NER for Albanian Language: A Manually Annotated Corpus and Machine Learning Models

Nelda Kote[1], Klea Kalliri[1], Kejsi Kalliri[1], Alba Haveriku[1], Besjana Muraku[1], Elinda Kajo Meçe[1]

[1] Polytechnic University of Tirana, “Mother Tereza” Square, Tirana, Albania
{nkote, klea.kalliri, kejsi.kalliri, alba.haveriku, besjana.muraku, ekajo}@fti.edu.al

Abstract. Recent advancements in artificial intelligence (AI) have significantly enhanced tasks like named entity recognition (NER), enabling the identification of people, organizations, products, events, places, and dates. This paper introduces an Albanian NER corpus with 1,003,836 tokens (56,595 sentences), including 89,850 labelled tokens, annotated with 10 NER tags. The corpus, sourced from well-known Albanian news platforms, are used to train and evaluate 10 models using algorithms such as Naïve Bayes, Logistic Regression, SVM, Random Forest, Gradient Boosting, Extreme Gradient Boosting, and Multi-Layer Perceptron variants. Among these, Extra Trees and Random Forest achieved the best results, with approximately 96% accuracy and a 95% F1 score. As the largest NER corpus in Albanian, this resource advances linguistic research and AI applications, enhancing NER tasks and advance natural language processing (NLP) developments for the Albanian language.

1 Introduction

Named Entity Recognition (NER) is the process of identifying and classifying segments of information in text into predefined categories such as persons, organizations, locations, dates, and more. NER functions as a form of sequence tagging, where to each word in a text is assigned an entity label. It plays a critical role in enhancing applications like question-answering, information retrieval, and knowledge extraction by enabling systems to accurately recognize and extract entities from text, facilitating more effective and relevant outputs [1].

In this paper, we present the largest NER corpus for Albanian language, annotated by two native speakers. The text of the corpus was collected through web scraping articles from three well-known Albanian news portals. With a total of 1,003,836 tokens (56,595 sentences), including 89,850 labeled tokens, annotated with ten NER tags, this corpus is a valuable resource for advancing NER in Albanian language. We trained and evaluated ten different models, incorporating both traditional machine learning algorithms such as Support Vector Machine (SVM), Naive Bayes (NB), Random Forest (RF), Extra Trees (ET), Gradient Boosting (GB), Extreme Gradient Boosting (XGBoost) and three Multi-Layer Perceptron neural network.

The rest of the paper is organized as follows: Section 2 discusses related works, Section 3 outlines the Albanian NER corpus, Section 4 introduces the selected machine learning algorithms, Section 5 presents the experiments and their results, and Section 6 concludes the paper and discuss directions for future work.

2 Background and related works

Research in the field of Natural Language Processing (NLP) for the Albanian language has emerged relatively recently, with ongoing efforts to develop intelligent systems across various domains. In papers [2, 3], the authors present annotated corpora for morphology and syntax, alongside neural models trained on these corpora. Notable advancements in opinion mining and sentiment analysis have been reported in papers [4, 5, 6, 7], with paper [8] leveraging sentiment analysis techniques and feature engineering to analyze sentiment polarity in Albanian fake news. Furthermore, the authors in [9, 10] introduced a question classification dataset and evaluated the performance of various machine learning and deep learning models.

Research on NER for the Albanian language is still limited, though there have been some efforts to develop resources through both manual and automated methods. The most recent NER corpus for Albanian, as outlined in [11], includes 900 sentences sourced from Albanian Wikipedia, manually annotated according to the CoNLL-2003 standard. This study evaluates the performance of language models like BERT and RoBERTa, concluding that multilingual pre-trained models—particularly those fine-tuned on Albanian-specific data—achieve the best results. The paper [12] introduced the first NER model for Albanian, using a maximum entropy approach with Apache OpenNLP. The authors manually annotated a corpus of historical and political texts, which was then used to train the models. The study in [13] proposes a deep learning-based approach for NER in Albanian, using a manually created corpus of 133,000 words. The model based on LSTM cells combined with a Conditional Random Field (CRFs) for output, shows promising results despite the limited resources available for the Albanian language. In paper [14], a NER approach for Albanian using CRFs is proposed. The authors developed a manually annotated corpus from Albanian news articles published in 2015 and 2016, which was subsequently used to train the NER model. In [15], the first automatically generated NER corpus for the Albanian language was created using news articles from Albanian media. The entities were automatically tagged using a gazetteer derived from the Albanian Wikipedia. The authors highlight that this corpus can serve as a baseline or a valuable training resource, especially in the absence of human-annotated corpora.

