Training a Chatbot with Microsoft LUIS: Effect of Intent Imbalance on Prediction Accuracy

Elayne Ruane
elayne.ruane@ucdconnect.ie
University College Dublin, Ireland
Lero - The Irish Software Research Centre

Robert Young
robert.young2@ucdconnect.ie
University College Dublin, Ireland
Lero - The Irish Software Research Centre

Anthony Ventresque
anthony.ventresque@ucd.ie
University College Dublin, Ireland
Lero - The Irish Software Research Centre

ABSTRACT

Microsoft LUIS is a natural language understanding service used to train Chatbots. Imbalance in the utterance training set may cause the LUIS model to predict the wrong intent for a user's query. We discuss this problem and the training recommendations from Microsoft to improve prediction accuracy with LUIS. We perform batch testing on three training sets created from two existing datasets to explore the effectiveness of these recommendations.

CCS CONCEPTS

• **Computing methodologies** → *Natural language processing*.

KEYWORDS

dataset imbalance, chatbot, LUIS, classification accuracy

ACM Reference Format:
Elayne Ruane, Robert Young, and Anthony Ventresque. 2020. Training a Chatbot with Microsoft LUIS: Effect of Intent Imbalance on Prediction Accuracy. In *25th International Conference on Intelligent User Interfaces Companion (IUI '20 Companion), March 17–20, 2020, Cagliari, Italy*. ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3379336.3381494

1 INTRODUCTION

Microsoft's LUIS is a cloud-based language understanding API that uses Machine Learning (ML) to predict sentence meaning and extract information for training chatbots. To develop a chatbot app using LUIS we define a number of `intents`, an abstraction of a task the user may want to do using the chatbot. LUIS is a supervised learning model requiring labelled example utterances to train an intent. For example, we can define an intent `OrderTaxi` and create utterances such as *"please book me a taxi now"*.

When a natural language utterance is submitted to the chatbot, the underlying LUIS model will parse it and try to classify it. LUIS will return the top intent along with a confidence score. Due to the nature of natural language, some intents will have more variability in how they may be expressed by the user than others. For example, a `Greeting` intent may have 30 example utterances but a `MakeBooking` intent may have 200. LUIS' prediction capabilities, like most ML algorithms, can suffer when dataset imbalance occurs due to predictions towards the majority class.

IUI '20 Companion, March 17–20, 2020, Cagliari, Italy

ACM ISBN 978-1-4503-7513-9/20/03.
https://doi.org/10.1145/3379336.3381494

2 RELATED WORK

The LUIS docs discuss best practice for designing `intents` [1]. It is recommended to use 15 to 30 specific and varying example utterances for each intent, only adding further examples after training and testing. A `None` intent should contain examples that fall outside the chatbot domain and comprise 10% of all training utterances. LUIS uses non-deterministic training so if two intents are trained on very similar utterances and have similar scores for an utterance, the top intent may invert and become the second-top intent. It is recommended to have a 15% difference in scores to avoid this. As such, we want to train our chatbot such that the top intent for recognised utterances has both a high confidence score and a significant margin between the top intent and the rest.

Much work has been done on dataset imbalance for ML models. Mirończuk and Protasiewicz (2019) [5] provide a detailed survey of text classification literature including work focusing on the development of models that can handle class imbalance. Other studies look at how to address the problem by processing the dataset itself using techniques such as under- and over-sampling [4]. However, re-sampling methods are not feasible here due to the small data set size and the redundancy of duplicate examples.

The class imbalance problem often occurs in binary classification or anomaly detection scenarios that try to identify a particular event that occurs much less frequently than the majority class e.g. insurance fraud detection. In our context, we have a greater number of classes (intents) with less structured data (natural language). As such, the imbalance will affect unseen utterances differently depending on their structure and content, unlike more clear-cut problems where the issue presents itself solely as prediction towards the majority class. The black-box nature of LUIS means examples are provided as strings and any pre-processing tasks such as tokenisation, stop-word removal, and lemmatisation, are all done "under the hood". Document representation, feature selection, and even the parameters of the model itself are unknown to us. As such, we focus solely on the training utterances.

3 DATASETS

We used two existing datasets to train three LUIS apps. The first dataset, "AskUbuntu", is a question and answer dataset scraped from askubuntu.com [2] containing questions seeking technical support. The dataset is unbalanced but the None intent has the recommended amount of example utterances. The second dataset is a large multi-topic crowdsourced dataset [3] with 7 task-based intents. We trained an app, MultiTask, using all of the example utterances in the dataset. The 7 task intents are balanced but there are no example utterances for a None intent.

