Goal-Oriented Dialogue with Sparse Language Models

Rita Fernandes Leite dos Santos Costa
ritaflscosta@tecnico.ulisboa.pt
Instituto Superior Técnico, Universidade de Lisboa, Portugal
January, 2021

Abstract

The purpose of goal-oriented dialogue systems is to provide automatic responses in a conversation with a specific goal. Given recent advances in Deep Learning, this task is now more flexible, as pre-existing knowledge from models trained with self-supervised learning can be transferred to conversation systems. However, the need to adapt the original answer to the dialogue context makes the task of generating it particularly challenging. Different strategies to decode a sentence have been proposed, aiming at making the generated text more fluent, coherent, and relevant. The goal of this study is to experiment sparse generation techniques in this framework, which sample from the recently proposed α-entmax transformation. We compare this technique with other state-of-the-art approaches, such as greedy search and nucleus sampling, by thoroughly assessing the different systems. Moreover, as the modularized approach is replaced by end-to-end architectures, goal-oriented systems become more difficult to be evaluated. Many works resort to evaluation methods imported from other tasks, namely machine translation, raising the question of whether they are suitable for evaluating dialogue. To address this problem, we conduct a study to determine the correlation between these automatic metrics and human perception of quality. The evaluation procedure is an important part of the performance analysis, since choosing an inappropriate method can lead to the wrong conclusions.

1. Introduction

The evolution of technology has allowed for the automation of several processes across diversified engineering industry fields. Namely, customer support services have drastically evolved with recent advances in Machine Learning. One of the biggest goals of Natural Language Processing is to develop a conversational system able to interact with humans in goal-oriented dialogue tasks. The study of these systems can be particularly relevant to Aerospace Engineering in two main frontiers. The most self-evident is the direct application in the aviation business, where conversational Artificial Intelligence leads to an improvement of customer support platforms, consequently enhancing the client experience.[1] Besides, recent studies support the introduction of speech dialogue systems in manufacturing processes. An efficient human-robot communication can reform the aerospace industry, not only enhancing their alone performance through cooperation between the two parts, but also supporting decision making with information-rich systems (Gaizauskas et al., 2018).

Advances in Deep Learning and language modelling allow for the development of neural approaches in dialogue generation, typically relying on Sequence-to-Sequence architectures (Sutskever et al., 2014; Wen et al., 2015). The traditional highly-handcrafted modular approaches are replaced by the joint optimization of multiple components, originating less complex systems with a stiffer architecture, learnable in an end-to-end manner (Wen et al., 2017).

Research on answer generation can be categorized in two different approaches: retrieval and generative based methods. The latter are more adaptable to different contexts, but more complex to design (Celikyilmaz et al., 2020). Recent advances in large-scaled pre-trained Language Models have paved the way for higher quality generation: Wolf et al. (2019) and Golovanov et al. (2019) have shown their applicability to open-domain conversational systems, with Budzianowski and Vulić (2019) introducing them to the goal-oriented framework.

Nonetheless, mimicking the human way of constructing a sentence is a challenging task and the chosen method can have great influence on the final result, motivating decoding techniques such as top-k sampling (Fan et al., 2018) and nucleus sampling (Holtzman et al., 2020). This work adds to the study of end-to-end generative conversation systems, by leveraging α-entmax transformations to decoding a sentence in a Language Model (Peters et al., 2019; Martins et al., 2020), in the goal-oriented setting. We compare this approach with state-of-the-art generation techniques, using both automatic and human metrics.

[1] As an example, we have the work done by the company Unbabel: https://unbabel.com/customer-service/travel/, last accessed on 14-12-2020.

Furthermore, as important as developing a system that resembles a human is the ability to correctly evaluate its performance. End-to-end goal-oriented dialogue generation demands a change in this paradigm, as traditional rule-based methods are less appropriate (Deriu et al., 2020). Many works have started reporting metrics created for different purposes, namely machine translation. Along with the increased usage of these metrics comes the inquiry of whether they are accurate indicators of quality in dialogue systems. To answer this question, we will exploit the most reliable tool to judge if a system is faithfully imitating the behaviour of a human — human judgement itself.

2. Background

2.1. Language Model

Defining a sentence as a sequence of words $w = (w_1, ..., w_T)$, a Language Model (Bengio et al., 2003) is able to look at a part of it and predict the next word, calculating the probability:

$$p_\theta(w) = \prod_{t=1}^{T} p_\theta(w_t | w_1, ..., w_{t-1}). \tag{1}$$

In a set S of training sentences, the strategy to learn the language modelling parameters θ is to minimize the cross-entropy, or the negative log-likelihood loss:

$$\mathcal{L}(\theta) = \sum_{i=1}^{|S|} \sum_{t=1}^{T} \log p_\theta(w_t | w_{<t}), \tag{2}$$

Language Models evolved to calculating p_θ using also some context information c, and more recently attention mechanisms were introduced, allowing to focus on specific parts of the input when decoding (Bahdanau et al., 2015).

