Avrupa Bilim ve Teknoloji Dergisi
Özel Sayı 32, S. 445-450, Aralık 2021
© Telif hakkı EJOSAT'a aittir
Araştırma Makalesi

www.ejosat.com ISSN:2148-2683

European Journal of Science and Technology
Special Issue 32, pp. 445-450, December 2021
Copyright © 2021 EJOSAT
Research Article

Mitigating Data Imbalance Problem in Transformer-Based Intent Detection

Osman Büyük[1*], Mustafa Erden[2], Levent M. Arslan[2, 3]

[1*] Izmir Demokrasi University, Faculty of Engineering, Department of Electrical and Electronics, İzmir, Turkey, (ORCID: 0000-0003-1039-3234), osman.buyuk@idu.edu.tr

[2] Sestek Speech Enabled Software Technologies Inc., Department of Research and Development, İstanbul, Turkey, (ORCID: 0000-0002-2661-1200), mustafa.erden@sestek.com

[3] Bogazici University, Faculty of Engineering, Department of Electrical and Electronics, İstanbul, Turkey, (ORCID: 0000-0002-6086-8018), arslanle@boun.edu.tr

(International Conference on Design, Research and Development- 15 – 18 Aralık 2021)

(**DOI:** 10.31590/ejosat.1044812)

ATIF/REFERENCE: Büyük, O., Erden, M. & Arslan, L. M. (2021). Mitigating Data Imbalance Problem in Transformer-Based Intent Detection. *European Journal of Science and Technology*, (32), 445-450.

Abstract

There are two major problems when deploying a practical intent detection system for a new customer. First, domain-specific data from the customer could be limited and imbalanced. Additionally, despite different customers might share the same domain, their intent categories might be different from each other. Thus, it might be difficult to combine the datasets collected for different customers into a single and larger one. In this paper, we use class weights in the loss computation to alleviate the data imbalance problem. The class weights are defined inversely proportional to the frequency of the class in the training set in order to give more influence to less observed classes. We also employ a two-pass fine-tuning procedure to utilize the information in different in-domain datasets. Experimental results show that intent detection performance is improved significantly when the weighted loss function is used together with the two-pass transfer learning procedure. The absolute performance improvement in percent detection accuracy is approximately 2% over a transformer-based baseline.

Keywords: Intent Detection, Deep Learning, Transformers, Data Imbalance, Transfer Learning.

Dönüştürücü Tabanlı Niyet Tespitinde Veri Dengesizliği Etkisinin Azaltılması

Öz

Bir niyet tespiti uygulamasını yeni bir müşteri için gerçekleştirirken iki temel problem ile karşılaşılmaktadır. İlki müşteriden gelen alana özgü veri miktarının genellikle az ve her sınıftan dengesiz sayıda örnek içermesidir. Ayrıca, müşteriler benzer alanlarda bir uygulama gerçekleştirmek isteseler de, belirledikleri niyet kategorileri genellikle farklı olmaktadır. Bu durum, farklı müşteriler için toplanan verilerin tek ve daha büyük bir veri seti haline getirilmesini zorlaştırmaktadır. Bu çalışmada veri dengesizliği problemini azaltmak için kayıp fonksiyonunda sınıf ağırlıkları kullanılmıştır. Sınıf ağırlıkları, eğitim verisinde az örneği olan sınıflara daha fazla ağırlık vermek için, sınıftaki örnek sayısı ile ters orantılı olarak belirlenmiştir. Ayrıca, benzer alanlarda toplanmış veri setlerindeki bilgiden faydalanmak için iki uyarlama aşaması olan bir transfer öğrenme yöntemi denenmiştir. Deneylerde, ağırlıklı kayıp fonksiyonu ile iki aşamalı transfer öğrenme yönteminin birlikte kullanılmasının niyet tespiti sınıflandırma başarımını önemli oranda arttırdığı gözlenmiştir. Yüzde tanıma oranındaki net artış dönüştürücü tabanlı referans sisteme göre %2 olarak gerçekleşmiştir.

