Toward Federated Learning Through Intent Detection Research

Daiga Deksne[1,2][0000−0002−8916−0320], Jurgita Kapočiūtė-Dzikienė[1,3][0000−0002−8402−4549], and Raivis Skadiņš[1,2][0000−0003−0929−2380]

[1] Tilde, Vienības gatve 75A, Riga, Latvia
{daiga.deksne,raivis.skadins}@tilde.com, jurgita.dzikiene@tilde.lt
[2] Faculty of Computing, University of Latvia, Raiņa bulv. 19, Riga, Latvia
[3] Faculty of Informatics, Vytautas Magnus University, Universiteto str. 10, Kaunas, Lithuania

Abstract. Virtual assistants are crucial in modern organizational communication, requiring robust Natural Language Understanding (NLU) models to ensure effective interaction. This research addresses the challenges of NLU model development specifically for English, German, Spanish, French, Italian, Latvian, and Estonian languages. By delving into intent detection methodologies, encompassing memory-based techniques such as vectorization using LaBSE, ADA, or SONAR models, along with similarity or distance-based approaches; supervised text classification (utilizing LaBSE + FFNN or fine-tuning LaBSE); and text generation (leveraging OpenAI's Davinci model) techniques, we emphasize the effectiveness of memory-based approaches, particularly for non-English languages. We demonstrate the efficacy of multilingual and cross-lingual LaBSE vectorization and the SONAR large language model. This paper also introduces open-source intent detection software for federated learning. A prototype demonstrates the integration of this framework into RASA-based virtual assistants, also providing practical guidance for organizations seeking to deploy intelligent and privacy-preserving conversational agents. This research contributes to advancing virtual assistant development and underscores the potential of federated learning in enhancing NLU models.

Keywords: Intent detection · Memory-based, supervised classification and generation approaches · English, German, Spanish, French, Italian, Latvian and Estonian languages · federated learning.

1 Introduction

In today's digital era, virtual assistants are crucial in facilitating seamless communication between organizations and stakeholders, offering support around the clock, and improving operational efficiency.

However, developing reliable and adaptable virtual assistants presents challenges, particularly in Natural Language Understanding (NLU). Accurately understanding user intents and providing suitable responses require sophisticated NLU models capable of interpreting various linguistic inputs.

This project aims to tackle the intricacies of NLU model development for virtual assistants, while also exploring the compatibility of applying federated learning approaches to address concerns like data privacy.

Our goal is twofold: 1) to build customized bots for individual organizations and 2) to create a unified bot that serves the collective needs of all participating entities. This approach ensures tailored bot development to meet specific organizational requirements while maintaining a consistent user experience overall. By combining embedding models, intent detection methods, and federated learning frameworks, we aim to enhance virtual assistant development outcomes.

In this paper, we present our project's methodology, experiments, results, and conclusions, exploring the potential of federated learning in NLU model development and its implications for the future of virtual assistants. Additionally, we introduce open-source intent detection software based on federated learning, designed to preserve data privacy. We have also developed a prototype demonstrating the integration of this framework into RASA-based virtual assistants, including popular platforms like Bürokratt. This prototype validates our approach and provides practical guidance for integrating federated learning into existing virtual assistant systems, enabling organizations to deploy more intelligent and privacy-preserving conversational chatbots.

2 Related work

This section provides a comprehensive overview of research on federated learning, including its tasks, algorithms, challenges, and commonly used datasets. Special attention is given to its application in intent detection, specifically for addressing text classification problems.

The main reasons federated learning is utilized in machine learning include:

- **Efficiency and Scalability**: by distributing computation to local devices, federated learning reduces data transmission costs and alleviates the load on central servers. This allows for scalable model training on large-scale distributed datasets. Despite advancements in hardware that have addressed historical computing challenges, the efficiency of training remains crucial. This is particularly evident in Large Language Model (LLM) training, where federated learning frameworks such as FATE-LLM and FedML reflect evolving machine learning practices ([4], [8], [2], [21]).
- **Customization and Edge Computing**: federated learning enables personalized model training by customizing centrally pre-trained models to individual user preferences while safeguarding data privacy. It also facilitates edge computing by enabling model fine-tuning directly on edge devices such as mobile and IoT devices. This approach reduces latency and enhances real-time inference capabilities.
- **Privacy Preservation, Data Sovereignty, Security, and Regulatory Compliance**: federated learning facilitates model training on decentralized

data sources, ensuring privacy by eliminating the need to share raw data. This approach allows organizations to maintain control over their data, comply with regulations, and minimize the risk of data breaches or unauthorized access. Since data remains decentralized, it enhances security and ensures regulatory compliance.

