های دامنه باز در مقیاس بزرگ آموزش داده شده اند و همزمان با افزایش عملکرد،
 وظایف پایین دستی را با دانش قبلی غنی ارائه می کنند .
 یکی دیگر از این مدل ها Switch-GPT-2 (Zhiyu Chen, Harini Eavani, Wenhui Chen, Yinyin Liu, و William Yang Wang 2020) است که از GPT-2 پیش آموخته به عنوان کدگشا برای تولید متن از روی جدول
 استفاده می کند.

مدل های زبانی مبتنی بر دانش

مدل های زبانی عصبی نشان داده اند که می توانند به خوبی وقوع n-gram ها را در جمله پیش بینی کنند اما در تولید متن
 سازگار با حقایق دنیا موفق نبوده. برای برطرف کردن این مسئله، مدل های زبانی مبتنی بر دانش (Sungjin Ahn,
 Heeyoul Choi, Tanel Parnamaa, Yoshua Bengio 2016) (Hiroaki Hayashi, Zecong Hu,
 Chenyan Xiong, Graham Neubig 2020) (Robert Logan, Nelson F Liu, Matthew E Peters,
 Matt Gardner, Sameer Singh 2019) پیشنهاد شده اند تا بتوانند دانش ساختاریافته را در مدل زبانی دخیل کنند.

دیالوگ

سیستم های دیالوگ مبتنی بر وظیفه ای همچون Rasa¹ و Microsoft Bot Framework² و Conversational
 Learner³ و chitchat system شامل Xiaocle (Li Zhou, Jianfeng Gao, Di Li, Heung-Yeung Shum)
 بدون تاریخ و DialoGPT (Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris)
 Meena (Daniel و Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, Bill Dolan 2019)
 Adiwardana, Minh-Thang Luong, David R So, Jamie Hall, Noah Fiedel, Romal Thoppilan, Zi
 Yang, Apoorv Kulshreshtha, Gaurav Nemade, Yifeng Lu, et al 2020). برای بخش NLG انتخاب های
 زیادی داریم شامل RNN (Tsung-Hsien Wen, Milica Gasic, Dongho Kim, Nikola Mrksić, Pei-) ها و attention RNN (Hao Su, David Vandyke, Steve Young 2015)
 Ondřej Dusek, Filip Jur)

¹ <https://rasa.com/>

² <https://dev.botframework.com/>

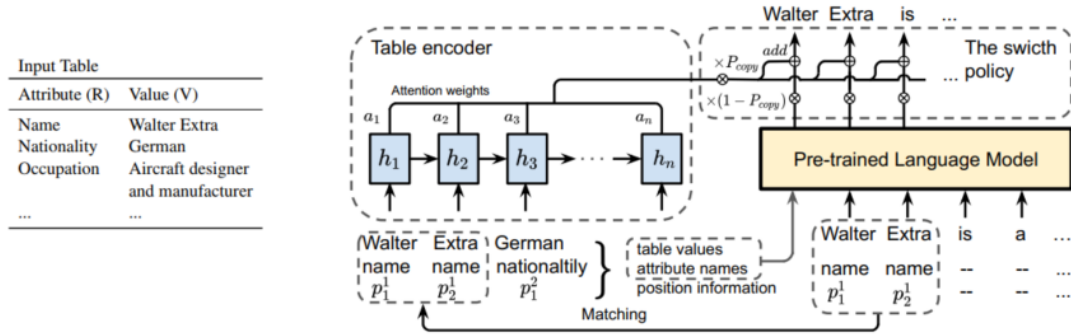
³ <https://www.microsoft.com/enus/research/project/conversation-learner/>

Tsung-Hsien Wen, Milica Gasic, Nikola Mrkšić, Pei-Hao (c) 2016 SC-LSTM ها و (Su, David Vandyke, Steve Young 2015) و انواع آن. همه این مدل ها به مقدار زیادی داده برچسب خورده نیاز دارند تا به کارایی قابل قبولی برسند.

فصل 3:

روش انتخابی حل مسئله

Few-Shot NLG with Pre-Trained Language Model



شکل 1: بررسی اجمالی روش مطرح شده با سیاست سوئیچ

دیتای ورودی در این مقاله یک جدول با دو ستون است. ستون اول ویژگی و ستون دوم مقدار ویژگی است. میتوانیم این جدول را به این صورت فرمول بندی کنیم: $\{R_i: V_i\}$ هر دوی R_i و V_i میتوانند از جنس رشته، عدد، عبارت یا جمله باشند.

هر R_i به صورت دنباله ای از کلمات در نظر گرفته می شود. $V_i = \{v_j\}_{j=1}^m$

برای هر کلمه v_j یک نام ویژگی R_i و اطلاعات مرتبط با موقعیت آن کلمه در V_i را نیز نگهداری می کنیم. هدف ایجاد یک توضیح متنی بر پایه این زوج داده ها است. برای دستیابی به این هدف ابتدا یک مدل کدگذار داده های جدول را کدگذاری می کند سپس به کمک یک مدل زبانی به صورت زیر کدگذاری انجام می شود: در هر قدم یک سوئیچ با احتمال p_{copy} مشخص می کند که آیا داده های جدول در خروجی ظاهر شوند یا بر اساس توزیع احتمال مدل زبانی کلمه بعدی تولید شود.

$$p_{copy} = \text{sigmoid}(W_c \cdot c_t + W_s \cdot s_t + W_x \cdot x_t + b)$$

$$c_t = \sum_i a_t^i \cdot h_i$$

$\{h_i\}$ حالت مخفی کدگشا، x_t ورودی کدگشا، s_t بردار حالت، a_t بردار توجه

مکانیزم سوئیچ یاد میگیرد که بین تولید متن از دانش قبلی مدل زبانی و انتخاب مستقیم کلمه از جدول ورودی در هر مرحله کدام را انجام دهد. مزیت بزرگ این روش تولید کلمات جدیدی در خروجی است که در جدول موجود هستند اما در واژه نامه مدل زبانی وجود ندارند. برای آموزش این مدل، باید دقیقاً به مدل نشان داد که چه زمانی از جدول رونوشت بگیرد و چه زمانی با مدل زبانی تولید کند. برای دستیابی به این هدف از تابع ضرر زیر استفاده می کنیم:

$$L = L_c + \lambda \sum_{w_j \in m, m \in \{V_i\}} \{1 - p_{copy}^j\}$$

با حداقل کردن تابع ضرر بالا، L_c که ضرر اصلی بین خروجی تولید شده و هدف است کاهش می یابد و احتمال تولید کلمه w_j که در بین کلمات درون جدول متناظر با ویژگی داده شده است افزایش میابد.