Anahtar Kelimeler: Niyet Tespiti, Derin Öğrenme, Dönüştürücüler, Veri Dengesizliği, Transfer Öğrenme.

[*] Corresponding Author: osman.buyuk@idu.edu.tr

1. Introduction

Intent detection is defined as the task of identifying the intent of a client from his/her text inquiry. In an intent detection task, the user text input is classified into one of the pre-defined intent categories. Due to the massive increase in internet usage, automatic intent detection systems are deployed in different sectors such as banking, retail and telecommunications. With the recent advances in intent detection systems, today automatic replies can be created for internet users when they inquire information about a product.

There are two major problems when developing an intent detection system for a new customer. First, domain-specific dataset collected for the customer is usually not adequate to train a robust model. Second, samples in the dataset are usually imbalanced, e.g., they are not equally distributed over the classes. An intent detection service provider can deploy an intent detection system for several different customers which operate in the same domain. On the other hand, the intent categories for the customers are usually different despite the domain similarity. Therefore, it is not easy to combine the different datasets into a single and larger one.

The transformer has been introduced in (Vaswani et al, 2017) and revolutionized natural language processing (NLP). The transformer entirely relies on attention mechanism to compute its output representations without the need for any recurrent or convolutional blocks. The transformer can handle long-range dependencies more efficiently with this novel architectural choice. Various general purpose language representations are trained using the transformer architecture and some of the models are made available publicly (Devlin et al, 2018; Liu et al, 2019; Radford et al, 2019; Song et al, 2020; Yang et al, 2019). These pre-trained models are adapted to the downstream task using a relatively small amount of task-specific supervised data. This pre-training and fine-tuning procedure has become a de-facto approach for many NLP tasks and achieved the best performances (Squad, 2021).

There are a few studies which investigate intent detection task for Turkish language. In (Deveci et al, 2020), term frequency - inverse document frequency (TF-IDF) features are employed for the task. In (Dündar et al, 2020), it is concluded that contextual word embeddings from transformers improves the intent detection accuracy compared to the classical machine learning models. In our previous study, we implement an intent detection system using the pre-trained transformer models (Büyük et al, 2021). In the study, we used three different intent detection datasets. The datasets are collected for finance domain but have different intent categories. We showed that the intent detection accuracy can be improved when a two-pass fine-tuning procedure is employed to utilize the information in in-domain datasets. In the first pass, the transformer parameters are updated using an in-domain dataset to the target set. Then, the adapted transformer parameters are used as the baseline model for the second pass. The second pass fine-tuning is performed with the target set.

In this study, we further improve the intent detection accuracy by using class weights in the loss computation to alleviate the data imbalance problem. The class weights are determined inversely proportional to the frequency of that class in the training set. As a result, we assign higher weights to the classes which are less represented in the dataset. Using the weighted loss function together with the transfer learning procedure in (Büyük et al, 2021) significantly improves the intent detection accuracy. We achieve approximately 2% absolute accuracy improvement in percent detection rate over a baseline model.

The remainder of the paper is organized as follows. Intent detection datasets are presented in Section 2. We provide a summary of the transformer architecture in Section 3. Section 4 is devoted to experimental results. Our paper is concluded with our key findings.

2. Datasets

2.1. Train Datasets

Statistics of the training datasets are provided in Table 1. All the datasets in the table are collected for intent detection tasks in banking domain. Banka118 and Banka120 are collected in Sestek Incorporation for Turkish. The datasets are created for conversational artificial intelligence solution of Sestek Inc. for two separate customers in finance domain. Banking77 is first presented in (Casanueva et al, 2020) and is originally in English. We use Google Translate to translate Banking77 to Turkish. We did not perform any manual corrections on the translator outputs.