2.2. GPT-2

GPT-2 is a Language Model trained in a self-supervised manner, on a zero-shot setting (Radford et al., 2019). The architecture is composed of independent heads of stacked decoder blocks, from the Transformer (Vaswani et al., 2017). Each block is composed of a Masked Self-Attention sublayer, meaning that each token only attends to its left context, and a Feed-Forward Neural Network sublayer. Each block has its own weights, and each head has a different pattern to attend to specific words. Similarly to the transformer, the output vector of the decoder is fed into a linear layer, projecting the output into a score vector over the vocabulary. The output of this Neural Network is a softmax layer, which turns these scores into probabilities, all positive and added up to 1. The next word is then chosen depending on the decoding strategy. Once a token is produced, it is added to the sequence of inputs, belonging to the input sequence in the next step — auto-regression.

2.3. α-entmax Transformation

Softmax is a dense distribution, meaning that a mass probability is always assigned to all the words. When sampling directly from it, the system can generate unnatural text, due to the unreliability of the tail of this distribution. Some techniques arose from this problem, such as nucleus and top-k sampling. However, they are only applied at decoding time, while at training time they are still optimized with the original softmax, creating a mismatch between training and testing (Martins et al., 2020). In this work, experiments will be performed sampling from the α-entmax transformation (Peters et al., 2019), which automatically produces sparse probability distributions, avoiding that mismatch.[2] The α-entmax transformation is defined as:

$$\alpha\text{-entmax}(z_t) := \underset{p \in \Delta^d}{\operatorname{argmax}}\, p^T z_t + H_\alpha(p), \tag{3}$$

where z_t are the scores produced by the model, $\Delta^d = \{p \in \mathbb{R}^d | \sum_{i=1}^{d} p_i = 1, p \geq 0\}$ is the probability simplex, and H_α is the Tsallis α-entropy:

$$H_\alpha := \begin{cases} \frac{1}{\alpha(\alpha-1)} \sum_j (p_j - p_j^\alpha), & \alpha \neq 1 \\ -\sum_j p_j \log p_j. & \alpha = 1 \end{cases} \tag{4}$$

The negative log-likelihood loss in Equation 2 is then:

$$\mathcal{L}(\theta) = \sum_{i=1}^{|S|} \sum_{t=1}^{T} l_\alpha(z_t(\theta, w_{<t}), w_t), \tag{5}$$

where $l_\alpha(z_t, w_t)$ is the proposed α-entmax loss:

$$l_\alpha(z_t, w_t) := (p_\theta - e_w)^T w_t + H_\alpha(p_\theta), \tag{6}$$

with $p_\theta = \alpha\text{-entmax}(z_t)$ and e_w a one-hot vector representing the ground truth word w.

3. Neural Dialogue Language Model

The generative systems follow the implementation introduced by Budzianowski and Vulić (2019), finetuning GPT-2 on the MultiWOZ Dataset[3], allowing to transfer language knowledge from this model to our systems.

3.1. Preprocessing MultiWOZ

The dataset is delexicalized, which is crucial to learn value independent parameters: all numeric, domain and non-domain specific slot values are replaced by generic slots, such as *[value_pricerange]* and *[domain_name]*. Besides, a *database pointer* is created, saving information regarding the number of entities matching the user query, which will be useful for the evaluation.[4]

[2] Using the entmax repository: `https://github.com/deep-spin/entmax`, last accessed on 29-06-2020.

[3] Based on: `https://github.com/huggingface/transfer-learning-conv-ai`, last accessed on 13-04-2020.

[4] Roughly following: `https://github.com/budzianowski/multiwoz`, last accessed on 17-06-2020.

3.2. Language Model Input

The input is composed of *word_level*, *token_level* and *position_level*, as in Figure 1. Following Ham et al. (2020), *dialogue_state embedding* are added to separate the input: <*belief*> for the *belief_state*, <*db*> for the *database_state*, <*usr*> for a user and <*sys*> for a system utterance. Besides, <*bos*> and <*eos*> identify the beginning and end of the sentence, with <*pad*> tokens being added to make the inputs' length constant.

Belief State. It summarizes the relevant information from the conversation history. In practice, it is composed of the informable and requestable slots with corresponding values in natural language, in the format:

```
Domain_1 Slot_1 Value_1 ... Slot_n Value_n
Domain_k Slot_1 Value_1 ... Slot_m Value_m.
```

Database State. It represents the database information, informing how many entities obey to the given restrictions. The database state is a simple text representation of the database pointer, following the form:

```
Domain_1 n_1... Domain_k n_k.
```

3.3. Softmax vs α-entmax

At training time, the loss is calculated between the scores returned by the decoder and the provided labels. Depending on the decoding technique, it is either the cross-entropy loss, if the system uses the softmax distribution, or the α-entmax loss, in the case of the α-entmax distribution. In the generation phase, the input is similar to the training phase's, but with no system response. The returned scores are then to be transformed into one of the two possible distributions, softmax or α-entmax, producing a vector of probabilities over the vocabulary. The decoding strategy will determine the chosen token.

3.4. Decoding Strategies

Greedy Search. The chosen token w_t is the one with the highest probability value at each timestep t: $w_t = \text{argmax}_w\, p_\theta(w|w_{<t})$. It has been reported to have some problems, such as the model starting to repeat itself; or the fact that sometimes high probability phrases are "hidden" behind less likely words, which end up being ignored (Shao et al., 2017).