These efforts represent significant advancements in developing NER applications for Albanian using both traditional and deep learning methods, demonstrating the potential for further progress and promising improvements for under-resourced languages like Albanian.

3 Albanian NER corpus

In this section, we present the annotated Albanian NER corpus, including the NER annotation schema and key statistics. This corpus is the largest NER resource proposed for the Albanian language, providing a comprehensive corpus for entity recognition tasks.

3.1 Text selection

The texts in the corpus are collected through web scraping from three well-known Albanian online news platforms: *Agjencia Telegrafike Shqiptare (Albanian Telegraphic Agency)*[1], *Radio Televizioni Shqiptar(Albanian Radio and Television)*[2], and *Koha Jonë (Our Time)*[3]. Initially, the URLs for each news article were gathered, and then, using a script, the titles and corresponding texts were retrieved and saved into text files. The corpus includes news articles published between 2020 and 2024, with the distribution of articles from each source presented in Table 1.

Table 1. The number of news articles collected from each online media.

Online news platform	No. of articles
Radio Televizioni Shqiptar/ Albanian Telegraphic Agency	4,901
Agjencia Telegrafike Shqiptare/Albanian Radio and Television	580
Koha Jonë/ *Our Time*	318
TOTAL	**5,799**

The collected texts cover topics including culture, technology, social issues, international affairs, and politics. This diversity ensures a comprehensive representation of the language across various domains, enabling the machine learning models to encounter a wide variety of terms and phrases. Since expressions can have different meanings depending on context, this corpus helps the model to learn to disambiguate these meanings. Including a variety of topics is essential for offering diverse examples of how entities are used and referenced and enhancing the quality of the training data. The use of multiple contexts also enables the model to distinguish between similar entities, thereby minimizing errors.

Table 2 presents the list of topics along with the number of articles collected for each.

Table 2. The number of news articles per topic.

Topic of article	No. of articles	No. of sentences	No. of tokens
Culture	1,248	15,015	316,302
Social	3,629	30,695	452,429
Politic	580	5,888	22,794
International affairs	318	4,639	79,893
Technology	24	358	132,419
TOTAL	**5,799**	**56,595**	**1,003,836**

1 https://ata.gov.al/
2 https://rtsh.al/
3 https://kohajone.com/

3.2 NER annotation schema

This section presents the NER tagset used to annotate the collected texts in the Albanian language. Table 3 shows the list of labels along with their corresponding entities annotated using this tagset. The inclusion of a wide range of entity types enriches our understanding of the language's structure and usage, while also enhancing the model's ability to recognize and interpret named entities.

Table 3. The NER tag-set used to annotate the texts in Albanian language.

Labels	Category	Explanation
PER	Person	name and surname, only name, only surname initials
ORG	Organization	political parties, government structures, departments, federations, forums, agencies, authorities, associations, foundations, etc.
PRO	Product	vaccines, objects, etc.
EVENT	Event	conferences, summits, conventions, congresses, championships, etc.
VEND_0	City	names of cities
VEND_1	Country	names of countries
SHESH	Square	names of squares
RRUGE	Street	names of streets
DATE_0	Date	full dates, only year, only month and year, only day and month
DATE_1	Days of the week	days of the week

3.3 Annotated corpus

The corpus was manually annotated by two native Albanian speakers. We utilized Label Studio [16], an open-source online data labeling platform, to facilitate the annotation process. Fig. 1 shows the environment used for annotation.

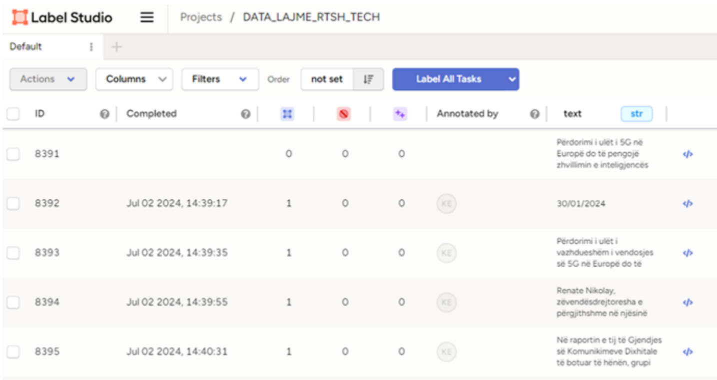

Fig. 1. The Label Studio environment used for annotation of the corpus.