As observed in Table 1, 2628 samples from 118 categories, 1694 samples from 120 categories and 10004 samples from 77 categories are included in Banka118, Banka120 and Banking77 datasets, respectively. Average numbers of words in the samples are 3.79, 4.18 and 8.08. There are 22.3 samples on average in each category with standard deviation of 26.4 in Banka118. The mean and standard deviation of the samples are 14.1 and 14.7 for Banka120 and 129.9 and 32.7 for Banking77. The number of samples in the intent categories varies from 5 to 191 in Banka118. It is from 5 to 99 in Banka120 and 35 to 187 in Banking77.

As a result, Banking77 includes longer sentences compared to Banka118 and Banka120. Additionally, Banka118 and Banka120 include more categories and fewer samples. Moreover, the sample distribution is much more imbalanced in Banka118 and Banka120. In order to visualize the data imbalance in the datasets, we provide the histogram plots of the number of samples in each intent category in Figure 1 and Figure 2 for Banka118 and Banking77, respectively. As observed in Figure 1, most of the categories contain few samples (e.g., less than 10 samples) in Banka118 while the average number of samples per category is approximately 22. On the other hand, Banking77 is relatively more balanced as seen in Figure 2. Here, we should emphasize that Banka118 and Banka120 are collected from a real-life application and thus they may provide difficult but realistic test case.

Table 1. Intent detection training and test datasets.

	Training Datasets			Test Datasets		
	Banka118	Banka120	Banking77	Banka118	Banka120	Banking77
# of intent categories	118	120	77	118	120	77
# of text samples	2628	1694	10004	580	120	3080
Mean/standard deviation of samples in intent categories	22.3 / 26.4	14.1 / 14.7	129.9 / 32.7	4.91 / 0.28	1 / 0	40 / 0
# of samples in the intent category with the most/least samples	191 / 5	99 / 5	187 / 35	5 / 4	1 / 1	40 / 40
Average number of words in the text samples	3.79	4.18	8.08	4.05	4.10	7.44

2.2. Test Datasets

We use a separate test set for each dataset in order to evaluate the classification performances. Statistics of the test sets are provided in Table 2. The test sets consist of 3080, 580 and 120 samples for Banking77, Banka118 and Banka120, respectively. As observed in Table 2, all the intent detection categories are represented in the test sets for all the datasets. Additionally, when we compare Table 1 and Table 2, we can conclude that the test sets are much more balanced compared to the training sets.

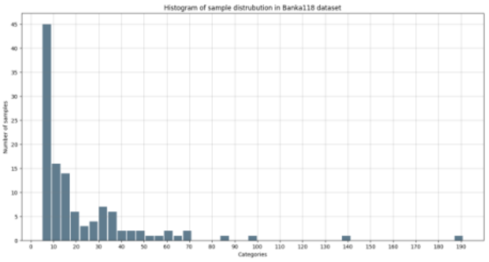

Figure 1. Histogram distribution of train samples in Banka118.

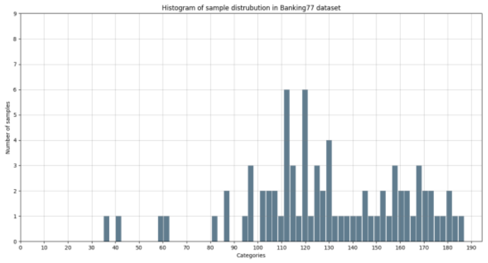

Figure 2. Histogram distribution of train samples in Banking77.

3. Methodology

The transformer has been proposed in (Vaswani et al, 2017) and become state-of-the-art method for many NLP tasks. Besides improving the performance, the transformer introduced a novel modeling architecture. Before the transformer, the dominant models for sequence to sequence (seq2seq) tasks were based on recurrent or convolutional layers. The models usually consist of an encoder block, a decoder block and an attention mechanism to connect the encoder to the decoder. On the other hand, the transformer is only relied on attention and removed the need for recurrent or convolutional layers. The transformer consists of several layers of self-attention which relate each position of the input sequence to the other positions. Due to its novel architecture, the transformer can handle long-range dependencies more effectively when compared to the other traditional neural network architectures.