Sampling. Aiming at avoiding repetition, sampling strategies introduce stochastic decisions in the generation process, by randomly picking the next word w_t given the distribution $w_t \sim p_\theta(w|w_{<t})$. To make the distribution sharper, it is common to lower down the temperature of the softmax, increasing the higher probabilities and decreasing the lower ones.

Top-k Sampling. Fan et al. (2018) proposes to restrict the sampling process to the k most probable words, reducing the probability of choosing out-of-the-box words, but also making the process more deterministic. Despite its good performance in text generation, it has the drawback of not dynamically adapting the number of words to consider.

Nucleus Sampling. Holtzman et al. (2020) proposes to sample from the smallest subset of words whose cumulative probability exceeds p, eliminating the possibility of choosing the less likely words and allowing to contract and expand the number of candidates dynamically, depending on the probability distribution.

α-entmax Sampling. In the metrics presented so far, the models sample from a new version of the softmax distribution at testing time, whose sparsity was not learned. To overcome this problem, we can apply the entmax transformation to the model scores, which, similarly to top-k and nucleus sampling, prevents implausible words from receiving any probability mass. The chosen token w_t at timestep t is:

$$w_t \sim p_\theta(w|w_{<t}) = \alpha\text{-entmax}(z_t(\theta, w_{<t})), \qquad (7)$$

where z_t are the scores given by the model.

3.5. Performance Evaluation

ϵ-perplexity. It translates into the model's ability to predict the next word, given the context (Jurafsky and Martin, 2019). In a Language Model, perplexity is the exponential average log-likelihood of a sequence. Computing it with sparse language models requires a smoothing process, by adding ϵ to all the terms followed by renormalization over the vocabulary size $|\mathcal{V}|$ (Martins et al., 2020):

$$\epsilon\text{-ppl}(w) = \exp\left\{-\frac{1}{T}\sum_{t=1}^{T}\log\frac{p_\theta(w_t|w_{<t})+\epsilon}{1+\epsilon|\mathcal{V}|}\right\} \qquad (8)$$

Sparsemax Score. Based on the sparsemax loss (Martins and Astudillo, 2016), it is defined by:

$$\text{sp} = 1 - \min\{l_2(z,w)|\,\text{sparsemax}(z) = p_\theta\}, \qquad (9)$$

where l_2 is as in Equation 6.

Inform Rate. Proposed along with the MultiWOZ dataset, it aims at assessing whether the offered entity matches all the constraints specified in the user goal. Calculated at the dialogue level, it checks if, given the context, there was indeed an available option in the dataset.

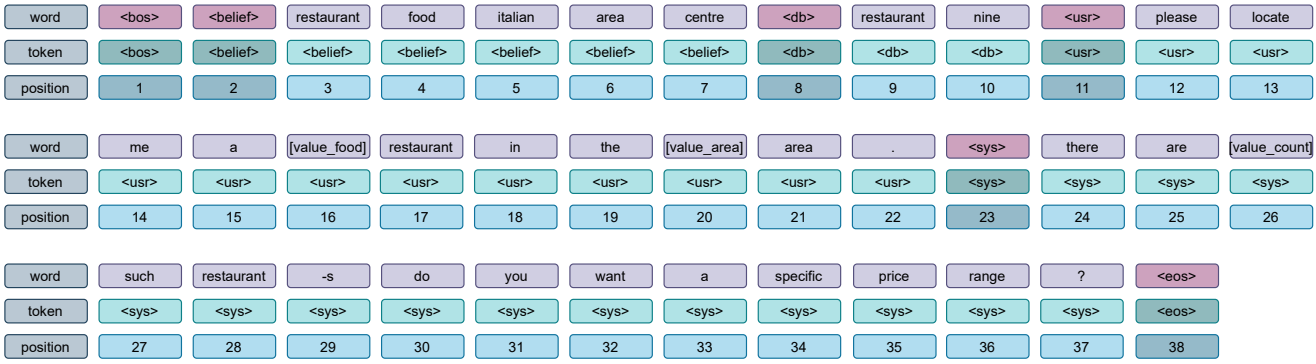


Figure 1: Language Model input.

Success Rate. Also suggested with the MultiWOZ dataset, its goal is to evaluate whether all the requestable slots were provided to the user, being calculated at the dialogue level too.

BLEU. Measures the fluency of the answer by analysing the overlap of n-grams (sequences of n words) between the proposed response and a set of one or more reference sentences, regardless of the word order (Papineni et al., 2002).

METEOR. Based on alignments between the generated sentence and the reference, creates matchings between unigrams of two different strings, being based on exact, stem, and synonym matches between words and phrases (Banerjee and Lavie, 2005).

BERTScore. Calculates a similarity between each token of both sentences, but instead of n-gram matching, this similarity is computed as a sum of cosine similarities between contextual embeddings of the tokens given by BERT (Devlin et al., 2019), being context aware (Zhang et al., 2020).