This research primarily focuses on the privacy, security, and data confidentiality aspects of federated learning, with less emphasis on large-scale data distribution or continuous bidirectional parameter synchronization methods. Federated learning finds applications across diverse domains in machine learning, including Image, Text, and Speech Processing, Healthcare, Finance, and more. However, this research specifically focuses on intent detection within NLP, which is not the most conventional application of federated learning in this field. The majority of research in NLP tends to focus on text classification, clustering, sentiment analysis, and recommendations. Classical datasets commonly used in these areas include the 20 newsgroup dataset [12], Reuters-21578 Text Categorization Collection [14], AG News [25], IMDb Reviews [17], Amazon Reviews [18], TREC [22], DBpedia [13], and others.

[16] offer a comprehensive exploration of federated learning within the realm of NLP. They argue that mounting concerns and regulatory frameworks surrounding data privacy and sparsity underscore the necessity for privacy-preserving, decentralized learning methods in NLP tasks. Federated learning emerges as a promising approach, enabling numerous clients such as personal devices or organizations to collaboratively learn a shared global model, thereby benefiting all participants while allowing individual users to maintain control over their data locally. The authors note a lack of systematic comparison and analysis in the existing literature and introduce FedNLP, a benchmarking framework designed to evaluate federated learning methods across four distinct tasks, including text classification, which bears relevance to intent detection.

Classical algorithms in federated learning aim to facilitate collaborative model training across distributed clients while preserving data privacy. FedAvg [19] serves as the foundational method, with clients and servers using SGD optimizer for model weight updates. FedProx [15] enhances training stability by constraining local updates closer to the global model through L2 regularization, addressing statistical heterogeneity. FedOPT [20] extends FedAvg by introducing federated versions of adaptive optimizers like Adagrad, Adam, and Yogi, representing crucial advancements for scalable and privacy-preserving model training across decentralized networks.

We analyzed various federated learning frameworks to explore their architectures, features, and potential applications in our project. The SEFT framework [23] emphasizes efficiency, scalability, and robustness with one central and multiple client nodes, leveraging efficient cryptography to handle client dropouts effectively. FLUTE (Federated Learning Utilities for Testing and Experimentation) [6] offers high-performance federated learning simulations with features for large-scale experiments, supporting millions of clients, single and multi-GPU setups, and multi-node orchestration, along with options for local or global differential privacy and model quantization techniques [24]. FEDML [8] is another ML library for large-scale distributed training and federated learning, enhanced by FEDML Launch, a cross-cloud scheduler enabling AI job execution on any GPU cloud or on-premise cluster.

3 Solution Architecture

This project aims to develop sophisticated bots tailored to the unique needs of multiple organizations. We create independent bots for each organization while also building a unified bot for collective use, ensuring a seamless user experience.

The rationale for a unified bot is that end users often don't distinguish between different bots they interact with.

Our solution architecture (Figure 1) includes remote bot training sites where trainers develop their bots independently. These trainers manage private training data and train local models specific to their bots.

Additionally, a central training site plays a crucial role, in training a single NLU model using a federated approach. This central model aggregates parameters from remote sites, creating a unified NLU model capable of identifying intents from shared training data and remote sites. This approach ensures a versatile bot system meeting diverse organizational needs.