Table 1: Dataset Details

Name	Intents	Utterances	Balanced	Accuracy
AskUbuntu	5	162	No	100%
MultiTask	8	13,585	Yes	99.5%
MultiSmall	9	114	Yes	100%

Table 2: Batch Test Results (25 utt.) - AskUbuntu

Intent	Precision	Recall	F-score	Utterances
MakeUpdate	1	0.6	0.75	47
None	N/A	0	0	8
SetupPrinter	1	0.8	0.89	23
ShutdownComp	1	1	1	27
SoftwareRec	0.38	1	0.55	57

We randomly selected 15 example utterances for each of the 7 task intents and trained a third app, MultiSmall, and manually added example utterances for a None intent to produce a dataset that followed all recommendations.

4 RESULTS AND DISCUSSION

Using the LUIS batch testing feature, we tested each app on 5 unseen utterances per intent. The results for the AskUbuntu dataset (Table 2) appear to be affected by intent imbalance. Majority intent SoftwareRec had perfect recall but very low precision due to a high number of false positives (Figure 1), including all five test utterances for the None intent. MultiSmall showed good performance on all but one intent (BookRestaurant) due to two false positives that were very similar in structure to the example utterances. Even better results were found for MultiTask despite a lower F-score for the SearchMedia intent which was due to four false positives because the None intent had no example utterances.

We explored these results by altering the datasets and repeating batch testing. We balanced the AskUbuntu dataset by reducing the lowest scoring utterances of the larger example sets and manually adding utterances to the other intents. Precision for the SoftwareRec class increased from 0.38 to 0.45 when the dataset was balanced suggesting that intent imbalance is part of the problem. When the balanced dataset MultiSmall was *imbalanced* towards BookRestaurant with an additional 10 training utterances there was no difference to the scores of any intent. However, when the SearchMedia intent example set was increased to 100, we see its precision drop to 0.31 on the test set due to an increase number of false positives. Of course, this means the recall of the other intents also drops as they are incorrectly classified as SearchMedia.

These initial experiments suggest intent imbalance can lead to a decrease in chatbot quality due to prediction towards the majority class. Further work is needed to explore more nuanced effects of example utterance structure similarities across intents.

ACKNOWLEDGMENTS

This work was supported, in part, by Science Foundation Ireland grant 13/RC/2094

Figure 1: Batch Test results for SoftwareRec (AskUbuntu)

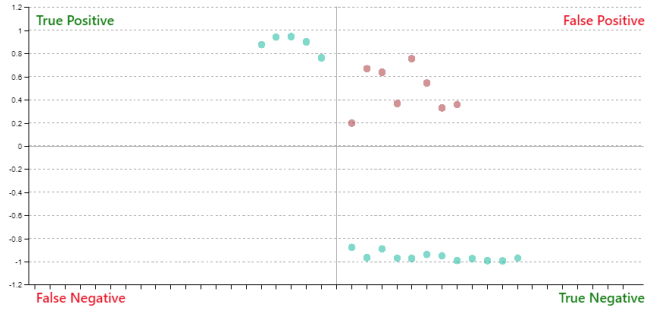

Table 3: Batch Test Results (40 utt.) - MultiSmall

Intent	Precision	Recall	F-score	Utterances
AddToPlaylist	1	1	1	15
BookRestaurant	0.71	1	0.83	15
GetWeather	1	1	1	15
None	1	0.8	0.89	9
PlayMusic	1	0.8	0.89	15
RateBook	1	0.8	0.89	15
SearchMedia	1	0.8	0.89	15
SearchEvent	0.83	1	0.91	15

Table 4: Batch Test Results (40 utt.) - MultiTask

Intent	Precision	Recall	F-score	Utterances
AddToPlaylist	1	1	1	1,944
BookRestaurant	1	1	1	1,968
GetWeather	1	1	1	1,991
PlayMusic	1	1	1	1,979
RateBook	1	1	1	1,907
SearchMedia	0.56	1	0.72	1,951
SearchEvent	0.83	1	0.91	1,845

REFERENCES

[1] 2019. Language Understanding (LUIS) Documentation - Azure Cognitive Services. https://docs.microsoft.com/en-us/azure/cognitive-services/luis/

[2] Daniel Braun, Adrian Hernandez-Mendez, Florian Matthes, and Manfred Langen. 2017. Evaluating Natural Language Understanding Services for Conversational Question Answering Systems. In *18th Annual SIGdial Meeting on Discourse and Dialogue*. Association for Computational Linguistics, Saarbrücken, Germany, 174–185. http://www.aclweb.org/anthology/W17-3622

[3] Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, David Leroy, Clément Doumouro, Thibault Gisselbrecht, Francesco Caltagirone, Thibaut Lavril, et al. 2018. Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. *arXiv preprint arXiv:1805.10190* (2018).

[4] Sotiris Kotsiantis, Dimitris Kanellopoulos, Panayiotis Pintelas, et al. 2006. Handling imbalanced datasets: A review. *GESTS International Transactions on Computer Science and Engineering* 30, 1 (2006), 25–36.

[5] Marcin Michał Mirończuk and Jarosław Protasiewicz. 2018. A recent overview of the state-of-the-art elements of text classification. , 36–54 pages.