4. Experiments

4.1. Decoding Techniques

The results for the different decoding techniques are shown in Table 1. The values for ϵ-ppl are in accordance to the expected, with low values for stochastic methods and high values for more deterministic ones, and the opposite happening for sparsemax score. Inform presents a regularity throughout the techniques, suggesting the systems' ability to effectively attend to the database state information. Success, however, presents a diverse range of values, proposing the best performance for greedy, nucleus and greedy sampling from α-entmax. Both BLEU and BERTScore are evidences of the optimized fluency of these three techniques. The values for METEOR suggest a similar behaviour, but in a shorter scale, being less conclusive. The best overall performance is for greedy sampling from α-entmax, presenting the highest score for Inform and Success, and therefore to the Tune Metric also, given by $0.5 \times (\text{Inform} + \text{Success}) + \text{BLEU}$. It is closely followed by greedy sampling and nucleus sampling, with top-k afterwards, and sampling and α-entmax sampling being the last in the ranking.

4.2. Context Importance

The importance of the structured context can be questioned, as in a real life application, identifying this information requires the implementation of an extra system. Some experiments were conducted with different formats for the context, to be found in Table 2, where the results are for nucleus sampling at test time, the best sampling technique reported in Budzianowski and Vulić (2019). There is an evident drop in systems with no belief state and no database state, confirming the importance of this structured information. The most surprising result is for experiences with only the last 3 utterances as context, even outperforming some techniques in most of the metrics. However, the low value for BERTScore suggests some faults in the generated text.

4.3. Discussion

Although the metrics transmit an idea of the model's performance, some surprising results ask for further inspection. After an extensive analysis, despite the similar performance in standard scenarios, there is a contrast among the techniques' performance in some specific situations:

- The standard **context importance** is confirmed: the belief state is crucial to the dialogue history awareness; systems with database state are always aligned with the available entities from the database; the 3 utterances system occasionally produces strange repetitive text. Its good scores in Inform and Success rates suggest that these metrics are not enough to evaluate a system's performance — even misunderstanding the

Table 1: Different models performance for MultiWOZ 2.0.

	ϵ-ppl (↓)	sp (↑)	Inform (↑)	Success (↑)	BLEU (↑)	BERTScore (↑)	METEOR (↑)	Tune Metric (↑)
Sampling	**2.3074**	**0.8443**	64.4	37.1	21.02	24.36	18.91	71.77
Greedy	25.3800	0.7695	64.6	53.4	**29.46**	**34.81**	**19.52**	88.46
Top-k	3.8036	0.8412	65.4	44.2	24.43	28.89	19.06	79.23
Nucleus	10.2294	0.8204	65.2	54.6	27.80	32.53	19.43	87.18
α-entmax sampling	2.4922	0.8440	63.4	35.8	21.26	24.11	18.87	70.86
α-entmax greedy	25.6698	0.7686	**66.8**	**57.4**	29.44	33.38	19.34	**91.54**

Table 2: Nucleus sampling performance for different types of context.

	ϵ-ppl (↓)	sp (↑)	Inform (↑)	Success (↑)	BLEU (↑)	BERTScore (↑)	METEOR (↑)	Tune Metric (↑)
Full context	10.2294	0.8204	65.2	54.6	27.80	32.53	19.43	87.18
No belief state	14.4673	0.7999	50.5	35.3	25.44	29.22	19.04	68.34
No database state	14.0166	0.8024	52.7	36.4	26.02	29.47	19.07	70.57
Only last utterance	13.5958	0.8045	51.3	36.4	25.30	29.54	19.00	69.15
Last 3 utterances	11.7259	0.8119	67.3	52.1	25.21	23.12	19.11	84.91

user's specificities, it can randomly suggest an entity (when it should be the case) and provide the necessary requestables, being wrongly considered an informable and successful dialogue.

- The systems sometimes generate **inadequate slots** given the domain, having an impact in the scores for Inform and Success.
- There is frequently **low understanding of the user's intentions** when the information is not present in the belief state. These cases are not contemplated in the Inform nor the Success metrics.
- When the user is looking for a simple suggestion, some techniques struggle to move forward in the conversation and get stuck in the process of narrowing down the search, **insisting instead of suggesting**.
- Another recurring fault is **booking without enough information**, which is grasped by Inform and Success.
- In some cases, the systems **outperform the reference answer**. In these cases, the metrics using sentence similarity do not translate into a good evaluation.

Some of these particularities raised the question of whether the evaluation methodology is appropriate. At one side, despite being created to evaluate MultiWOZ dialogues, Inform and Success are only evaluated at dialogue level, missing some important aspects. At the other side, BLEU, METEOR and BERTScore were not designed for this task, but are able to be measured at the turn level. Since they have been imported from machine translation, the question of whether they translate into the human perception of quality arises. Despite having previously been shown to correlate with human judgement in a goal-oriented setting, this study evaluated the quality of the translation from dialogue acts into a proper sentence, which is more similar to the machine translation task (Sharma et al., 2017). As the generation process evolves from pipeline towards an end-to-end approach, some conclusions should be revisited.

5. Human Evaluation

To determine how indicative the automatic metrics are of the sentences' quality, we will resort on static human evaluation at the turn level, followed by a correlation study between the automatic metrics and human annotations.

5.1. Evaluation Dimensions

Little has been done regarding turn level goal-oriented dialogue evaluation, therefore common practices from open-domain will serve as inspiration. Given the relevance of frequently used evaluation dimensions (Finch and Choi, 2020) to the goal-oriented framework, we narrow them down to 5 crucial fields of evaluation: *Grammaticality*, *Informativeness*, *Relevance*, *Consistency* and *Overall Quality*.

