Service registration chatbot: collecting and comparing dialogues from AMT workers and service's users

Luca Molteni **Mittul Singh** **Juho Leinonen** **Katri Leino** **Mikko Kurimo**
Department of Signal Processing and Acoustics, Aalto University, Finland
firstname.lastname@aalto.fi
Emanuele Della Valle
Department of Electronics, Information and Bioengineering, Politecnico of Milano, Italy
emanuele.dellavalle@polimi.it

Abstract

Crowdsourcing is the go-to solution for data collection and annotation in the context of NLP tasks. Nevertheless, crowdsourced data is noisy by nature; the source is often unknown and additional validation work is performed to guarantee the dataset's quality. In this article, we compare two crowdsourcing sources on a dialogue paraphrasing task revolving around a chatbot service. We observe that workers hired on crowdsourcing platforms produce lexically poorer and less diverse rewrites than service users engaged voluntarily. Notably enough, on dialogue clarity and optimality, the two paraphrase sources' human-perceived quality does not differ significantly. Furthermore, for the chatbot service, the combined crowdsourced data is enough to train a transformer-based Natural Language Generation (NLG) system. To enable similar services, we also release tools for collecting data and training the dialogue-act-based transformer-based NLG module[1].

1 Introduction

Task-specific neural dialogue models demand high-quality annotated dialogue data. Unfortunately, gathering human-generated and annotated dialogues is a costly and time-consuming task. Easily accessible sources, like social-network feeds and online forums, are cursed by systematic problems such as extra-linguistic annotations, irregular turn-taking, and the lack of a standard format leading to an intense pre-processing phase. Even so, models trained with this type of data might not work well in a more natural domain (Leino et al., 2020). In recent times, thanks to online platforms like Amazon Mechanical Turk (AMT) [2], crowdsourcing has become the most popular solution to tackle the problem of manually generating and annotating written dialogues.

However, as a small business, minimizing such added costs while automating user-based workflows is essential. In this work, we consider leveraging voluntary submissions by business users for creating a chatbot.

For the chatbot service, we consider a new class of broadly diffused tasks that we name Service Registration Tasks (SRTs), which involves the domain-agnostic act of registering to an online service. As a use case, we work with SiirtoSoitto[3] to provide users with a chatbot for service registration. SiirtoSoitto is a free online service offered to the city of Helsinki that notifies users about scheduled roadworks and imminent car towings. We employ a dialogue templating method called Machine Talking to Machines (M2M) (Shah et al., 2018b,a). It simulates the interaction between a user and system to automatically generate templates, which are then paraphrased by AMT and service users.

In this work, we make the following contributions. **1)** We release the data collection tools to the public, including an integration with popular instant messaging platforms to engage with service's users (Section 4). **2)** We analyze and compare the data collected via AMT workers and service's users in an empirical and human evaluation (Section 5). **3)** We show the usefulness of collected data by training a dialogue act induced transformer-based language generation module (Section 6). We also release the module's code publicly.

2 Service Registration Task (SRT)

Here, we focus on a class of tasks named Service Registration Tasks that consists of registering to a general online service. This human-machine interaction is characterized by the collection and val-

[1] https://github.com/Molteh/M2M
[2] https://www.mturk.com/
[3] https://www.siirtosoitto.com

Proceedings of the 2020 EMNLP Workshop W-NUT: The Sixth Workshop on Noisy User-generated Text, pages 116–121

idation of information and preferences from the user. As a specific instance of this class of tasks, we picked the use case of SiirtoSoitto, an online service that warns and notifies vehicle owners in the city of Helsinki about road maintenance and imminent towings.

3 Machines Talking to Machines (M2M)

For chatbot development, we employ the Machine Talking to Machines (M2M) framework (Shah et al., 2018b,a) to setup the annotated data collection. Conceived as being domain-independent, M2M generates dialogues centered on completing a specific task.