Many general purpose language representations are proposed and released publicly relying on the transformer architecture. In (Devlin et al, 2018), bidirectional encoder representations from transformers (BERT) are introduced. BERT is pre-trained using next sentence prediction and masked language model (MLM) objectives. In the MLM objective, a percentage of the input tokens are replaced with a special [MASK] token before feeding them into the model. The model attempts to predict the original tokens based on the context provided by the unmasked tokens. The MLM objective enables the representation to combine the left and the right context and allows the training of a bidirectional model.

Table 3. Comparison of several different adaptation strategies. In the experiments with subscript 'Fixed', BERT is used as a fixed feature extractor. In the experiments with subscript 'Tuned', both the transformer and classifier layers are fine-tuned to the classification task.

	Banking77			Banka118	Banka120
	10	30	Full	Full	Full
BERTurk$_{\text{Fixed}}$	70.25±1.80	82.52±0.56	88.68±0.18	60.43±0.40	87.17±0.85
BERTurkIntent$_{\text{Fixed}}$	74.31±1.70	84.41±0.42	89.49±0.08	61.90±0.65	86.92±1.45
BERTurk$_{\text{Tuned}}$	79.18±0.99	88.16±0.76	**92.52±0.15**	**74.43±0.69**	93.58±1.29
BERTurkIntent$_{\text{Tuned}}$	**81.15±0.70**	**88.38±0.54**	92.50±0.30	74.14±0.82	**94.58±1.31**

The general purpose pre-trained models are fine-tuned to the downstream task using relatively small amount of labeled task-specific text. A classifier layer is added on top of the stacked transformers to fine-tune the network to the target task. The classifier is usually a multi-layer perceptron (MLP) with a few hidden layers. The transformer model can either be used as a fixed feature extractor or can be fine-tuned to the task. In the feature extractor case, the transformer parameters are fixed and only the classifier is updated. On the other hand, both the transformer and classifier parameters are updated in the latter case. In this paper, both adaptation techniques are investigated for Turkish intent detection task.

4. Experimental Setup and Results

Our experimental setup and results are provided in this section. In the experiments, we use BERTurk in https://huggingface.co/dbmdz/bert-base-turkish-cased as the baseline pre-trained transformer model since it provided the best accuracies in our previous study (Büyük et al, 2021). The model has 12 transformer layers with 12 attention heads. It is trained using the Turkish Wikipedia, Turkish OSCAR corpus and OPUS corpora. Its training corpus size is 35GB.

The baseline BERTurk model is fine-tuned to the intent detection task using the datasets in Table 1. We use all the samples in the datasets for fine-tuning except Banking77. In Banking77, we perform few-shot fine-tuning settings similar to (Casanueva et al, 2020). In the few-shot settings, only 10 or 30 samples from each intent category are used to update the baseline model parameters.

We use the percent detection accuracies as the evaluation metric. In order to alleviate the randomness in different test runs and get more reliable results, each test is repeated for 10 times. In each repetition, we use a different random seed. The fine-tuning is performed for 15 epochs and the best accuracy is recorded. In the intent detection results, we provide the mean and standard deviation of the best accuracies.

The baseline transformer model can be tuned to the intent detection task using two procedures. First, the baseline model can be adapted with the MLM objective using the intent detection datasets. Second, the baseline model can be fine-tuned to the task with an additional classifier layer as described in Section 3. The first sub-section is devoted to the comparison of the two adaptation techniques. In the second sub-section, we present results for data imbalance and transfer learning experiments.

4.1. Comparison of Adaptation Procedures

In order to observe the effects of different fine-tuning objectives, we experiment four different models. The models are abbreviated as BERTurk$_{\text{Fixed}}$, BERTurkIntent$_{\text{Fixed}}$, BERTurk$_{\text{Tuned}}$ and BERTurkIntent$_{\text{Tuned}}$.