KGPT: Knowledge-Grounded Pre-Training

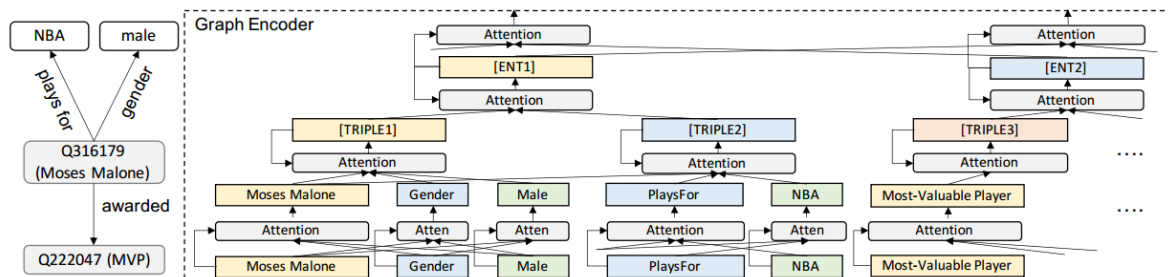
تنظیمات مسئله: در این مقاله، داده ها را ورودی های ساختار یافته با فرمت های متنوع در نظر می گیریم. ابتدا آنها را به یک ساختار دیکشنری واحد تبدیل می کنیم که کلیدها نشان دهنده موضوعات و مقادیر نشان دهنده جفت های مسند-مفعول مرتبط با موضوع است.

WebNLG	<code><triple> Stuart_Parker_(footballer) club Chesterfield_F.C. 1_Decembrie_1918_University nickname Uab.</code>	→	Stuart Parker: [(club, Chesterfield F.C.), ...], 1 Decembrie 1918 University: [(nickname, Uab), ...]								
E2ENLG	<code>name[The Eagle], eatType[coffee shop], priceRange[moderate]</code>	→	The Eagle: [(eat type, coffee shop), (price range, moderate)]								
WikiBio	<table border="1"> <tr> <th>Born</th><th>Education</th><th>Employer</th><th>Article</th></tr> <tr> <td>September 1972</td><td>Northwestern</td><td>Houston Rockets</td><td>Morey</td></tr> </table>	Born	Education	Employer	Article	September 1972	Northwestern	Houston Rockets	Morey	→	Daryl Morey: [(Born, 1972), (Education, Northwestern), (Employer, Houston Rockets), ...]
Born	Education	Employer	Article								
September 1972	Northwestern	Houston Rockets	Morey								

شکل 2: معیار مکالمه برای یکپارچه کردن ورودی داده ساختار یافته به فرم عمومی

کدگذار: از دو نوع کدگذار استفاده می کنیم:

کدگذار گراف: این کدگذار بر اساس شبکه توجه گراف برای رمزگذاری صریح اطلاعات ساختار است. به طور خاص، هر مفعول، مسند و موضوع را به عنوان گره های برگ در نظر می گیریم. همچنین [ENT] و [TRIPLE] را به عنوان شبه گره ها به هدف ارسال پیام اضافه می کنیم. گراف بدست آمده در تصویر مشاهده می شود:



شکل 3: کدگذار گراف انتشار سلسله مراتبی (از پایین به بالا)

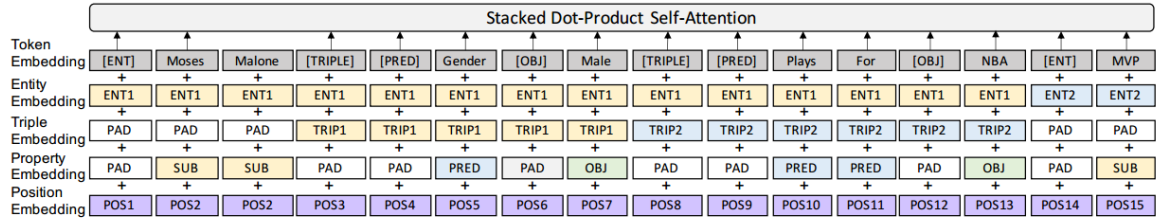
در ابتدا، هر گره را با میانگین امبدینگ زیرکلمه های آن واحدها مقداردهی اولیه می کنیم. مثلاً گره 'Moses Malone' با $(E[Mos] + E[es] + E[Ma] + E[lone]) / 4$ مقداردهی اولیه می شود. بعد از بدست آوردن مقداردهی اولیه گره ها، از انتشار پیام برای به روز رسانی نمایش گره ها بر اساس اطلاعات همسایه استفاده می کنیم. در لایه اول، اطلاعات میان گره های داخل یک triple را تبادل می کنیم. در لایه دوم، اطلاعات را از گره های sub/pred/obj به گره های [TRIPLE] جمع می کنیم. در لایه سوم، اطلاعات را از گره های مختلف [TRIPLE] به گره های [ENT] جمع می کنیم. در لایه چهارم، اطلاعات را از گره های مختلف [ENT] برای افزایش تعاملات cross-entity تبادل می کنیم. برای به روز رسانی نمایش گره i ام با شبکه توجه چند-سر که اطلاعات گره های همسایه را جمع می کند، از فرمول زیر استفاده می شود:

$$\alpha_j^m = \frac{e^{(w_Q^m g_i)^T (w_K^m g_j)}}{\sum_{j \in N_i} e^{(w_Q^m g_i)^T (w_K^m g_j)}}$$

$$v = \text{concat}[\sum_{j \in N_i} \alpha_j^m w_v^m(g_j)]$$

$$g_i = \text{LayerNorm}(MLP(v + g_i))$$

کدگذار دنباله: این کدگذار بر اساس ترنسفورمر با یک امبدینگ خاص به عنوان ورودی کمکی برای تزریق اطلاعات ساختاری به مدل دنباله ای است. مفهوم امبدینگ خاص در ابتدا توسط مقاله BERT پیشنهاد شد. از امبدینگ entity، امبدینگ triple و امبدینگ property برای کدگذاری ساختار زیرگراف در قالب یک دنباله خطی استفاده شده است.