The annotated corpus is formed according to the CONLL-2003 standard [17], which is widely used for training and evaluating NER systems. This format provides standardized labels, streamlining the identification and classification of named entities across various texts. A key feature of the CONLL-2003 format is that sentences within a document are separated by an empty line, clearly delineating sentence boundaries. This structured format operates at the word (token) level, with each line representing a single word. Such an approach is essential for NER tasks, as it allows for the classification of each word into specific categories, such as PER for persons, ORG for organizations, and others. This format not only enhances clarity but also ensures consistent labeling across texts.

The entity labels are categorized as follows:

- B-label: Indicates the beginning of a labeled entity (Beginning).
- I-label: Indicates that this unit is part of a labeled entity (Inside).
- O: Indicates that this unit is not part of any labeled entity (Outside)

Fig. 2 provides an example of an annotated sentence using the CONLL-2003 format.

a		b
Analiza	O	Analysis
e	O	of
FMN- së	B-ORG	FMN
vjen	O	comes
ndërsa	O	as
liderët	O	leaders
globalë	O	global
të	O	of
biznesit	O	business
dhe	O	and
politikës	O	politics
mblidhen	O	gather
në	O	at
Forumin	B-ORG	Forum
Ekonomik	I-ORG	Economic
Botëror	I-ORG	World
në	O	in
Davos	B- VEND_0	Davos
,	O	
Zvicër	B- VEND_1	Switzerland
.	O	.

Fig. 2. (a) Example of a sentence annotated in the CONLL-2003 format; (b) English translation of each word.

Since the corpus was annotated by two annotators, we conducted a consistency evaluation to assess the reliability of their work. In this test, each annotator was presented with the text annotated by the other and asked to independently annotate it. We then compared the results and found that, in a text of 40 words, only one label differed, resulting in nearly 100% consistency. This rigorous annotation process ensures high-quality and reliable data, significantly enhancing the robustness of the corpus for future analysis.

Table 4 presents the number of words annotated for each label. The statistics show that the most frequent label is PER (person), followed by DATE (date), highlighting a strong focus on identifying individuals and temporal expressions within the annotated texts.

Table 4. Number of tokens per label in the corpus.

Label	Tokens
PER	32,789
DATE_0	19,925
VEND_0	12,012
ORG	10,985
VEND_1	10,378
DATE_1	5,929
EVENT	1,685
PRO	444
RRUGE	419
SHESH	283

4 Machine learning algorithms

We selected the following machine learning algorithms for the NER task using our corpus:

- **Support Vector Machine (SVM):** SVM is a powerful machine learning algorithm used for classification and regression tasks. The main objective of the SVM algorithm is to find the optimal hyperplane in an N-dimensional space that can separate data points into different classes within the feature space. This hyperplane enables the algorithm to determine the class of each data point [18].
- **Naive Bayes (NB):** The NB algorithm is a probabilistic classifier used for classification problems. It assumes that each feature contributes independently to the predictions, with no correlation between them. This algorithm calculates the probability of each class given the input features. Here, Y represents the class, and X denotes the feature. The class with the highest probability is chosen as the final decision [19].

$$P(Y=y|X=(x_1,x_2,\ldots x_n)) \tag{1}$$

- **Random Forest (RF):** The RF algorithm is a powerful tree-based learning technique in machine learning. It belongs to the bagging method within ensemble learning. During the training phase, multiple trees are created using random subsets of the dataset, with each tree measuring a random subset of features at each split. The algorithm aggregates the results of all trees through either voting or averaging [20].
- **Extra Trees (ET):** This algorithm is similar to Random Forest and also belongs to the bagging method. The main difference is that in Extra Trees, a random subset of features is selected at each split in the decision tree, and the best split point is chosen from this subset. Thus, the distinction lies in how the decision trees are constructed [21].
- **Gradient Boosting (GB):** The GB algorithm belongs to the boosting method within ensemble learning techniques. In each iteration, the algorithm trains a new model to minimize the gradient of the loss function. The predictions from the

new model are added to the ensemble, and this process is repeated until a stopping criterion is met [22].

- **Extreme Gradient Boosting (XGBoost):** The XGBoost algorithm follows the same principles as Gradient Boosting but creates a more refined model. Unlike Gradient Boosting, XGBoost fits a special type of tree designed specifically for this algorithm [22].
- **Logistic Regression (LR):** The LR algorithm is used for binary classification, utilizing the Sigmoid function (also known as the logistic function) to take input variables and output a probability value between 0 and 1. In binary classification, the goal is to categorize the input sample into one of two classes. Logistic Regression can also be extended to multi-class classification, which is used when there are more than two classes [23].
- **Multi-Layer Perceptron (MLP):** The MLP is a type of feedforward neural network that consists of an input layer, one or more hidden layers, and an output layer. It learns complex patterns in data through supervised learning by adjusting weights during training using backpropagation. MLPs are effective for tasks such as classification and regression due to their ability to model nonlinear relationships [24].