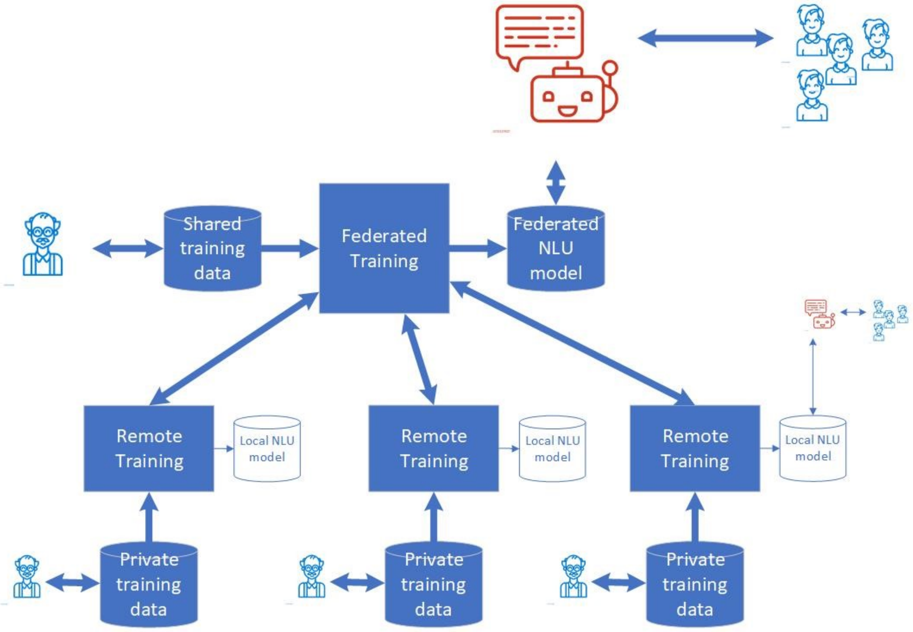

Fig. 1. System overview

4 Intent Detection

The NLU module in virtual assistants may employ either intent-detection-based or generative approaches. The former typically offers higher accuracy, requiring less training data, which aligns better with customer expectations. Hence, our problem-solving approach is formulated as intent detection, specifically, supervised text classification.

4.1 Dataset

The intent detection research aimed to test the effectiveness of various text classification approaches for different languages and compare their results at the same time checking the possibilities of integrating other languages into the model in the future. Consequently, the experiments were performed with two datasets:

- *Multi-language segregated dataset* [4] (for the statistics see Table 1) containing several languages (English, German, French, Italian, Spanish, and Latvian). It contains 37 intents (classes) and is structured to ensure that each text instance is associated with only one class, assuring the nature of single-label classification. The texts were shuffled and split into training (80%) and testing (20%) subsets within each language. The dataset is well-balanced resulting in low majority (1) and random baselines (2).
- *Estonian datasets* containing purely Estonian texts. These three datasets (for the statistics see Table 2) [5] were constructed using data provided by RIA from the Bürokratt project.

$$majority_{baseline} = max(P_i) \tag{1}$$

, where P_i is the probability of the class.

$$random_{baseline} = \sum (P_i)^2 \tag{2}$$

4.2 Approaches

Within the frame of this project, we have tested the following text classification methods:

[4] The dataset has been created in the StairwAI project (`https://stairwai.nws.cs.unibo.it/`) funded by the European Union's Horizon 2020 research and innovation programme under grant agreement 101017142.

[5] All Estonian datasets have been made publicly accessible on `https://github.com/tilde-nlp/fnlu/tree/main/Other` assuring transparency and their usage by others in the future.

Table 1. Statistics of the multi-language segregated dataset.

Language	Examples in training split	Examples in testing split	Majority baseline	Random baseline
English	386	94	0.074	0.033
German	192	47	0.064	0.032
Spanish	193	47	0.064	0.032
French	193	47	0.064	0.032
Italian	193	47	0.064	0.032
Latvian	183	46	0.065	0.031

Table 2. Statistics of the Estonian datasets.

Dataset	Number of intents	Number of examples	Majority baseline	Random baseline
Rahvusraamatukogu (National Library)	36	1 104	0.130	0.053
Sotsiaalkindlustusamet (Social Insurance)	7	79	0.418	0.235
Kriisijuhtimine (Crisis Management)	23	287	0.105	0.054