Grammaticality. It evaluates the grammar construction of the sentence, detecting if it is free of grammatical and semantic errors. Scored in a scale of 1–3:

1. The answer is not fluent at all, being poorly structured and almost not understandable.
2. There are some minor flaws regarding the grammar, but the sentence is understandable.
3. The answer is fluent and grammatically perfect, with no flaws.

Informativeness. It assesses whether the system's answer brings relevant information to the table and can be measured considering the amount of user queries tackled by the system. In a scale of 1–3:

1. The reply is not informative and not helpful in reaching the conversation goal.
2. The answer is slightly informative, but one would like to get some more information at that point. It can be used when not all the user queries are covered.
3. The system utterance is totally informative, tackling all the required queries.

Relevance. Aims to evaluate whether the response is appropriate given the user query, using a 1–3 scale:

1. The response does not make any sense given the dialogue history, as it is completely out of context.
2. The answer can not be considered inappropriate, but there were more relevant aspects to tackle or replies to be given at that stage.
3. The system reply is the most appropriate given the user utterance.

Consistency. It was introduced to assess any contradictions within the dialogue, both with system and user utterances. In a scale of 1–3:

1. The answer is not consistent with what has been said previously, holding some sort of contradiction, with either user or system previous utterances.
2. The history awareness is not clear, making it hard to determine if the answer is consistent or not. It can also be used in the cases where the system assumes to know a certain type of information which has not been tackled.
3. The sentence is consistent and in accordance with the whole dialogue history.

Overall Quality. It is a less strict dimension, aiming at gathering a more personal opinion regarding the answer, in a 1–5 scale. Overall Quality represents how satisfied is the user with the response, taking into account the scores for the other 4 metrics and giving the annotators space to differentiate between answers.

5.2. Annotation Process

The annotation process consisted of rating 6 possible responses to the same dialogue context, as schematized in Figure 2. The evaluation was divided in two parts: the preliminary experience and the extended evaluation. The annotators were provided annotation guidelines.

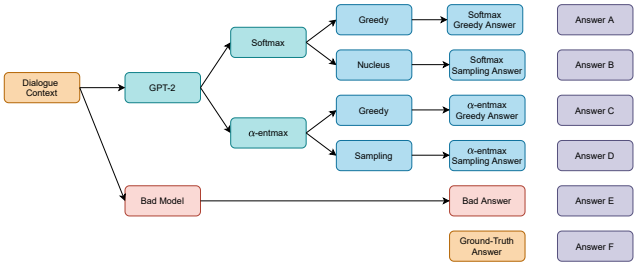


Figure 2: The possible answers provided by the systems.

Preliminary Experience. 92 students from the *"Natural Language"* subject at Instituto Superior Técnico evaluated the same 6 answers for 4 dialogues, resulting in 24 annotated responses. Relevance and Consistency were initially proposed to be binary, and therefore evaluated between 0–1; after the feedback, they evolved to a scale of 1–3.

Extended Evaluation. The group of annotators was composed of 9 people, from the age of 22 to 36, 5 female and 4 male. Four of the participants had never annotated before. Each annotator scored 6 answers for 4 dialogues. As the goal was to evaluated the maximum number of sentences, each annotator had an individual set of 3 dialogues, and a common dialogue among all, to allow measuring the agreement, which was also present in the preliminary phase (serving as a control variable). In this phase, 168 possible responses were evaluated.

5.3. Results

Annotation Scores. Table 3 comprises the average scores of the techniques at each dimension, with the best results from the four models highlighted. The annotators preferred the greedy techniques, in accordance with the automatic metrics. For a more thorough evaluation of each sampling technique, box plots are presented in Figure 3.

The greedy techniques present an overall better performance than the sampling techniques and a very similar performance between each other. However, α-entmax greedy outperforms softmax greedy in the Informativeness dimension. Although softmax greedy presents a higher Consistency average value, Figure 3b shows that α-entmax greedy results are more concentrated in the upper part, demonstrating the influence that some outliers can have in the mean values. Nevertheless, the span of Overall Quality is more consistent in softmax greedy, suggesting its general better performance.

α-entmax sampling has the longest span of values across all the dimensions, suggesting to have the worst performance. However, in the Informativeness dimension, it presents a higher median value than both softmax sampling and greedy, meaning that at least half of the responses were

Table 3: Average scores at each evaluation domain.

	Grammaticality	Informativeness	Relevance	Consistency	Overall Quality
Softmax sampling	2.8929	2.1825	2.3929	2.4286	3.6548
α-entmax sampling	2.6746	2.2421	2.2103	2.2857	3.6746
Softmax greedy	2.8571	2.3016	**2.5992**	**2.6389**	**4.1270**
α-entmax greedy	**2.9206**	**2.4484**	2.5119	2.5992	3.9365
Bad	2.4206	1.8095	1.7698	1.9563	2.6468
Original	2.8929	2.2738	2.5278	2.6746	4.1310

rated with the maximum score. From Figure 3d, it is possible to understand that, although α-entmax can produce good results, there is lack of regularity in its performance, generating sometimes replies which are grammaticaly incorrect, not informative, not relevant and not consistent with the conversation history.