The M2M consists of four major steps. **1)**, the developer provides the task-specific knowledge used by the system. It can be seen as a collection of all the units of information exchanged during the dialogue. **2)** Given a task specification, a simulated interaction of a user and the system generates sequences of dialogue acts exhaustively. The output sequences enclose the semantic content of the dialogue. The user is modeled as an agenda-based user simulator (Schatzmann et al., 2007) while the system is designed as a *Mealy* machine. This process is also called *self-play*, where a simulated user interacts with the system. A generated example is shown in the first row of Table 1. **3)** Using the semantic parses, we can then build dialogue templates using a simple domain grammar. The templates are slightly unnatural computer-generated dialogue utterances paired with their semantic representation in the form of dialogue acts (second row of Table 1). **4)** Finally, the dialogue templates enter a paraphrasing phase where crowdsource workers provide natural and contextual rewrites of the machine-generated sentences (last row of Table 1).

4 Applying M2M to SRT

Our SRT is characterized by exchanging information such as telephone numbers, license plates, areas of interest, and the acceptance of terms and conditions. These characteristics form the task-specification used to initialize the M2M's first step. A dialogue scenario is sampled by assigning a valid or invalid value to each entity.

Through self-play, we can generate sequences of dialogue acts until the goal of registering is reached or some invalid state is encountered (e.g., the user provides invalid values). Next, we build a simple rule-based domain grammar that converts the annotated sequences into templates, first turning them into syntactic skeletons with proper punctuation and conjunctions, and then substituting the entity values with custom terms to increase readability.

Self-play annotations	request(license_plate), request (phone_number)
Template utterance	provide reference for: License plate and Phone number
Paraphrase	please list your license plates and your phone number

Table 1: A single-turn sample showcasing the M2M generation process.

In the next step, the same dialogues are used to set up a paraphrasing task on AMT and on the rule-based chatbot that makes SiirtoSoitto available to the public. Chatbot users are asked to participate voluntarily in an experimental task. They are presented with dialogue turns to rewrite sequentially on their preferred instant messaging application. A quick manual quality check removed roughly 25% of all AMT feedback due to a lack of compliance with the instructions. In contrast only 10% of SiirtoSoitto users failed to understand their task and produced unusable data.

In the above process, instead of annotating natural utterances, we are building dialogues upon annotations. The automatic generation of the outlines guarantees greater diversity and explores all the relevant paths conceived by the task designer. Finally, employing human writers ensures the *naturalness* of the utterances, and the variety is boosted by asking them to rephrase highly generic machine-generated sentences. This reverse processing guarantees the quality of the semantic annotations.

In Table 4, we present the statistics of the data collected by employing AMT and service users (SiirtoSoitto). In each case, we ran the paraphrasing step over multiple sessions across five days. We presented the same dialogue set to both the groups to improve the comparability among generated paraphrases. Then, we performed a human evaluation to validate paraphrase quality and removed any spurious paraphrases. We were able to collect 98 and 83 dialogues via AMT and SiirtoSoitto users, respectively. With a larger number of dialogues and turns, AMT workers produced more data than SiirtoSoitto users.

In terms of effort, we set up the paraphrasing task on AMT and the chatbot service in similar

Metric	AMT	SiirtoSoitto
Dialogues	98	83
#Turns	898	718
#Tokens	7723	5069
Lexical richness (#Unique *n-grams*/ #Tokens)		
Unigrams	0.104	**0.161**
Bigrams	0.103	**0.122**
Trigrams	0.28	**0.387**
Diversity		
Tdiv	155	**270**
Jaccard distance	0.432	**0.490**

Table 2: Summary of the quantitative evaluation.

amounts of time. For the chatbot service, we introduced some additional conversational interaction and integrated the M2M-generated templates into the service. For the AMT setup, we had to design and implement the paraphrasing task in AMT task's single HTML page and import batches of dialogue templates by hand. From a monetary standpoint, as we recruited users voluntarily, paraphrasing with chatbot users did not lead to any costs. On AMT, we spent a total of 63$ which includes the cost for each single task (0.5$) and the platform fees.

5 Evaluation

In this section, we compare the data collected via the two different crowdsourcing sources. We compare them quantitatively based on the lexical richness and language diversity. We also ask human evaluators to grade dialogues qualitatively.