- **FastText+CNN**. This supervised text classification approach combines FastText embeddings [9] with a customized CNN architecture [11]. FastText utilizes subword embedding information, enabling it to construct word vectors even in cases of out-of-vocabulary words or typos. CNN, on the other hand, focuses on identifying token n-grams, which is particularly useful for text classification tasks reliant on keywords rather than the entire contextual meaning of the text. In our experiments, we have used the proprietary FastText embedding model supporting English and Latvian languages together with the CNN's architecture presented in [10]. This method will exclusively serve as the baseline.
- **LaBSE+FFNN**. We utilized the frozen Language Agnostic BERT Sentence Embedding (LaBSE) approach proposed by [5], alongside a two-layer fully connected (FFNN) model optimized and fine-tuned for our intent detection tasks. LaBSE differs from traditional BERT embeddings by providing sentence-level representations, capturing the semantics of entire texts simultaneously. It supports 109 languages, including Estonian, generating fixed-size vectors for texts without retaining word boundaries. LaBSE is not sensitive to word order, therefore it is well-suited for languages with flexible sentence structures, requiring less training data for various variations. However, LaBSE's cross-lingual capability may vary across languages.
- **LaBSE-LangChain-k1** (which is the memory-based approach) leverages the LangChain framework, enabling context-aware applications without the need for training. Once training instances are vectorized and stored in the Chroma vector database, the method employs cosine similarity and a greedy

search to find the closest training instance to the testing one, thereby assigning its label. LaBSE serves as the vectorization model.

- **LaBSE-LangChain-k10-mv**. This method closely resembles LaBSE-LangChain-k1, but instead of searching for a single similar instance, it searches for the 10 closest instances, collects their class labels, and conducts a majority vote to determine the final class.
- **ADA-LangChain-k1**. This approach resembles LaBSE-LangChain-k1, but instead of LaBSE, OpenAI's text-embedding-ada-002 embeddings [7] are used instead.
- **ADA-LangChain-k10-mv**. This approach is similar to LaBSE-LangChain-k10-mv, but instead of LaBSE, it uses text-embedding-ada-002 embeddings.
- **Davinci-fine-tuning**. For this approach, we used OpenAI's davinci-002 model [1]. This model is a generative transformer model, but we adjusted it to generate only one first token as the input text's label. In our experiments, we fine-tuned the added layers' parameters while keeping the hyperparameter values at their defaults.
- **SONAR-LangChain-k1** adopts a similar approach to LaBSE-LangChain-k1 but: 1) employs the SONAR vectorization model [3] developed by META research instead of LaBSE; 2) applies the Euclidean distance measure metric instead of cosine similarity; and 3) uses the Faiss database instead of Chroma. SONAR is an open-source large language model that supports all our target languages and was selected to expand the scope of experimentation as it is lately receiving much attention in many text classification tasks for various languages.

4.3 Experiments and Results

The initial experiments utilized a multi-language segregated dataset. For methods involving training, the dataset was shuffled and divided into training (80%) and validation (20%) sets. Accuracy was chosen as the primary evaluation metric.

Approaches with randomized parameter initialization were tested five times, with averaged results and confidence intervals calculated. This protocol didn't apply to FastText+CNN, the baseline supporting only two languages, or Davinci-fine-tuning, which stores models on third-party servers incurring charges. Davinci-fine-tuning was solely for comparison. Results with the multi-language segregated dataset are summarized in Fig. 2.

The English dataset performed best with Davinci-fine-tuning, followed by LaBSE-LangChain-k1 and ADA-LangChain-k1. However, LaBSE-fine-tuning was most effective for other languages (German, Spanish, French, Italian, and Latvian). Despite its superior performance, LaBSE-fine-tuning is impractical due to lengthy training times (up to 1 hour on our small dataset) and high hardware demands (exceeding 12 GiB of GPU RAM). Considering shorter training times, LaBSE-LangChain-k1 emerges as the next best option, delivering good results for non-English languages (except German). ADA and Davinci models outperform