Inter Annotator Agreement. Used to evaluate the reliability of the human evaluation task and its reproducibility, it shows how uniform the annotations are. To allow the measurement between a group of multiple annotators, *Fleiss's Kappa* (Fleiss, 1971) is used, ranging from 0–1. Table 4 shows that, for all the dimensions, there is either fair (0.2–0.4) or moderate agreement (0.4–0.6). The lowest values suggest some difficulties in annotating certain dimensions. According to Celikyilmaz et al. (2020), low agreement between annotators can also indicate that there are not significant differences in the possible answers, which was the case in some dialogues. From the preliminary to the extended evaluation phase, the agreement increased in Grammaticality, Consistency and Overall Quality, while it decreased in Informativeness and Relevance. This drop would be expected in the Relevance and Consistency dimensions, as the scale span was enlarged. In what comes to Informativeness, a possible explanation can be the difficulty to score it in dialogues where there are no specific queries to fill, making it more ambiguous.

Table 4: *Fleiss Kappa* for the two experiences.

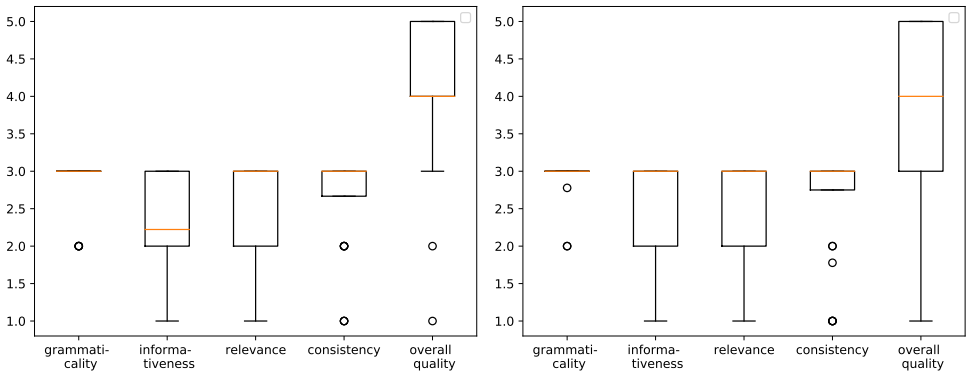
	Preliminary 4 dialogues	Preliminary common dialogue	Extended common dialogue
Grammaticality	0.2517	0.1565	0.2969
Informativeness	0.3221	0.3261	0.2901
Relevance	0.5205	0.5421	0.4904
Consistency	0.4305	0.4633	0.5857
Overall Quality	0.2225	0.2572	0.2945

Correlation with Automatic Metrics. The calculated correlation coefficients are the *Pearson*, measuring the linear relationship, and the *Spearman*, determining how well the two variables correlate through a monotonic function. The values can range between -1.0 and 1.0, with 0.0 meaning no correlation. The values for segment and system correlation can be found in Tables 5 and 6.

At the segment level, the highest correlation value is between Overall Quality and BLEU, with a still low Spearman of 0.3309. BERTScore and METEOR also present the highest correlation values for Overall Quality, which is a good indicator regarding their fidelity. Contrary to what would be expected, all metrics correlate poorly with Grammaticality, suggesting a low understanding of language nuances. For a certain dimension, the correlation with different metrics are relatively close to each other, proposing agreement among the three metrics.

At system level, the correlation values are significantly higher. The Spearman values are basically the same for all the metrics, confirming that the three metrics are able to extract very similar information, despite being calculated in different manners. We can infer that no metric is superior to one other in terms of correlation with human judgement. Taking BERTScore as an example, although it is calculated using contextual embeddings and therefore able to understand some context, its behaviour is identical to BLEU, which simply resorts on similarity between word embeddings. This suggests that the sentence alone is not enough to grasp the whole meaning behind a reply in a goal-oriented dialogue. There is a contrast between Pearson and Spearman values for correlation, with the latter holding much higher results. It indicates that, for close values of automatic metric scores, the slight difference between them is not enough to choose between the systems. The high correlation with Relevance, Consistency and Overall Quality indicate that the three automatic metrics can be useful to compare the performance of different systems, despite being less informative when evaluating a sentence alone. However, it was not expected that the highest correlation values are for Consistency, as this domain is probably the one with least information present in each sentence. It can indicate that these correlation values have low fundament, representing a coincidence of high and low scores, and can be not be considered representative. Many authors report how difficult it is to show correlation between human annotations and automatic metrics, such as in Mathur et al. (2020).

In Figure 4, it is possible to see the correlation between Overall Quality and the metrics, at the system level. It con-



(a) Softmax greedy.

(b) α-entmax greedy.

(c) Softmax sampling.

(d) α-entmax sampling.

