Drastically increasing accuracy in Rasa and other ML-based bot platforms: from 60% to 90% accuracy

One of the main problems with the current generation of chatbots is that they require large amounts of training data. If you want your chatbot to recognize a specific intent, you need to provide it with a large number of sentences that express that intent. Until now, these large training corpora had to be generated manually. This is a time-consuming task rather than a creative one, and it makes successful bot development very costly. To solve this problem, at Bitext we offer our Natural Language Generation (NLG) technology, which automatically generates many different sentences with the same meaning as the original one, in order to automate the most resource-intensive part of the bot creation process.

The Rasa chatbot-building platform is becoming increasing popular, so we have chosen it for our tests. We tested how Rasa NLU can benefit from the NLG approach, comparing NLU models trained with hand-tagged sentences with models extended with no effort via NLG. Our tests show that if we train with just 1 or 2 example sentences per intent in Rasa, we get very bad results (3% accuracy). When we train with 10 sentences per intent, we only get mediocre results (68% accuracy). In contrast, extending these hand-tagged corpora with additional variants generated via NLG, we get a drastic improvement and overall really good results (93% accuracy).

We have carried out two different tests (A and B). Both use the five following intents related to the management of lights in a house:

- Switch on lights (*switch on the lights in the living room*)
- Switch off lights (*switch off the lights in the living room*)
- Change the color of lights (*change the lights to blue*)
- Dim lights (*dim the living room lights to 20%*)
- Program lights for a specific hour (*program the garden lights for 21:00*)

In both tests, we have also used the same five types of slots: ACTION, OBJECT, PLACE, PERCENTAGE and HOUR.

In the first test (A), we trained two different models, both of them using Rasa's default pipeline. A first model (A1) was trained with only 12 hand-tagged sentences: 2 or 3 sentences per intent. A second model (A2) was trained with a set of 455 sentences, around 90 per intent. Those sentences were the result of automatically generating variants of the sentences in A1 using the Bitext NLG system. In fact, the system generated 569 sentences. 80% of them (455) were used for training; the rest (114), randomly selected, were set aside as the evaluation set.

To evaluate both A1 and A2, we used those 114 sentences. We analyzed them with A1 and A2 and with the same standard pipeline used in training. We recorded results for both intent detection and slot filling.

	Accuracy	
	Intent detection	**Slot filling**
A1: With hand-tagged training set	84%	3%
A2: With NLG-generated training set	99%	93%

For the intent detection task, we obtained relatively good results with both models. That said, A2 (the model trained with the extended NLG training set) shows an improvement of 5% over A1 (the model trained with the hand-tagged training set). In the slot filling task, A2 shows a drastic improvement of 90% over A1.

The second test (B) was very similar to the first one. The only difference is the number of sentences used in the training and evaluation sets. In this case, the first model (B1) was trained with a hand-tagged training set of 50 sentences (10 per intent). Using those sentences as input, our Bitext NLG system generated 1132 variants, around 180 per intent. 80% of them were used to train the second model B2, and the rest of sentences (226) were used as the evaluation set.

	Accuracy	
	Intent detection	**Slot filling**
B1: With hand-tagged training set	95%	68%
B2: With NLG-generated training set	99%	93%

In this case, the results were also good in intent detection both with B1 and B2. In slot filling, B1 gets better results than A1, but not good enough to have the right user

experience. Once again, the model trained with the extended NLG training set (B2) improves drastically those results, reaching 93% accuracy.

In summary, the Bitext NLG system lets you create big training sets with no effort. If you only want to write one or two sentences per intent, our system is able to generate the rest of variants needed to go from really poor results to great accuracy. And even if you want to write tens of variants per intent, our system will also significantly increase the accuracy of your model, obtaining really good results. We have carried out these tests with Rasa, but our conclusions are relevant for ML-based bot platforms in general. We can conclude that our NLG system is able to drastically improve the results of bot platforms that are highly dependent on training data.