شکل 4: کدگذاری گراف دانش به عنوان یک دنباله با استفاده از جابجایی ویژه

در مقایسه با کدگذار گراف، کدگذار دنباله ساختار را به عنوان یک قید سخت اعمال نمی کند و انعطاف بیشتری برای مدل به همراه دارد تا بتواند تعاملات cross-entity و cross-triple را اجرا کند. ضرب داخلی توجه به خود مطابق با تعریف آن در مقاله Transformer است:

$$f_{att}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{D}\right)V$$

$$G_m = f_{att}(QW_Q^m, KW_K^m, VW_V^m)$$

$$G = \text{MLP}(\text{Concat}(G_1, \dots, G_m))$$

where Q, K, V are computed from the input embeddings

m represents m – th head and f_{att} is the core attention function

$$\alpha_j = \frac{e^{o_i^T G_j}}{\sum_j e^{o_i^T G_j}}, p_{gen} = \sigma(\text{MLP}(o_i))$$

$$P(y_i = w) = p_{gen} P_{voc}(w) + (1 - p_{gen}) \sum_{j: x_j = w} \alpha_j$$

$.o_i$ is the last layer hidden state of decoder

α_j is the copy probability over the whole input token seq.

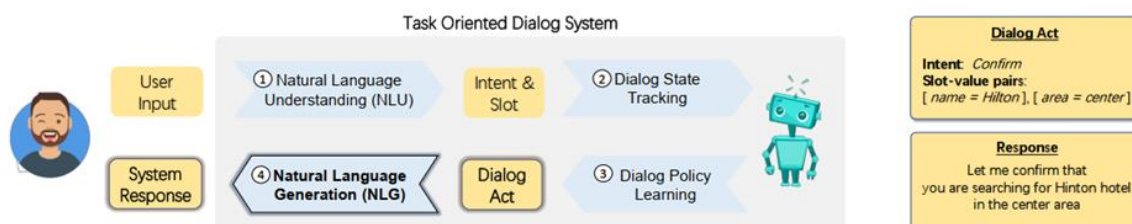
کدگشا: معماری کدگشا بر اساس Transformer و مکانیزم کپی است. در هر زمان، مدل دارای یک گیت کپی p_{gen} است که تعیین می کند y_i باید از مجموعه کلمات ساخته شود و یا از توکن های ورودی کپی شود.

Few-shot Natural Language Generation for Task-Oriented Dialog

هر نوبت گفتگو با استفاده از یک روش چهار مرحله ای پردازش میشود: (1) ورودی کاربر ابتدا به ماژول درک زبان طبیعی (NLU) منتقل میشود و قصد کاربر و سایر اطلاعات کلیدی استخراج میشود. (2) سپس این اطلاعات به عنوان ورودی ردیابی وضعیت گفتگو (Dialog State Tracking-DTS) منتقل میشود و وضعیت فعلی گفتگو را حفظ میکند. (3) سپس خروجی DTS به عنوان ورودی به ماژول سیاست گفتگو منتقل میشود و در آنجا یک dialog-act تولید میشود. (4) سپس خروجی قسمت قبل به عنوان ورودی قسمت NLG وارد میشود و پاسخ سیستم تولید میشود.

$$dialog act: A = [I, (s_1, v_1), \dots, (s_p, v_p)]; I: Intent, (s, p): slot - value pairs$$

قصد، در واقع نوع کنش های سیستمی متفاوت را نشان می دهد. برای مثال اطلاع دادن، درخواست کردن، تایید کردن و انتخاب و ... جفت شکاف-مقدار نشان دهنده دسته و محتوای اطلاعات برای بیان در گفته است.



شکل 5: تصویر مازول تولید زبان طبیعی - مثالی از کنش گفتگو

هدف تولید زبان طبیعی، ترجمه dialog act به یک پاسخ زبان طبیعی است.

ترنسفورمرهای تولید کننده پیش آموخته شرطی معنایی:

روش حل مسئله در این مقاله استفاده از مدل های زبانی شرطی است که با استفاده از N نمونه آموزشی،

هدف این است که یک مدل آماری با پارامترهای θ بسازیم تا بتواند $P(x|A)$ را بدهد (بر اساس توکن های قبل از t).

$$p_{\theta}(x|A) = \prod_{t=1}^T p_{\theta}(x_t|x_{<t}, A)$$

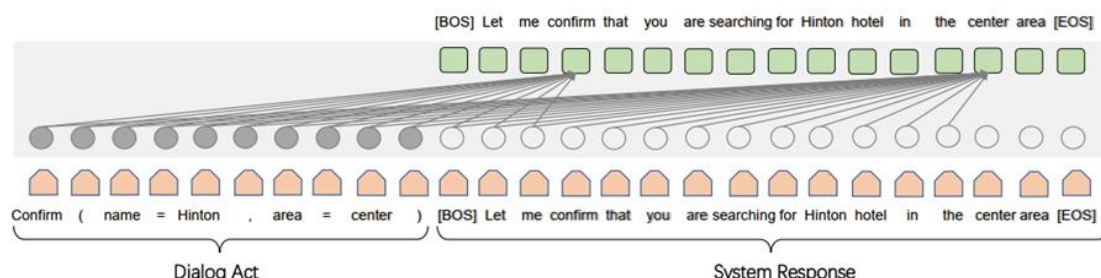
یادگیری نیز شامل پیدا کردن مقادیری برای θ است که بتواند MLE را حداکثر کند.

$$L_{\theta}(D) = \sum_{n=1}^{|D|} \sum_{t=1}^{T_n} \log p_{\theta}(x_{t,n}|x_{<t,n}, A_n)$$

برای مدل اصلی پیشنهاد شده، از معماری GPT-2 استفاده شده است.