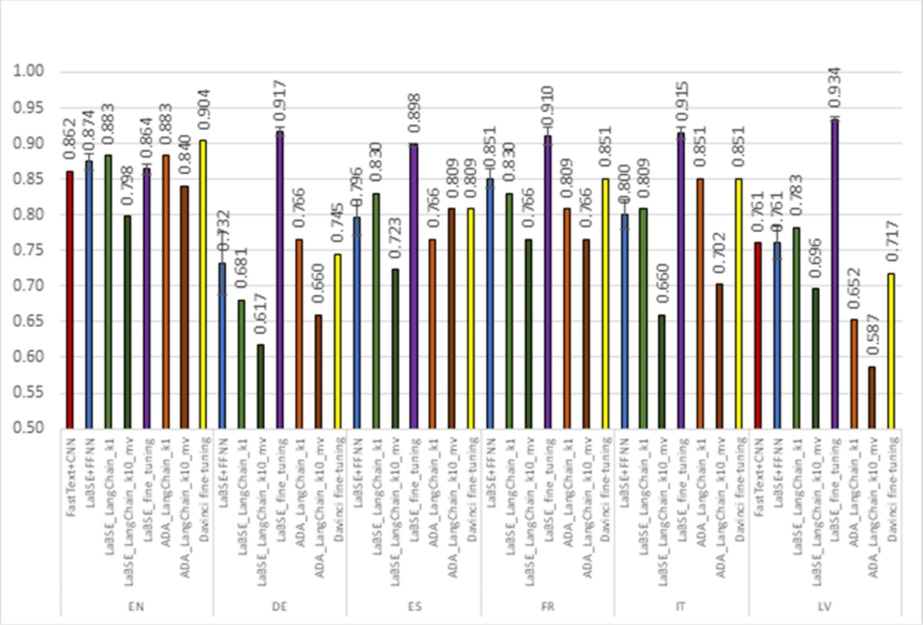

Fig. 2. The accuracy values with approaches and languages on the multi-language segregated dataset.

LaBSE for German, likely due to better representation. Additionally, precision, recall, MicroF1, and MacroF1 metrics were evaluated. Results in Table 3 show that SONAR-LangChain-k1 slightly outperforms LaBSE-LangChain-k1. Both LaBSE and SONAR have their advantages and drawbacks in terms of speed and resource usage. LaBSE is faster, completing requests in 27-30 milliseconds compared to SONAR's 120-130 milliseconds, but SONAR consumes less disk space at 2.85 GiB, while LaBSE requires 5.27 GiB.

4.4 Discussion

The experiments with a multi-language segregated dataset demonstrate that all tested approaches are suitable for our problem solving, as achieved accuracies significantly outperform random and majority baselines. Davinci-fine-tuning is the most accurate technique for the well-supported English language; whereas, LaBSE-fine-tuning (with unfrozen parameters and additional layers) is the best technique for all other tested non-English languages (German, Spanish, French, Italian, and Latvian). It is not surprising, as this model is not only multilingual (supporting all our target languages) but also cross-lingual (able to benefit from other languages in this model). Despite significantly good performance, LaBSE-fine-tuning is less practical compared to the second-best LaBSE-LangChain-k1, which requires much less training time and GPU computational resources.

Table 3. Evaluation results on Estonian datasets

Dataset	**LaBSE-LangChain-k1**	**SONAR-LangChain-k1**
Rahvusraamatukogu (National Library) 36 intents	accuracy: 0.760 precision: 0.760 recall: 0.760 MicroF1: 0.760 MacroF1: 0.726	accuracy: 0.763 precision: 0.763 recall: 0.763 MicroF1: 0.763 MacroF1: 0.758
Sotsiaalkindlustusamet (Social Insurance) 7 intents	accuracy: 0.608 precision: 0.608 recall: 0608 MicroF1: 0.608 MacroF1: 0.589	accuracy: 0.709 precision: 0.709 recall: 0.709 MicroF1: 0.709 MacroF1: 0.638
Kriisijuhtimine (Crisis management) 23 intents	accuracy: 0.5 precision: 0.5 recall: 0.5 MicroF1: 0.5 MacroF1: 0.484	accuracy: 0.503 precision: 0.503 recall: 0.503 MicroF1: 0.503 MacroF1: 0.479

The experiments with the Estonian datasets claim the slight superiority of the SONAR-LangChain-k1 approach over LaBSE-LangChain-k1, which is not surprising: the SONAR model is the multilingual and innovative large language model, recently demonstrating very good performance on various text classification tasks for different languages. This research allows us to confirm that it is also suitable for the Estonian language. However, the Estonian datasets used in our experiments are rather small, which complicates the evaluation if differences between the results are statistically significant. Nonetheless, this dataset is in line with our customers' usual expectations, and as this research demonstrated, it delivers favorable outcomes even when working with very limited data.