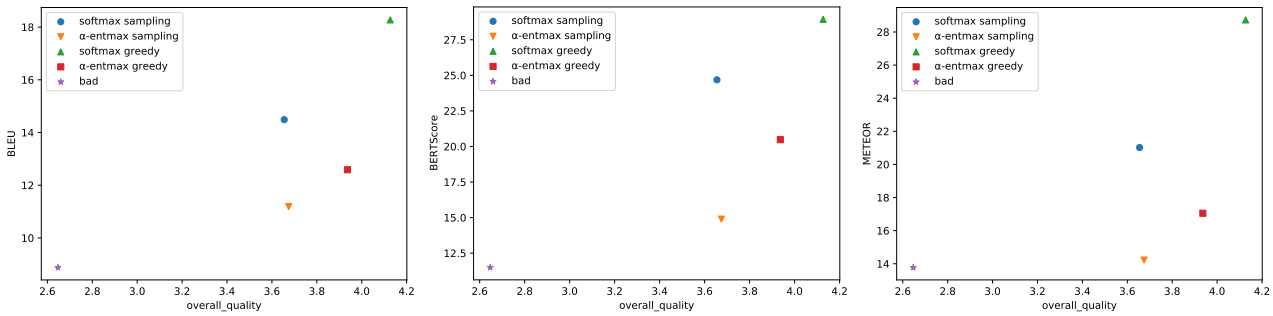
Figure 3: Box plot of the annotation results.

Table 5: Correlation at the segment level.

	BLEU		BERTScore		METEOR	
	Pearson	Spearman	Pearson	Spearman	Pearson	Spearman
Grammaticality	0.1257	0.1757	0.1575	0.1615	0.1201	0.1182
Informativeness	0.0917	0.1798	0.0941	0.1333	0.0530	0.0766
Relevance	0.1673	0.2423	0.2173	0.2258	0.1789	0.2185
Consistency	0.2107	0.3076	0.2514	0.2560	0.2046	0.2512
Overall Quality	0.2360	**0.3309**	**0.2975**	0.2805	0.2341	**0.2679**

Table 6: Correlation at the system level.

	BLEU		BERTScore		METEOR	
	Pearson	Spearman	Pearson	Spearman	Pearson	Spearman
Grammaticality	0.3535	0.5508	0.4367	0.5508	0.1201	0.1182
Informativeness	0.1962	0.4857	0.2567	0.4857	0.2081	0.4857
Relevance	0.3765	0.8857	0.4593	0.8857	0.4149	0.8857
Consistency	0.5002	**0.9429**	0.5747	**0.9429**	0.5348	**0.9429**
Overall Quality	0.4497	0.8286	0.5177	0.8286	0.4792	0.8286



(a) BLEU vs Overall Quality. (b) BERTScore vs Overall Quality. (c) METEOR vs Overall Quality

Figure 4: Plot of automatic metrics vs Overall Quality at the system level.

firms that there is slight correlation between scores among the generation techniques. Both the metrics and human judgement find in greedy sampling from softmax the highest generation quality, closely followed by greedy sampling from α-entmax. Nevertheless, automatic metrics can lead one into poor conclusions, which is the case of sampling from softmax: although humans find it the worst quality technique, automatic metrics rate it as the second best one. Besides, despite their close scores in Overall Quality, softmax sampling and α-entmax sampling have distinct automatic metrics values, confirming the possibility of high quality answers with less overlap.

To conclude, metrics such as BLEU, BERTScore and METEOR can give useful information in the comparison between different systems. In the context of answer generation in goal-oriented dialogues, these metrics are able to extract similar features, being in concordance with each other. However, as these metrics rely on the comparison with a database reference, they never recognize a better sentence than the original and techniques with more word diversity are naturally damaged. The lack of history context awareness leads these metrics into the inability of grasping certain language nuances.

6. Conclusions

In this work, we studied how different techniques can influence the quality of a generated automatic reply, in a goal-oriented setting. The main achievements lie in the application of sparse attention mechanisms to automatic response generation in the goal-oriented setting, making use of α-entmax. Although stochastic strategies were found to have their positive attributes, we conclude that, in goal-oriented dialogue generation, the prime systems rely on greedy techniques, using either the standard softmax or the proposed α-entmax. Moreover, we concluded that goal-oriented response generation benefits from having a more informative context, as it significantly improves the dialogue awareness. The thorough analysis of the systems' behaviour led to realizing that many aspects are not grasped by the chosen automatic evaluation metrics, motivating a further analysis.

Futhermore, we successfully conducted a study to find correlations between the adopted automatic metrics and human perception of quality. A set of evaluation dimensions was developed, supported by some illustrative guidelines, allowing the collection of a significant amount of reliable human annotations. After a probabilistic analysis, we found that BLEU, METEOR and BERTScore substantially correlate with human judgement, being useful to roughly compare the performance of different systems. Nonetheless, these metrics are inappropriate to understand nuances in cases where the systems show a similar performance, making it essential to resort on human evaluation for a more detailed comparison.

References

Dzmitry Bahdanau, Kyung Hyun Cho, and Yoshua Bengio. Neural Machine Translation by Jointly Learning to Align and Translate. In *Proc. of the International Conference on Learning Representations*, 2015.

Satanjeev Banerjee and Alon Lavie. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In *Proc. of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, 2005.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. A Neural Probabilistic Language Model. *Journal of Machine Learning Research*, 3:1137–1155, 2003.

Paweł Budzianowski and Ivan Vulić. Hello, It's GPT-2 - How Can I Help You? Towards the Use of Pretrained Language Models for Task-Oriented Dialogue Systems. In *Proc. of the Workshop on Neural Generation and Translation*, 2019.