آموزش شامل سه مرحله بوده است:

- 1) پیش آموزش بر روی متن های ساده زیاد
- 2) پیش آموزش کنترل شده: dialog-act: مدل خود را بر روی داده های بسیار زیاد برچسب گذاری شده، آموزش دادند. برای آموزش مدل از مجموعه دادگان Schema-Guided Dialog و MultiWOZ و Frame و Facebook Multilingual Dialog استفاده کرده اند. تعداد داده های برچسب گذاری شده شامل 400 هزار نمونه است.
- 3) تنظیم دقیق: برای دامنه های جدید، مدل خود را تنها بر روی تعداد کمی از داده های برچسب گذاری شده در آن زمینه، fine tune میکنیم.



شکل 6: تصویر SC-GPT

از مزایای مدل ارائه شده میتوان به (1) قابلیت انعطاف (2) قابلیت کنترل (3) قابلیت تعمیم پذیری اشاره کرد.

از معایبی که باعث شد تا مجموعه داده معیار جدیدی معرفی کنند این بود که مجموعه دادگان گذشته، مقدار زیادی داده برچسب گذاری شده داشتند که در واقعیت اینگونه نیست و علاوه بر آن، درصد دیالوگ حذف شده متمایز بین داده های آموزشی و تست بسیار کم است برای مثال دیالوگ های حذف شده در آزمون، در مجموعه داده E2E به صورت 100٪ پوشش داده میشود و این باعث میشود تا در ارزیابی تعمیم مدل برای زمینه های جدید دچار مشکل شویم. FEWSHOTWOZ دارای زمینه های بیشتری است و علاوه بر آن نمونه های آموزشی در آن کمتر است و همپوشانی داده های آموزشی و تست نیز بسیار کم است .

فصل 4:

نتایج و تفسیر آنها

Few-Shot NLG with Pre-Trained Language Model

در جدول شماره یک نتایج معیار ارزیابی BLEU-4 در قیاس با کار قبلی که بهترین نتایج تا امروز را داشته است قرار داده شده است. همانگونه که مشاهده می شود، کار قبلی در تنظیمات با داده های اندک کارایی خوبی از خود نشان نداده است. این مدل توانایی تولید داده های موجود در جدول را به خوبی ندارد و به سرعت دچار بیش برازش می شود. همانگونه که مشخص است نسبت به حالتی که از قالب استفاده شده است نیز ضعیف تر است. با افزودن سیاست سوییچ کردن به مدل مبنا، Base+switch، ابتدا به طور متوسط بیش از 10.0 درصد BLEU بهبود می یابد. این نشان می دهد که توانایی انتخاب محتوا با چند نمونه آموزشی آسان تر قابل یادگیری است. با این حال، جملات بسیار محدود و غیر وان را تشکیل می دهد. با تقویت مدل زبان از پیش آموزش دیده، مدل ما Base+switch+LM یک پیشرفت قابل توجه دیگر با میانگین بیش از 8.0 امتیاز BLEU به ارمغان می آورد.

جدول 1: نتایج BLEU-4 بر روی 3 دامنه

Domain	Humans				Books				Songs			
# of training instance	-	100	200	500	-	100	200	500	-	100	200	500
Template	16.3	-	-	-	25.6	-	-	-	30.1	-	-	-
Base + Original	2.2	3.7	4.9	5.1	5.8	6.1	7.4	6.7	9.2	10.7	11.1	11.3
Base	2.9	5.1	6.1	8.3	7.3	6.8	7.8	8.8	10.4	12.0	11.6	13.1
Base + Switch	15.6	17.8	21.3	26.2	24.7	26.9	30.5	33.2	29.7	30.6	32.5	34.9
Base + Switch + LM (Sctrach)	6.6	11.5	15.3	18.6	7.1	9.2	14.9	21.8	11.6	16.2	20.6	23.7
Base + Switch + LM (Ours)	25.7	29.5	36.1	41.7	34.3	36.2	37.9	40.3	36.1	37.2	39.4	42.2

KGPT: Knowledge-Grounded Pre-Training

همانطور که در بخش های قبل گفته شد، مدل KGPT در ۳ آزمایش با ناظر، بدون نمونه و با نمونه محدود انجام شد. در این بخش به بررسی نتایج یادگیری با نمونه محدود می پردازیم. هدف از این کار، مطالعه پتانسیل شبکه پیش آموخته پیشنهادی در کاهش حاشیه نویسی (annotation) در وظیفه تولید متن از روی داده است. مقایسه در دو بخش کلی انجام می گیرد: (۱) مقایسه با مدل های مبنایی که پیش آموخته نیستند (با هدف بررسی اینکه چگونه پیش آموزش می تواند به توانایی یادگیری با نمونه محدود مدل کمک کند) (۲) مقایسه با مدل های زبانی پیش آموخته (برای دیدن مزایای KGPT در مقابل مدل های زبانی پیش آموخته موجود) مجموعه داده های WebNLG و E2ENLG: در این دو دیتاست از ۰.۱٪، ۰.۵٪، ۱٪، ۵٪، ۱۰٪ از داده های آموزشی برای آموزش مدل و مشاهده عملکرد آن با معیار BLEU-۴ استفاده شد. در چالش WebNLG، وضعیت یادگیری با نمونه محدود در طول زمان آزمایش، بسیاری از entity ها دیده نمی شود. تحت آزمایش ۵٪، مدل های پایه که پیش آموخته نیستند فقط می توانند متن های نامفهوم تولید کنند در حالیکه مدل پیش آموخته KGPT می تواند بدلیل توانایی تعمیم دهی قوی، به امتیاز ۴۰ برسد.