4.1.1. BERTurk$_{\text{Fixed}}$

In this experiment, BERTurk is used as a fixed feature extractor. Its parameters are not updated during the classification task fine-tuning. Only the classifier parameters are updated. The outputs of the last transformer layer are mean pooled before feeding them into the classifier. The classifier is a MLP with one hidden layer. The number of neurons in the hidden layer is 512. We did not use any dropout. Maximum token length is 64. Batch size is set to 128. We use Adam optimizer with learning rate 0.001. The loss function is categorical cross entropy. We perform 100 epochs for fine-tuning. The number of epochs is chosen higher compared to the other experiments since an epoch is completed much faster in the 'Fixed' scenarios.

4.1.2. BERTurkIntent$_{\text{Fixed}}$

In this experiment, the BERTurk model is first adapted with the MLM objective using the intent detection datasets. For this purpose, the intent detection datasets in Table 1 are merged. We run 3 epochs of the MLM training with the combined dataset. Batch size is set to 8. We use Adam optimizer with a learning rate of $5x10^{-5}$. Then, the adapted transformer model is used as a fixed feature extractor similar to BERTurk$_{\text{Fixed}}$. The classification task fine-tuning parameters are the same with BERTurk$_{\text{Fixed}}$.

4.1.3. BERTurk$_{\text{Tuned}}$

In this experiment, both the transformer and classifier models are updated during the classification task fine-tuning. The fine-tuning is performed for 15 epochs. We choose this setting since an epoch takes much longer time in that scenario. Batch size is set to 16. We use Adam optimizer with an initial learning rate of $5x10^{-5}$ and weight decay of 0.01. The fine-tuning parameters are kept similar to BERTurk$_{\text{Fixed}}$.

4.1.4. BERTurkIntent$_{Tuned}$

In this experiment, we use the adapted model to the intent detection task using the MLM objective in BERTurkIntent$_{\text{Fixed}}$ as the baseline model. Different from the BERTurkIntent$_{\text{Fixed}}$, the baseline model is not used as a fixed feature extractor. In the intent classification fine-tuning, both the transformer and classifier parameters are updated. As a result, the intent detection training samples are used two times in this experiment, first for the MLM intent pre-training and then for the downstream task fine-tuning. The MLM pre-training and the downstream task fine-tuning parameters are the same with the BERTurkIntent$_{\text{Fixed}}$ and BERTurk$_{\text{Tuned,}}$ respectively.

Percent intent detection accuracies are presented in Table 3. As observed in the table, the MLM intent pre-training improves the performance especially if the transformer model will be used as a fixed feature extractor. This can be attributed to the fact the transformer parameters are updated only in the MLM intent pre-training stage in the ‘Fixed’ scenarios. In the ‘Tuned’ scenarios, the improvement with the MLM intent pre-training is not significant. When we compare BERTurk$_{\text{Tuned}}$ to BERTurkIntent$_{\text{Tuned}}$, we observe that the improvement is slightly higher in the few-shot settings of Banking77. In the few-shot settings, all samples of Banking77 dataset are not used for downstream task fine-tuning; only 10 or 30 samples are used for each category. Therefore, the performance improvement can be partly attributed to the fact that some of the training samples in Banking77 are only observed in the MLM intent pre-training stage. As a last observation, when the accuracies in ‘Tuned’ scenarios are compared to the corresponding ‘Fixed’ scenarios, we can conclude that updating the transformer parameters together with the classifier layer in the downstream task fine-tuning results in significant performance improvement.

4.2. Data Imbalance and Transfer Learning Experiments

From this point on, we use Banka118 since it is originally collected for Turkish and larger than Banka120. We use BERTurk$_{\text{Tuned}}$ as the reference model since it provided the best accuracy in Table 3 for Banka118. In order to alleviate the data imbalance problem in the dataset, we use a weighted loss function. The weights are determined inversely proportional to the frequency of each intent category in the training set. This experiment is abbreviated as ‘WL’ in Table 4. In the ‘TF’ experiment in Table 4, we perform the fine-tuning in two passes. In the first pass, the baseline BERTurk model is fine-tuned using Banka120 dataset. In this pass, the final layer of the classifier has 120 output classes. The adapted transformer parameters are kept for the second pass. On the other hand, the classifier is replaced with a new MLP which has 118 output classes. In the second pass, Banka118 is used for fine-tuning. In the experiment abbreviated as ‘TF-WL’, the weighted loss and the two-pass fine-tuning procedures are used together.