5.1 Lexical Richness and Diversity

Lexical rich and diverse paraphrases can allow the chatbot to feel more real and natural. In effect, it helps the users to have a more satisfying experience even in a simple task. Hence, having lexical rich and diverse data is desirable.

Lexical richness is calculated as the ratio between unique *n-grams* and total tokens per collection source (Hout and Vermeer, 2007). Interestingly, even with a lower dialogue count, the SiirtoSoitto dataset presents a higher lexical richness than the AMT dataset. This effect indicates greater language variety associated with expert user rewrites. Moreover, higher bigram and trigram lexical richness for SiirtoSoitto dataset than AMT datasets highlights a greater construct variety in SiirtoSoitto dataset. Table 3 displays an example this effect

Dialogue template	provide reference for: Phone number
AMT rewrite	please provide phone number.
SiirtoSoitto rewrite	can you still give me your phone number please?

Table 3: Example of rewrite collected from AMT and SiirtoSoitto chatbot service users.

where the SiirtoSoitto users rewrite with more constructs than AMT workers.

Diversity is measured by using two metrics: Term Frequency - Inverse Document Frequency (TF-IDF) diversity metric (Tdiv) (Liu et al., 2019) and Jaccard distance.

Tdiv is the sum of TF-IDF scores over *n-grams* ($n \leq 3$) in a document (D), as defined below. TF-IDF reflects the importance of an *n-gram*. *n-grams* with lower frequency in the collected data have higher IDFs. Thus, the Tdiv metric denotes the extent of diversity of an expression in the dataset.

$$Tdiv(R) = \sum_{n=1}^{N} \frac{\sum_{n\text{-}gram \in R} TF\text{-}IDF(n\text{-}gram)}{V_n}$$

$$V_n = \frac{1}{|D|} \sum_{R \in D} \sum_{n\text{-}gram \in R} TF\text{-}IDF(n\text{-}gram)$$

The Tdiv score for a sentence has little meaning, as it needs to be compared with Tdiv scores of sentences that entail the same semantic content. Given two rewrites for the same turn, one from the AMT dataset and one from SiirtoSoitto, if the latter has a higher Tdiv score, it is considered having more vibrant expressions than the former. For an overall comparison, we keep track these *wins* for each type of dataset per turn. We observe that SiirtoSoitto wins almost two out of three times, thus having paraphrases with richer expressions.

The Jaccard distance is a metric based on the Jaccard similarity coefficient that measures the dissimilarity between two finite sets of elements, in this case, the words that make up a sentence. This coefficient has been used as a proxy of the effort put in by the crowdsource to write paraphrases with different wordings from the proposed templates. In terms of average Jaccard distance, SiirtoSoitto (0.490) users outperform Amazon Turkers (0.432). This effect is exemplified by the example shown in

	AMT	SiirtoSoitto
Naturalness	4.05 (0.74)	4.15 (0.65)
Clearness	4.30 (0.71)	4.05 (0.80)
Grammaticality	3.85 (0.65)	4.20 (0.67)
Optimality	4.05 (0.49)	4.00 (0.63)

Table 4: Results of human evaluation on the collected dialogues. Numbers shows average scores of per dialogue grading. Standard deviation in brackets.

Table 3, where SiirtoSoitto users use more words than AMT workers.

5.2 Qualitative evaluation

Human evaluators assessed the perceived quality of the generated and paraphrased dialogues. Each dialogue was judged for four qualities: naturalness, clearness, grammaticality, and optimality. Naturalness indicates how well the sentences resemble typical human expressions. Clearness refers to the extent to which the meaning conveyed by the dialogue turns is easily understandable. Grammaticality reflects the absence of misspellings or badly formatted sentences. Finally, optimality refers to how quickly the proposed rewrites seem to go straight to the point. The scores were provided on a scale of one to five, with one representing the lowest quality and five being the highest. Table 4 details the score average across the twenty evaluators. Both AMT- and SiirtoSoitto-based datasets were judged to be similar from a human standpoint, as their differences were not significant. Also, both datasets scored highly on the four dimensions attesting the quality of the data collected.