جدول 2: نتایج یادگیری با نمونه محدود روی مجموعه آزمون WebNLG

Model	0.5%	1%	5%	10%
Seq2Seq	1.0	2.4	5.2	12.8
Seq2Seq+Delex	4.6	7.6	15.8	23.1
KGPT-Graph w/o Pre	0.6	2.1	5.9	14.4
KGPT-Seq w/o Pre	0.2	1.7	5.1	13.7
Template-GPT-2	8.5	12.1	35.3	41.6
KGPT-Graph w/ Pre	22.3	25.6	41.2	47.9
KGPT-Seq w/ Pre	21.1	24.7	40.2	46.51

در چالش E2ENLG، وظیفه در مقایسه با قبلی ساده تر همراه با موارد محدودتری است. در این آزمایش، با کاهش تعداد نمونه ها به ۰.۱٪، مدل های پایه نمی توانند عملکرد خوبی داشته باشند در حالیکه مدل پیش آموخته KGPT همچنان می تواند به امتیاز بالای ۴۰ برسد.

جدول 3: نتایج یادگیری با نمونه محدود روی مجموعه آزمون E2ENLG

Model	0.1%	0.5%	1%	5%
TGen	3.6	27.9	35.2	57.3
KGPT-Graph w/o Pre	2.5	26.8	34.1	57.8
KGPT-Seq w/o Pre	3.5	27.3	33.3	57.6
Template-GPT-2	22.5	47.8	53.3	59.9
KGPT-Graph w/ Pre	39.8	53.3	55.1	61.5
KGPT-Seq w/ Pre1	40.2	53.0	54.1	61.1

مجموعه داده: WikiBio: تنظیمات این آزمایش مشابه با ۲-Switch-GPT و Pivot است که از ۵۰، ۱۰۰، ۲۰۰ و ۵۰۰ نمونه از مجموعه آموزشی برای آموزش مدل استفاده می شود. مدل KGPT می تواند به بهترین امتیازات دست یابد و هر دو مدل Template-GPT-۲ و ۲-GPT را در اکثر موارد پشت سر بگذارد.

جدول 4: نتایج یادگیری با نمونه محدود روی مجموعه آزمون WIKIBIO

Model	50	100	200	500
Field-Infusing	1.3	2.6	3.1	8.2
KGPT-Graph w/o Pre	0.2	1.1	3.8	9.7
KGPT-Seq w/o Pre	0.6	1.7	3.0	8.9
Pivot	7.0	10.2	16.8	20.3
Switch-GPT-2	17.2	23.8	25.4	28.6
Template-GPT-2	19.6	25.2	28.8	30.8
KGPT-Graph w/ Pre	24.5	27.5	28.9	30.1
KGPT-Seq w/ Pre	24.2	27.6	29.1	30.0

مطالعه کمی: در این مطالعه، بیشتر بررسی می کنیم که KGPT تا چه میزان می تواند پیچیدگی نمونه را کاهش دهد. به طور مشخص، یک میزان امتیاز BLEU-۴ را تعیین کرده و تعداد نمونه های آموزشی را تغییر می دهیم تا ببینیم چه مقدار نمونه آموزشی برای حفظ عملکرد مورد نیاز است. در اینجا BLEU=۳۰ تعیین شده است. نتایج در جدول زیر نشان داده شده است.

جدول 5: تعداد نمونه آموزشی مورد نیاز برای رسیدن به امتیاز BLEU مشخص در مجموعه داده های مختلف

Model	WebNLG	E2ENLG	WikiBio
KGPT w/o Pre	~1000	~300	~8000
KGPT w/ Pre	~700	~20	~500
Ratio	14x	15x	16x

همچنین نسبت کمی نمونه ها برای توصیف مزایای پیش آموزش محاسبه شده است. به طور کلی، پیش آموزش می تواند پیچیدگی نمونه برای آموزش را ۱۵ برابر کاهش دهد.

Few-shot Natural Language Generation for Task-Oriented Dialog

جدول 6: مقایسه دیتاست های موجود برای NLG

statistics	E2E NLG	BAGEL	RNNLG	FEWSHOTWOZ
# Domains	1	1	4	7
Avg. # Intents	1	8	11.25	8.14
Avg. # Slots	8	10	21	16.15
Avg. # Delexicalised DAs in Training	109	23.9	794.5	50
Avg. # Delexicalised DAs in Testing	7	14.3	566.5	472.857
Overlap Percentage	100	99.6	94.00	8.82
Avg. # Training instances	42056	363	4625.5	50
Avg. # Testing instances	630	41	1792.5	472.86

ارزیابی خودکار:

به عنوان معیار های ارزیابی اتوماتیک از BLEU score و slot error rate (ERR) استفاده میشود. BLEU score نشان میدهد که متن تولید شده در مقایسه با یک خواننده انسانی چقدر طبیعی است.

ارزیابی انسانی:

برای ارزیابی انسانی برای هر dialog act سه نفر در باره اطلاع رسانی (Informativeness) و طبیعی بودن (Naturalness) قضاوت کنند. اطلاع رسانی مشخص میکند که گفته تولید شده دارای اطلاعات لازم هست یا نه. طبیعی بودن نیز نشان دهنده این است که گفته تولید شده چقدر شبیه گفته های انسانی است. برای هر کدام از این دو ویژگی از ۱ تا ۳ نمره میدهند.

به عنوان خطوط مبنا از سه مدل استفاده کردند: (۱) SC-LSTM (۲) GPT-2 (۳) HDSA

عملکرد روی مجموعه داده ها:

۱. FEWSHOTWOZ مدل SC-GPT از بقیه مدل های مبنا، هم از لحاظ BLEU score و هم از لحاظ

ERR عملکرد بهتری داشت و این نشان دهنده این است که مدل ارائه شده، دارای قدرت کنترل پذیری زیادی دارد

و اطلاعات را به خوبی میتواند از dialog act منتقل کند در حین تولید متن در حالی که روان بودن آن را حفظ کند.