The efficacy of memory-based approaches presents additional opportunities for their flexible integration within the framework of federated learning. Notably, the federated learning approach utilized offers flexibility, as it eliminates the need for a predefined set of intents during central model training. Each client has the autonomy to introduce new intents, modify existing ones, recompile local models, and update the central model, thus addressing parameter aggregation issues commonly observed in standard federated learning algorithms. Moreover, the system can accommodate variations in the number of training samples and intents among users, and deliberately incorrect data does not compromise the integrity of the central model, as intent records in the index operate independently of one another.

5 Solution Implementation

Our research in intent detection has identified the most effective methodology, which now needs to be seamlessly integrated into the federated learning framework. The customer raw data remains secured as only vectorized representations leave remote training sites and are used for model parameter computation.

Our federated intent detection solution is implemented as open-source software, accessible on GitHub [6] under the Apache 2.0 license. It includes two components: a federated intent detector used for both training and real-time inference, and vectorization services offered through Docker containers with options for LaBSE and SONAR embedding models.

The prototype showcases an example implementation integrating federated learning into Rasa [7] bot software, with setup instructions for configuring and deploying the federated NLU system in distributed environments. This approach can be adapted for integration into other Rasa-based software products like Bürokratt.

Additionally, there are also setup instructions detailing how to set up the federated NLU system with one central site and multiple remote sites. These instructions serve as a comprehensive guide for configuring and deploying the system in a distributed environment.

6 Conclusions

The pursuit of developing sophisticated virtual assistants capable of meeting the diverse needs of multiple organizations has led us to explore innovative solutions in NLU. Our project aims to build independent bots customized for each organization while also facilitating the future creation of a unified bot serving all entities within the framework of federated learning.

In this study, we explored different intent detection methodologies, encompassing memory-based, supervised text classification, and text generation. Our results indicate that memory-based techniques show notable effectiveness, especially for non-English languages, which is in line with our project's focus on the Estonian language. Moreover, the superior performance of memory-based approaches suggests the potential for fluent implementation of federated learning.

Furthermore, it introduces open-source intent detection software based on federated learning, improving accuracy while safeguarding data privacy and security. A comprehensive prototype has been developed to showcase the integration of this framework into RASA-based virtual assistants, including platforms like Bürokratt.

Acknowledgments. This research has been supported by "Eesti keeletehnoloogia 2018–2027" project: EKTB78 Liitõppe rakendamise võimalused dialoogiandmete põhjal.

[6] https://github.com/tilde-nlp/fnlu

[7] https://www.kratid.ee/en/burokratt

References

1. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J.D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al.: Language models are few-shot learners. Advances in neural information processing systems **33**, 1877–1901 (2020)
2. Chen, C., Feng, X., Zhou, J., Yin, J., Zheng, X.: Federated large language model: A position paper (2023)
3. Duquenne, P.A., Schwenk, H., Sagot, B.: Sonar: Sentence-level multimodal and language-agnostic representations (2023)
4. Fan, T., Kang, Y., Ma, G., Chen, W., Wei, W., Fan, L., Yang, Q.: Fate-llm: A industrial grade federated learning framework for large language models (2023)
5. Feng, F., Yang, Y., Cer, D., Arivazhagan, N., Wang, W.: Language-agnostic BERT sentence embedding. In: Muresan, S., Nakov, P., Villavicencio, A. (eds.) Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 878–891. Association for Computational Linguistics, Dublin, Ireland (May 2022). https://doi.org/10.18653/v1/2022.acl-long.62, https://aclanthology.org/2022.acl-long.62
6. Garcia, M.H., Manoel, A., Diaz, D.M., Mireshghallah, F., Sim, R., Dimitriadis, D.: Flute: A scalable, extensible framework for high-performance federated learning simulations (2022)
7. Greene, R., Sanders, T., Weng, L., Neelakantan, A.: New and improved embedding model. Available at https://openai.com/blog/new-and-improved-embedding-model, last accessed 2024/02/13 (December 2022)
8. He, C., Li, S., So, J., Zeng, X., Zhang, M., Wang, H., Wang, X., Vepakomma, P., Singh, A., Qiu, H., Zhu, X., Wang, J., Shen, L., Zhao, P., Kang, Y., Liu, Y., Raskar, R., Yang, Q., Annavaram, M., Avestimehr, S.: Fedml: A research library and benchmark for federated machine learning (2020)
9. Joulin, A., Grave, E., Bojanowski, P., Mikolov, T.: Bag of tricks for efficient text classification. In: Lapata, M., Blunsom, P., Koller, A. (eds.) Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers. pp. 427–431. Association for Computational Linguistics, Valencia, Spain (Apr 2017), https://aclanthology.org/E17-2068
10. Kapočiūtė-Dzikienė, J., Balodis, K., Skadiņš, R.: Intent detection problem solving via automatic dnn hyperparameter optimization. Applied Sciences **10**(21), 7426 (2020)
11. Kim, Y.: Convolutional neural networks for sentence classification. In: Moschitti, A., Pang, B., Daelemans, W. (eds.) Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 1746–1751. Association for Computational Linguistics, Doha, Qatar (Oct 2014). https://doi.org/10.3115/v1/D14-1181, https://aclanthology.org/D14-1181
12. Lang, K.: Newsweeder: Learning to filter netnews. In: Machine learning proceedings 1995, pp. 331–339. Elsevier (1995)
13. Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P.N., Hellmann, S., Morsey, M., Van Kleef, P., Auer, S., et al.: Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. Semantic web **6**(2), 167–195 (2015)
14. Lewis, D.: Reuters-21578 text categorization test collection. Distribution 1.0, AT&T Labs-Research (1997)