6 Transformer-based language generator

We train a neural model for Natural Language Generation (NLG) to observe the effectiveness of the collected data. The neural model is a Transformer network (Vaswani et al., 2017) that converts the next dialogue acts into an output sentence. For the NLG use case, our Transformer architecture includes two separate encoders. The first encoder inputs a sequence of dialogue acts capturing the semantic meaning of the sentence that needs to be generated. The second encoder inputs the user's turn. As a single person writes each paraphrase of an entire dialogue, the person's style is reflected in both user and system turns. Intuitively, the second encoder employs the user's style to adapt the generated utterance to the user's persona. Our transformer implementation is trained with the Noam optimizer on negative log-likelihood loss (Vaswani et al., 2017). Encoders and decoder are characterized by three identical replicated blocks, 16 attention heads and a dropout rate of 0.1. Both the first encoder and decoder have 1024 hidden nodes while the second encoder uses 256 hidden nodes. We release our dialogue-act based transformer implementation with this work[4].

```
> hi ! EOS
> act greeting slot none value none  act propose
slot registration value none EOS
< hello ! would you like to register to this
service ? EOS

> hi EOS
> act greeting slot none value none act propose
slot registration value none EOS
< hello would you like to register to SiirtoSoitto
EOS

> my area is Helsinki Central EOS
> act request slot terms and conditions value
none EOS
< please accept the terms and conditions EOS
```

Figure 1: Examples of some test set sentences generated with the NLG module.

Figure 1 showcases some of the sentences generated with the NLG module. It also includes an instance in which the same sequence of input dialogue acts results in different system output sentences given the different user's utterances.

7 Related work

In our work, we applied M2M via two types of crowdsourcing methods. Earlier work (Kittur et al., 2008) has shown that AMT workers achieve significantly lower performances when the degree of experience and contextual knowledge is important. However, their performance improves with a more guided task structure. In our experiment, the service's users already had the background knowledge necessary for the task. Moreover, considering the generated dialogue's lexical richness and diversity, their paraphrases were ranked higher than AMT workers. However, at a qualitative level, both types of paraphrase ranked similarly.

Prior work (Walker et al., 2018; Budzianowski et al., 2018) has been concerned about the *unnatural* process of dialogue generation in the M2M approach. In our perspective, this issue affects scenarios where a simulated user cannot model the ambiguities of a real user, but for a simplistic SRT

[4] https://github.com/Molteh/M2M

use case, we disregard this issue.

For creating the NLG module, we focus on the generation of surface expression based on sequences of dialogue acts. Similarly, quite a few prior work (Stent, 2001; Wen et al., 2015; Liu and Liu, 2019; Varshney et al., 2020; Chen et al., 2019; Nayak et al., 2017) have employed semantic structures to generate dialogue utterances. Stent (2001) leveraged custom dialogue acts to implement a rule-based utterance generator as part of a bigger modular conversational system. Recently, LSTM-based machine translation models (Wen et al., 2015; Nayak et al., 2017) and Transformers (Liu and Liu, 2019; Varshney et al., 2020; Chen et al., 2019) have also been successfully explored in NLG tasks for open-domain and task-specific dialogue systems. For both open-domain and task-specific modules, large corpora of annotations are required for training the modules. In contrast, our work considers a simple SRT where even small amounts of crowdsourced data can help build good models. Additionally, unlike most of the prior work, we release our NLG module code to the public.

8 Conclusions

Collecting annotated datasets for NLG is a challenging task which sees crowdsourcing as the preferred solution to balance costs and time. In this work, we considered voluntarily engaging SiirtoSoitto's users to contribute towards a paraphrasing task for building a chatbot. Our findings suggest that engaging SiirtoSoitto users might produce more diverse and lexically rich results than engaging AMT workers empirically whereas, from a qualitative standpoint, both the datasets are similar for a simple service registration task. We can obtain similar amounts of data while running the data collection effort employing both sets of users for a comparable time. More importantly, through this process, we were able to reduce our costs of collecting data.