Asli Celikyilmaz, Elizabeth Clark, and Jianfeng Gao. Evaluation of Text Generation: A Survey. 2020. URL `https://arxiv.org/abs/2006.14799`.

Jan Deriu, Alvaro Rodrigo, Arantxa Otegi, Guillermo Echegoyen, Sophie Rosset, Eneko Agirre, and Mark Cieliebak. Survey on Evaluation Methods for Dialogue Systems. *Artificial Intelligence Review*, 2020.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proc. of the Conference of the North American Chapter of the Association for Computational Linguistics*, 2019.

Angela Fan, Mike Lewis, and Yann Dauphin. Hierarchical Neural Story Generation. In *Proc. of the Annual Meeting of the Association for Computational Linguistics*, 2018.

Sarah E. Finch and Jinho D. Choi. Towards Unified Dialogue System Evaluation: A Comprehensive Analysis of Current Evaluation Protocols. In *Proc. of the Annual Meeting of the Special Interest Group on Discourse and Dialogue*, 2020.

Joseph L. Fleiss. Measuring Nominal Scale Agreement Among Many Raters. *Psychological Bulletin*, 76:378–382, 1971.

Rob Gaizauskas, James Law, and Emma Barker. Investigating Spoken Dialogue to Support Manufacturing Processes. Technical report, University of Sheffield, 2018.

Sergey Golovanov, Rauf Kurbanov, Sergey Nikolenko, Kyryl Truskovskyi, Alexander Tselousov, and Thomas Wolf. Large-Scale Transfer Learning for Natural Language Generation. In *Proc. of the Annual Meeting of the Association for Computational Linguistics*, 2019.

Donghoon Ham, Jeong-gwan Lee, Youngsoo Jang, and Kee-eung Kim. End-to-End Neural Pipeline for Goal-Oriented Dialogue System using GPT-2. In *Proc. of the Annual Meeting of the Association for Computational Linguistics*, 2020.

Ari Holtzman, Jan Buys, Leo Du, Maxwell Forbes, and Yejin Choi. The Curious Case of Neural Text Degeneration. In *Proc. of the International Conference on Learning Representations*, 2020.

Daniel Jurafsky and James Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition*. 3rd edition, 2019.

Andre F.T. Martins and Ramon F. Astudillo. From Softmax to Sparsemax: A Sparse Model of Attention and Multi-Label Classification. In *Proc. of the International Conference on Machine Learning*, 2016.

Pedro Henrique Martins, Zita Marinho, and André F. T. Martins. Sparse Text Generation. In *Proc. of the Conference on Empirical Methods in Natural Language Processing*, 2020.

Nitika Mathur, Johnny Tian-Zheng Wei, Markus Freitag, Qingsong Ma, and Ondrej Bojar. Results of the WMT20 Metrics Shared Task. In *Proc. of the Conference on Machine Translation*, 2020.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zu. BLEU: A Method for Automatic Evaluation of Machine Translation. In *Proc. of the Annual Meeting on Association for Computational Linguistics*, 2002.

Ben Peters, Vlad Niculae, and André F. T. Martins. Sparse Sequence-to-Sequence Models. In *Proc. of the Annual Meeting of the Association for Computational Linguistics*, 2019.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language Models are Unsupervised Multitask Learners. *OpenAI Blog*, 2019. URL `https://openai.com/blog/better-language-models/`.

Louis Shao, Stephan Gouws, Denny Britz, Anna Goldie, Brian Strope, and Ray Kurzweil. Generating High-Quality and Informative Conversation Responses with Sequence-to-Sequence models. In *Proc. of the Conference on Empirical Methods in Natural Language Processing*, 2017.

Shikhar Sharma, Layla El Asri, Hannes Schulz, and Jeremie Zumer. Relevance of Unsupervised Metrics in Task-Oriented Dialogue for Evaluating Natural Language Generation. 2017. URL `http://arxiv.org/abs/1706.09799`.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to Sequence Learning with Neural Networks. In *Proc. of the International Conference on Neural Information Processing Systems*, 2014.

Ashish Vaswani, Google Brain, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention Is All You Need. *Proc. of the Conference on Neural Information Processing Systems*, 2017.

Tsung Hsien Wen, Milica Gašić, Nikola Mrkšić, Pei Hao Su, David Vandyke, and Steve Young. Semantically Conditioned LSTM-based Natural Language Generation for Spoken Dialogue Systems. In *Proc. of the Conference on Empirical Methods in Natural Language Processing*, 2015.

Tsung Hsien Wen, David Vandyke, Nikola Mrkšíc, Milica Gašíc, Lina M. Rojas-Barahona, Pei Hao Su, Stefan Ultes, and Steve Young. A Network-based End-to-End Trainable Task-oriented Dialogue System. In *Proc. of the Conference of the European Chapter of the Association for Computational Linguistics*, 2017.

Thomas Wolf, Victor Sanh, Julien Chaumond, and Clement Delangue. TransferTransfo: A Transfer Learning Approach for Neural Network Based Conversational Agents. 2019. URL `http://arxiv.org/abs/1901.08149`.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. BERTScore: Evaluating Text Generation with BERT. In *Proc. of the International Conference on Learning Representations*, 2020.