15. Li, T., Sahu, A.K., Zaheer, M., Sanjabi, M., Talwalkar, A., Smith, V.: Federated optimization in heterogeneous networks. Proceedings of Machine learning and systems **2**, 429–450 (2020)
16. Lin, B.Y., He, C., Ze, Z., Wang, H., Hua, Y., Dupuy, C., Gupta, R., Soltanolkotabi, M., Ren, X., Avestimehr, S.: FedNLP: Benchmarking federated learning methods for natural language processing tasks. In: Carpuat, M., de Marneffe, M.C., Meza Ruiz, I.V. (eds.) Findings of the Association for Computational Linguistics: NAACL 2022. pp. 157–175. Association for Computational Linguistics, Seattle, United States (Jul 2022). `https://doi.org/10.18653/v1/2022.findings-naacl.13`, `https://aclanthology.org/2022.findings-naacl.13`
17. Maas, A.L., Daly, R.E., Pham, P.T., Huang, D., Ng, A.Y., Potts, C.: Learning word vectors for sentiment analysis. In: Lin, D., Matsumoto, Y., Mihalcea, R. (eds.) Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. pp. 142–150. Association for Computational Linguistics, Portland, Oregon, USA (Jun 2011), `https://aclanthology.org/P11-1015`
18. McAuley, J., Leskovec, J.: Hidden factors and hidden topics: understanding rating dimensions with review text. In: Proceedings of the 7th ACM conference on Recommender systems. pp. 165–172 (2013)
19. McMahan, B., Moore, E., Ramage, D., Hampson, S., y Arcas, B.A.: Communication-efficient learning of deep networks from decentralized data. In: Artificial intelligence and statistics. pp. 1273–1282. PMLR (2017)
20. Reddi, S., Charles, Z., Zaheer, M., Garrett, Z., Rush, K., Konečnỳ, J., Kumar, S., McMahan, H.B.: Adaptive federated optimization. arXiv preprint arXiv:2003.00295 (2020)
21. Roth, H., Xu, Z., Renduchintala, A.: Adapting llms to downstream tasks using federated learning on distributed datasets. nvidia technical blog. Available at `https://developer.nvidia.com/blog/adapting-llms-to-downstream-tasks-using-federated-learning-on-distributed-datasets/`, last accessed 2024/02/13 (July 2023)
22. Voorhees, E.M., Harman, D.K., et al.: TREC: Experiment and evaluation in information retrieval, vol. 63. MIT press Cambridge (2005)
23. Wang, C., Deng, J., Meng, X., Wang, Y., Li, J., Lin, S., Han, S., Miao, F., Rajasekaran, S., Ding, C.: A secure and efficient federated learning framework for nlp. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. pp. 7676–7682 (2021)
24. Wu, X., Xu, L., Zhu, L.: Local differential privacy-based federated learning under personalized settings. Applied Sciences **13**(7), 4168 (2023)
25. Zhang, X., LeCun, Y.: Text understanding from scratch (2015)