جدول 7: مقایسه روش های موجود بر روی FEWSHOTWOZ

Model	Restaurant		Laptop		Hotel		TV		Attraction		Train		Taxi	
	BLEU	ERR	BLEU	ERR	BLEU	ERR	BLEU	ERR	BLEU	ERR	BLEU	ERR	BLEU	ERR
SC-LSTM	15.90	48.02	21.98	80.48	31.30	31.54	22.39	64.62	7.76	367.12	6.08	189.88	11.61	61.45
GPT-2	29.48	13.47	27.43	11.26	35.75	11.54	28.47	9.44	16.11	21.10	13.72	19.26	16.27	9.52
SC-GPT	38.08	3.89	32.73	3.39	38.25	2.75	32.95	3.38	20.69	12.72	17.21	7.74	19.70	3.57

جدول 8: مقایسه آماری روش های موجود توسط ارزیابی انسانی

Model	Informativeness	Naturalness
SC-LSTM	2.29	2.13
GPT-2	2.54	2.38
SC-GPT	2.64	2.47
Human	2.92	2.72

۲. MultiWOZ:

ارزیابی این قسمت در جداول زیر آورده شده است. برای این قسمت از Entity F1 برای ارزیابی دقت پوشش موجودیت استفاده شده است. باز هم در این قسمت، SC-GPT به بهترین عملکرد روی BLEU score رسیده است. علاوه بر آن، GPT-2 نیز به عملکرد مشابهی با SC-GPT رسیده است و این به دلیل آن است که MultiWOZ دارای ۵۷ هزار داده برچسب گذاری شده است که برای عملکرد خوب GPT-2 کافی است. علاوه بر آن، نتایج نشان می دهد که با داشتن مقدار کافی داده برچسب گذاری شده، مدل های زبانی شرطی به طرز چشمگیری بهتر از HDSA عمل میکنند که از یک ساختار درختی استفاده میکند.

جدول 9: عملکرد روی MultiWOZ

Model	Entity F1	BLEU
SC-LSTM	80.42	21.6
HDSA	87.30	26.48
GPT-2	87.70	30.71
SC-GPT	88.37	30.76

جدول 10: امتیاز BLEU مدل های مختلف بر روی MultiWOZ با مقدار داده های مختلف

Model	Data size						
	0.1%	0.5%	1%	5%	10%	20%	50%
SC-LSTM	9.05	15.15	15.38	18.26	18.97	19.99	21.07
HDSA	9.40	15.32	18.27	22.19	22.89	24.16	25.01
GPT-2	11.96	18.88	20.29	24.18	25.39	26.25	27.40
SC-GPT	12.70	19.65	20.67	24.45	25.67	26.37	27.89

جدول 11: ارزیابی انسانی MultiWOZ

Model	Informativeness	Naturalness
-------	-----------------	-------------

SC-LSTM	2.14	2.33
GPT-2	2.34	2.42
SC-GPT	2.71	2.69
Human	2.77	2.61

علاوه بر آزمایش های فوق، برای اندازه گیری اثر اندازه مجموعه داده بر روی آموزش، مدل را با درصد های مختلفی از مجموعه داده، آموزش دادند.

نمونه هایی از پاسخ های تولید شده را در زیر میتوانیم مشاهده کنیم:

#	Model	Generated Responses from Different Models
1	Input DA	Laptop{inform(name=satellite proteus 84; type=laptop; memory=8 gb; drive=1 tb; weight=2.3 kg)}
	Reference	the satellite proteus 84 is a laptop with a 1 tb drive , 8 gb memory and weighs in at 2.3 kg
	SC-LSTM	the satellite proteus 84 is a laptop with 8 gb , with a 1 tb drive , and is for business computing , and is for business computing % [businesscomputing=true] [weight=2.3kg]
	GPT-2	the satellite proteus 84 laptop has a 1 tb drive , weighs 2.3 kg and is used for business computing % [businesscomputing=true] [memory=8 gb]
	SC-GPT	the satellite proteus 84 is a laptop with 8 gb memory , 1 tb drive , and a weight of 2.3 kg
2	Input DA	Restaurant{inform_only_match(name=marnee thai and thanh long; pricerange=moderate; near=outer sunset; goodformeal=dinner)}
	Reference	marnee thai and thanh long are the only 2 moderately priced restaurants near outer sunset that are good for dinner
	SC-LSTM	marnee thai and thanh long is is moderate restaurant restaurant and good good good dinner % [near=outer sunset] [inform_only_match]
	GPT-2	there is a moderately priced restaurant called marnee thai and thanh long that is near the outer sunset area % [goodformeal=dinner] [inform_only_match]
	SC-GPT	marnee thai and thanh long is the only restaurant that serves moderately priced food near outer sunset and good for dinner
3	Input DA	Train{inform(time=50 minutes)}, hotel{request(stars=?; area=?), inform(choice=5)}
	Reference	travel time is 50 minutes . i found 5 hotels you may like . do you have a star rating request or an area you prefer
	SC-LSTM	there are 5 trains that are 50 minutes . do you have a preferred departure location ? % [train{inform(choice=5),request(departure=?)}] [hotel{request(stars=?; area=?), inform(choice=5)}]
	HDSA	there are 5 trains that meet your criteria . do you have a preference on the area or star rating ? % [train{inform(choice=5, time=50 minutes)}] [hotel{inform(choice=5)}]
	SC-GPT	there are 5 hotels that meet your criteria . the trip will last 50 minutes . do you have an area preference or star rating you would like ?