Additionally, in simple use cases like the SRT, this data are enough to build a transformer-based NLG module conditioned on dialogue acts. To support other small businesses, we make our data collection pipeline and code to train the transformer-based NLG module public.

Acknowledgments

We thank Twenty Hexagons Oy, the company behind SiirtoSoitto service, which provided us the opportunity to work with their infrastructure and engage with their user base. We also thank anonymous reviewers for their helpful comments.

References

Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. 2018. MultiWOZ - a large-scale multi-domain wizard-of-Oz dataset for task-oriented dialogue modelling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5016–5026, Brussels, Belgium. Association for Computational Linguistics.

Wenhu Chen, Jianshu Chen, Pengda Qin, Xifeng Yan, and William Yang Wang. 2019. Semantically conditioned dialog response generation via hierarchical disentangled self-attention.

Roeland Hout and Anne Vermeer. 2007. Comparing measures of lexical richness. *In: H. Daller, J. Milton J. Treffers-Daller (eds.), Modelling and assessing vocabulary knowledge (93-116). Cambridge: Cambridge University Press.*

Aniket Kittur, Ed H. Chi, and Bongwon Suh. 2008. Crowdsourcing user studies with mechanical turk. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, page 453–456, New York, NY, USA. Association for Computing Machinery.

Katri Leino, Juho Leinonen, Mittul Singh, Sami Virpioja, and Mikko Kurimo. 2020. Finchat: Corpus and evaluation setup for finnish chat conversations on everyday topics. *arXiv preprint arXiv:2008.08315*.

D. Liu and G. Liu. 2019. A transformer-based variational autoencoder for sentence generation. In *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7.

L. Liu, J. Tang, X. Wan, and Z. Guo. 2019. Generating diverse and descriptive image captions using visual paraphrases. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4239–4248.

Neha Nayak, Dilek Hakkani-Tur, Marilyn Walker, and Larry Heck. 2017. To plan or not to plan? discourse planning in slot-value informed sequence to sequence models for language generation. pages 3339–3343.

Jost Schatzmann, Blaise Thomson, Karl Weilhammer, Hui Ye, and Steve Young. 2007. Agenda-based user simulation for bootstrapping a POMDP dialogue system. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Companion Volume, Short Papers*, pages 149–152, Rochester, New York. Association for Computational Linguistics.

Pararth Shah, Dilek Hakkani-Tür, Bing Liu, and Gokhan Tür. 2018a. Bootstrapping a neural conversational agent with dialogue self-play, crowdsourcing and on-line reinforcement learning. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 3 (Industry Papers)*, pages 41–51, New Orleans - Louisiana. Association for Computational Linguistics.

Pararth Shah, Dilek Hakkani-Tür, Gökhan Tür, Abhinav Rastogi, Ankur Bapna, Neha Nayak, and Larry P. Heck. 2018b. Building a conversational agent overnight with dialogue self-play. *CoRR*, abs/1801.04871.

Amanda Stent. 2001. *Dialogue Systems as Conversational Partners: Applying conversation acts theory to natural language generation for task-oriented mixed-initiative spoken dialogue*. Ph.D. thesis.

Deeksha Varshney, Asif Ekbal, Ganesh Prasad Nagaraja, Mrigank Tiwari, Abhijith Athreya Mysore Gopinath, and Pushpak Bhattacharyya. 2020. Natural language generation using transformer network in an open-domain setting. In *Natural Language Processing and Information Systems*, pages 82–93, Cham. Springer International Publishing.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

Marilyn Walker, Albry Smither, Shereen Oraby, Vrindavan Harrison, and Hadar Shemtov. 2018. Exploring conversational language generation for rich content about hotels. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*, Miyazaki, Japan. European Languages Resources Association (ELRA).

Tsung-Hsien Wen, Milica Gašić, Nikola Mrkšić, Pei-Hao Su, David Vandyke, and Steve Young. 2015. Semantically conditioned LSTM-based natural language generation for spoken dialogue systems. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1711–1721, Lisbon, Portugal. Association for Computational Linguistics.