Table 4. Percent intent detection accuracies for weighted loss (WL) and two-pass fine-tuning (TF) methods on Banka118 dataset.

Model	% Accuracy
BERTurk$_{\text{Tuned}}$	74.43±0.69
BERTurk-WL$_{\text{Tuned}}$	74.81±1.52
BERTurk-TF$_{\text{Tuned}}$	75.50±0.57
BERTurk-TF-WL$_{\text{Tuned}}$	**76.45±0.74**

As observed in Table 4, using class weights in the loss computation slightly improves the perfromance compared to the reference BERTurk$_{\text{Tuned}}$ model. The two-pass fine-tuning procedure also results in approximately 1% absolute performance improvement. We achieve the best accuracy when the two-pass fine-tuning is used together with the weighted loss function. In the BERTurk-TF-WL$_{\text{Tuned}}$ experiment, the absolute accuracy improvement reaches 2%.

5. Conclusions

In this paper, we use pre-trained transformer model for Turkish intent detection task in banking domain. In order to alleviate the data imbalance problem, we use class weights in the loss computation. The weights are determined inversely proportional to the frequency of that class in the training set. We also employ a two-pass fine-tuning strategy to leverage the information in similar intent detection datasets. In the experiments, we observed that the intent detection accuracy is significantly improved when the weighted loss function and two-pass fine-tuning methods are employed together. Absolute improvement in percent intent detection rate is more than 2% over a baseline model.

6. Acknowledge

This work was supported by The Scientific and Technological Research Council of Turkey (TUBITAK) under the project number 3189149.

References

Büyük, O., Erden, M. and Arslan, L. M. (2021). "Leveraging the information in in-domain datasets for transformer-based intent detection," Innovations in Intelligent Systems and Applications Conference (ASYU 2021), 2021, pp. 1-4, doi: 10.1109/ASYU52992.2021.9599055.

Casanueva, I., Temčinas, T., Gerz, D., Henderson, M., Vulić, I. (2020). “Efficient intent detection with dual sentence encoders,” arXiv preprint, arXiv:2003.04807.

Deveci, C., Demirbağ, S., Erden, M., Arslan, L.M. (2020) “Query Intent Classification with Short Sentences in Agglutinative Languages,” IEEE 28th Signal Processing and Communications Applications Conference (SIU 2020), Gaziantep, Turkey.

Devlin, J., Chang, M. W., Lee, K., Toutanova, K. (2018) “BERT: Pre-training of deep bidirectional transformers for language understanding,” arXiv preprint, arXiv:1810.04805.

Dündar, E.B., Kiliç, O.F., Çekiç, T., Manav, Y., Deniz, O. (2020) "Large scale intent detection in Turkish short sentences with contextual word embeddings," 12th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (KDIR 2020), pp. 187-192.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V. (2019). "Roberta: A robustly optimized bert pretraining approach," arXiv preprint, arXiv:1907.11692.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I. (2019). "Language models are unsupervised multitask learners," OpenAI blog, 1(8), 9.

Squad, SQuAD2.0 The Stanford Question Answering Dataset (2021), https://rajpurkar.github.io/SQuAD-explorer/.

Song, K., Tan, X., Qin, T., Lu, J., Liu, T.Y. (2020). "MPnet: Masked and permuted pre-training for language understanding," arXiv preprint, arXiv:2004.09297.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., Polosukhin, I. (2017). "Attention is all you need," arXiv preprint, arXiv:1706.03762.

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R., Le, Q.V. (2019). "XLnet: Generalized autoregressive pretraining for language understanding," arXiv preprint, arXiv:1906.08237, 2019.