Table 8: Examples of generated utterances from different models, along with its corresponding dialog acts (DAs) and references. The first two examples are sampled from FEWSHOTWOZ and the last one is from MultiWOZ. Each generated utterance is followed by a brief description explaining the errors (starting with “%”). (Better viewed in color. wrong, redundant, missing information)

#	Model	Generated Responses from Different Models
	Original DA Reference	ack_makerreservation (price = moderate ; restaurantcuisine = chinese ; location = center ; peoplecount = 2) ok making a reservation for restaurant moderate chinese in center for two people?
1	Input DA	ack_makerreservation (price = moderate ; restaurantcuisine = japanese ; location = bellevue square ; peoplecount = 2)
	GPT-2	ok making reservation for moderate japanese restaurant in bellevue square for 2 people?
	SC-GPT	ok making a reservation for a moderate japanese cuisine restaurant in bellevue square for 2 people?
2	Input DA	ack_makerreservation (restaurantname = sakura ; price = moderate ; restaurantcuisine = japanese ; location = bellevue square ; peoplecount = 2)
	GPT-2	ok making a reservation for 2 people sakura moderate japanese restaurant located bellevue square for 10 minutes?
	SC-GPT	ok making a reservation for sakura, moderate japanese cuisine in bellevue square for 2 people?
3	Input DA	ack_makerreservation (price = expensive ; restaurantcuisine = japanese ; location = center ; peoplecount = 2)
	GPT-2	ok making reservation for japanese cuisine expensive?
	SC-GPT	ok making reservation for expensive japanese cuisine?
#	Model	Generated Responses from Different Models
	Original DA Reference	ack_makerreservation (childrenfriendly = true) make reservation for a children friendly restaurant ?
1	Input DA	ack_makerreservation (restaurantname = sakura ; childrenfriendly = true ; price = moderate)
	GPT-2	make reservation at sakura child friendly restaurant and make table moderate price?
	SC-GPT	make reservation for restaurant sakura moderate price and children friendly restaurant?

Table 10: Examples of generated utterances with novel dialog acts. SC-GPT produces better utterances than GPT-2 for with edited dialog acts. Since both models produce similar responses to references for the original dialog act, the results are not shown here. (Better viewed in color. **insert a slot**, **substitute a slot value**, **delete a slot**).

فصل 5:

جمع‌بندی و پیشنهادها

جمع‌بندی:

در این تحقیق سعی بر آن شد که روش‌های متفاوت تبدیل متن به داده در یادگیری با نمونه محدود مورد بررسی قرار گیرند. سایر روش‌ها نیز توانسته‌اند به نتایج قابل قبولی در این زمینه برسند اما نیازمند تعداد بسیار زیادی داده مختص به همان وظیفه هستند.

در مقاله نخست (Few-Shot NLG with Pre-Trained Language Model)، دو چیز بیان شد: (۱) گیت کپی برای تصمیم به تولید کلمه یا کپی از داده (۲) استفاده از مدل زبانی برای تولید جملات منسجم. پیاده‌سازی این رویکرد ضمن دستیابی به عملکرد قوی در حوزه‌های مختلف ساده است. ایده اولیه دستیابی به مدل‌سازی زبان از قبل را میتوان به طور بالقوه به دامنه وسیع‌تری از وظایف بر اساس ساختار داده ورودی (به طور مثال گراف دانش، کوئری SQL و ...) گسترش داد.

در مقاله دوم (KGPT: Knowledge-Grounded Pre-Training for Data-to-Text Generation)، از یک دستورالعمل پیش‌آموزش برای داده‌های بدون برچسب خارجی برای وظیفه تولید داده به متن بهره‌برداری می‌کند. همچنین یک مدل تولیدکننده مبتنی بر دانش ارائه می‌شود. این مدل توانسته به نتایج خوبی در یادگیری بدون نمونه و یادگیری با نمونه محدود دست یابد.

در مقاله سوم (Few-shot Natural Language Generation for Task-Oriented Dialog)، دو چیز برای دیالوگ‌های مبتنی بر وظیفه ارائه شد: (۱) مجموعه داده‌ای به نام FEWSHOTWOZ برای یادگیری با نمونه‌های محدود در سناریوهای دنیای واقعی. (۲) یک مدل جدید به نام SC-GPT با قابلیت کنترل معنایی قوی و تعمیم‌پذیری زیاد. روی هر دو مجموعه داده FEWSHOTWOZ و MultiWOZ بهترین نتایج را به دست آورد.

پیشنهادهای:

دو ایده برای کارهای آینده پیشنهاد می‌کنیم: (۱) طراحی مکانیسم‌هایی برای تولید پاسخ‌های بین‌فردی بیشتر که ثابت شده است به بهبود تجربیات کاربر کمک می‌کند. (۲) تعمیم ایده تولیدکننده پیش‌آمورخته به هر چهار ماژول در سیستم گفتگو برای آموزش انتها به انتها. از آنجایی که این چهار ماژول اطلاعات را به ترتیب پردازش می‌کنند، میتوان ورودی/خروجی آنها را به عنوان بخش، سازماندهی کرده و مدلی را در سطح بخش پیش‌آموزش داد.

مراجع

مراجع

- Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever .2019 .
 “Language models are unsupervised multitask learners”.
- Ankur P Parikh, Xuezhi Wang, Sebastian Gehrmann, Manaal Faruqui, Bhuvan Dhingra, Diyi Yang, Dipanjan Das“ .بدون تاریخ. A controlled table-to-text generation dataset”.
- Chen Wenhui, Yu Su, Xifeng Yan, William Yang Wang“ .2020 .KGPT: Knowledge-Grounded Pre-Training for Data-to-Text Generation ”<https://arxiv.org/abs/2010.02307>.
- Chen Zhiyu, Harini Eavani, Wenhui Chen, Yinyin Liu, William Yang Wang“ .2019 .Few-Shot NLG with Pre-Trained Language Model”.
- Claire Gardent, Anastasia Shimorina, Shashi Narayan, Laura Perez-Beltrachini“ .2017 .The webnlg challenge: Generating text from RDF data ”*INLG* .
- Daniel Adiwardana, Minh-Thang Luong, David R So, Jamie Hall, Noah Fiedel, Romal Thoppilan, Zi Yang, Apoorv Kulshreshtha, Gaurav Nemade, Yifeng Lu, et al .2020 .
 “Towards a human-like open-domain chatbot ”*arXiv* .
- Ehud Reiter, Robert Dale“ .1997 .Building applied natural language generation systems”.
- Gatt Albert, Emiel Krahmer“ .2018 .Survey of the State of the Art in Natural Language Generation: Core tasks, applications and evaluation”.
- Hady ElSahar, Christophe Gravier, and Fred ´ erique ´ Laforest“ .2018 Zero-shot question generation from knowledge graphs for unseen predicates and entity types ”*NAACL* .
- Hiroaki Hayashi, Zecong Hu, Chenyan Xiong, Graham Neubig“ .2020 .Latent relation language models ”*AAAI* .
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova“ .2018 .Bert: Pre-training of deep bidirectional transformers for language understanding ”*arXiv* .
- Jeffrey Pennington, Richard Socher, and Christopher D.Manning“ .2014 .Glove: Global vectors for word representation ”*EMNLP* .
- Kukich, Karen “ .1983 .Design of a knowledge-based re-port generator ”*Association for Computational Linguistics*.145-150
- Lebret Remi, David Grangier, Michael Auli “ .2016 .Neural Text Generation from Structured Data with Application to the Biography Domain”.
- Li Zhou, Jianfeng Gao, Di Li, Heung-Yeung Shum“ .بدون تاریخ. The design and implementation of xiaoice, an empathetic social chatbot ”*Computational Linguistics* .

