

Core AI Research and Statistic Report for RASA

Sparse Feature: CountVectorizer vs TF-IDF Featurizer:

	CountVectorizer	TF-IDF Featurizer
Training 100 epoch	53.55%	2.84%
Training 2000 epoch	83.0%	87.7%
Feature Size	900MB	1.2GB

- TF-IDF features perform better than counterVector when training with enough time. As a drawback, it needs more space to store tf-idf score of each element, while CountVector just needs to store which word and its count in the vocabulary.

Dense vs Sparse Feature:

- Test on SARA

	Featurizer	Training Time	Infer Time	Accuracy	Model Size
Dense	ConveRT	17:14 min	1:59 min	0.86%	Medium
	BERT	17:24 min	7:20 min	80%	Large
Sparse	Count	17:06 min	1:23 min	78%	Small

- In the general case, we should not use BERT due to Industry product
- Use ConveRT if the domain of custom is general, do not include too much specific vocabulary and just use English in conversation, because it performs better than Sparse vector used in Athenka now
- If the customer wants to build a bot for their specific subject, we use the sparse feature so that bot can cover these case, back draw is it takes more space to store feature for each bot, not perform well in a more general case like:
 - Eat to Life vs Life to Eat (drawback of countVector)

Merge Dense and Sparse feature:

- Dataset: SQUAD
- Multi-Response Selector
- Intent Classifier: DIET
- How this works:
 - For each word W:
 - Find Sparse features for W
 - Embed Dense feature for W
 - Two types of vectors are concatenated into a unique vector U
 - Vector U is concatenated with a length of the input sentence, this is the input of DIET intent classifier.

	ConveRT Featurizer	Countvectorizer	ConveRT + Count
Accuracy	85.16%	53.5%	86.1%
Model weight	237MB	921MB	2.54GB

- Try to merge both advantages of the Sparse and Dense features, the accuracy is improved a little
- The drawback is we need more disks to store these feature

FAQ - Response Selector Strategy

- Response Selector is Intent Classification only vs DIET
- DIET tries to classify which intent, then Response Selector tries to classify which question to map the answer.
- 800 test case FAQ

	One Selector for all FAQ	Each Set FAQ is one Selector
Number Selector	1	30
DIET	100% for sure	93% (detect right intent)
Train Selector 100 epoch	---	48.8%
Train Selector 300 epoch	23.8%	49.1%

- Should use multi-selector if the user can build many sets of FAQ

FAQ - Response Selector Perform on GPU:

- 42 Model: 1 model for DIET + 41 model for Response Selector
- 150 epoch for DIET
- 100 epoch for each 41 Response Selector

	GPU	CPU
Training Time	8 min	28 min
Model Size	1.4GB	1.4GB

DIET - Multi-Intent testing

- With CountVector Feature:

200 intents	1 min 8s on GPU	50MB
-------------	-----------------	------