- Liu Tianyu, Kexiang Wang, Lei Sha, Baobao Chang, Zhifang Sui“ .2017 .Table-to-text Generation by Structure-aware Seq2seq Learning”.
- Marilyn A. Walker, Owen Rambow, Monica Rogati“ .2001 .Spot: A trainable sentence planner ”*NAACL* .
- Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, Luke Zettlemoyer“ .2018 .Deep contextualized word representations ”*NAACL* .
- Nitish Shirish Keskar, Bryan McCann, Lav R Varshney, Caiming Xiong, Richard Socher .
“ .2019Ctrl: A conditional transformer language model for controllable generation ”.
arXiv .
- Ondřej Dusek, Filip Jurc'ıcek“ .2016 .Sequence-to-sequence generation for spoken dialogue via deep syntax trees and strings ”*ACL* .
- Ondřej Dusek, Jekaterina Novikova, Verena Rieser“ .2019 .Evaluating the state-of-the-art of end-to-end natural language generation ”*The E2E NLG Challenge* .
- Peng Baolin, Chenguang Zhu, Chunyuan Li, Xiujun Li, Jinchao Li, Michael Zeng, Jianfeng Gao“ .2020 .Few-shot Natural Language Generation for Task-Oriented Dialog”.
- Percy Liang, Michael I. Jordan, Dan Klein“ .2009 .Learning semantic correspondences with less supervision ”*ACL* .
- Quoc V. Le ,Tomas Mikolov“ .2014 .Distributed representations of sentences and documents ”.
ICML .
- Ratish Puduppully, Li Dong, Mirella Lapata“ بدون تاریخ. Data-to-text generation with content selection and planning”.
- Remi Lebret, David Grangier ,Michael Auli“ .2016 .Neural text generation from structured data with application to the biography domain ”*EMNLP* .
- Robert Logan, Nelson F Liu, Matthew E Peters, Matt Gardner, Sameer Singh“ .2019 .Barack's wife hillary: Using knowledge graphs for fact-aware language modeling ”*ACL* .
- Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, Yejin Choi“ .2019 .Defending against neural fake news”.
- Ryan Kiros, Yukun Zhu, Ruslan Salakhutdinov, Richard S. Zemel, Raquel Urtasun, Antonio Torralba, Sanja Fidler“ .2015 .Skip-thought vectors”.
- Sam Wiseman, Stuart M. Shieber, and Alexander M.Rush“ .2018 .Learning neural templates for text generation ”*EMNLP* .
- Sam Wiseman, Stuart Shieber, Alexander Rush“ .2017 .Challenges in data-to-document generation ”*EMNLP* .

- See Abigail, Peter J Liu, and Christopher D. Manning“ .2017 .Get To The Point: Summarization with Pointer-Generator Networks”.
- Shuming Ma, Pengcheng Yang, Tianyu Liu, Peng Li, Jie Zhou, Xu Sun“ .2019 .Key fact as pivot: A two-stage model for low resource table-to-text generation ”*ACL* .
- Sungjin Ahn, Heeyoul Choi, Tanel Parnamaa, Yoshua Bengio“ .2016 .A neural knowledge language model ”*arXiv* .
- Tianyu Liu, Kexiang Wang, Lei Sha, Baobao Chang, Zhifang Sui“ .2018 .able-to-text generation by structure-aware seq2seq learning ”*AAAI* .
- Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S.Corrado, and Jeffrey Dean .2013 . “Distributed representations of words and phrases and their compositionality”.
- Tsung-Hsien Wen, Milica Gasic, Dongho Kim, Nikola Mrksic, Pei-Hao Su, David Vandyke, Steve Young“ .2015 .Stochastic language generation in dialogue using recurrent neural networks with convolutional sentence reranking ”*Association for Computational Linguistics* .
- Tsung-Hsien Wen, Milica Gasic, Nikola Mrksic, Pei-Hao Su, David Vandyke, Steve Young“ .2015 .Semantically conditioned LSTM-based natural language generation for spoken dialogue systems ”*ACL* .
- Wei Lu, Hwee Tou Ng, Wee Sun Lee“ .2009 .Natural language generation with tree conditional random fields ”*EMNLP* .
- Wen Tsung-Hsien, Milica Gasic, Nikola Mrksic, Pei-Hao Su, David Vandyke, Steve Young . “ .2015Semantically Conditioned LSTM-based Natural Language Generation for Spoken Dialogue Systems”.
- Wenhu Chen, Jianshu Chen, Yu Su, Zhiyu Chen, William Yang Wang“ .2020 .Logical natural language generation from open-domain tables ”*ACL* .
- Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, Bill Dolan“ .2019 .Dialogpt: Large-scale generative pre-training for conversational response generation ”*arXiv* .
- Zhiyu Chen, Harini Eavani, Wenhu Chen, Yinyin Liu, William Yang Wang“ .2020 .Few-shot nlg with pre-trained language model ”*ACL* .