5 The experiments

In this section, we present the setup of the experiments conducted to train the models, along with the experimental results evaluating the performance of each model.

5.1 Experiments setting

Each algorithm was used to train a model, followed by an evaluation of its accuracy. For each algorithm, three tests were conducted, and both the mean and standard deviation were calculated. The three tests were necessary because the data for training and testing were split using the *train_test_split()* method, which generates a new random split each time it is executed. As a result, the corpus was not pre-divided into training and testing sets. This approach was chosen to avoid the risk of having the last (or first) 20% of data representing a completely different lexicon than the rest, as the collected data spans various topics. The random and mixed splitting provided by *train_test_split()* helps to create a more generalized model, preventing performance evaluations based solely on a specific set of examples.

Models were created for the following algorithms: SVM, Naive Bayes (NB), Random Forest (RF), Extra Trees (ET), Gradient Boosting (GB), Extreme Gradient Boosting (XGBoost), and Logistic Regression (LR). Additionally, three models were developed for the Multi-Layer Perceptron (MLP), labeled as MLPS (MLP Small), MLPM (MLP Medium), and MLPL (MLP Large). For the RF and ET algorithms, the hyperparameter *n_jobs=-1* was utilized, allowing for parallel computation across all available CPU cores. This parameter is specifically applied to RF and ET since they are ensemble methods that independently train multiple decision trees, enabling efficient parallelization. Other algorithms do not use this parameter as they do not involve independent base models that can be trained in parallel. In LR, only the *multi_class* hyperparameter was specified to determine the technique for multi-class classification, with all other hyperparameters left at their default settings.

Starting with the MLP models, the first, MLPS, contains *one hidden layer* with *40 nodes* in addition to the *input* and *output layers*. The second model, MLPM, consists of *two hidden layers* with *120* and *40 nodes*, respectively. Finally, the MLPL model features *three hidden layers* with *100, 120,* and *40 nodes*, respectively. The number of nodes in each layer was chosen to create a pyramid-shaped network structure, which has proven effective in various experimental observations. Basic hyperparameters, such as the number of hidden layers, number of nodes in each layer, and maximum iterations, were set for the MLP models, while other training-related hyperparameters (e.g., *solver='adam'*, *learning_rate_init=0.001*) were left at their default values.

Initially, the corpus was transformed into a list, and sentence strings were converted into lists of tuples. From these, all words and tags were extracted, resulting in two separate lists that identified the unique words and tags within the corpus. Subsequently, a word encoding dictionary and a tag encoding dictionary were created using *enumerate()* function. The words and tags were then organized into lists of lists and encoded based on the dictionaries. These lists were flattened into a single large list and converted into a *NumPy array*, which is crucial for speed and efficiency, given that *NumPy* is implemented in C, making it faster and more memory efficient.

The other models were trained using the default hyperparameters provided by their respective implementations in the scikit-learn package. The initial calculations focused on assessing model accuracy. When distinguishing entities, it is also preferred to evaluate performance using the F1 score. To gain a comprehensive understanding and compare the effectiveness of the algorithms in identifying entities, the average training times for each algorithm were measured.

5.2 Experimental results

We conducted experiments with the selected algorithms to evaluate and compare their performance in identifying entities in Albanian texts. The analysis focused on accuracy, F1 score, and training time for each algorithm, providing a comprehensive evaluation of their overall effectiveness. For each model evaluated, data from three distinct training trials were collected, and the arithmetic mean and standard deviation were calculated for each.

The experimental results regarding the accuracy of each model are presented in Table 5. The Extra Trees (ET) algorithm achieves the highest classification accuracy, with a score of 0.9604, or approximately 96%. This algorithm also has the lowest standard deviation, measuring just 0.0008. The Random Forest (RF) algorithm follows closely, achieving a similar accuracy of 0.9600. In contrast, the Gradient Boosting algorithm, which uses a boosting strategy, performs relatively poorly with an accuracy of only 0.5277. The XGBoost algorithm, which is similar, performs adequately with an accuracy of 0.9148. The Naive Bayes and SVM algorithms achieve accuracies of 0.9122 and 0.9112, respectively, both demonstrating considerable stability with standard deviations not exceeding 0.00118. The Logistic

Regression algorithm achieves an accuracy of 0.9129, with a standard deviation of only 0.00064. The three MLP neural networks show similar classification accuracies, each around 91%.

Table 5. Classification accuracy for each model across the three trials, including the arithmetic mean and standard deviation.

Model	Trial 1	Trial 2	Trial 3	Arithmetic Mean	Standard Deviation
LR	0.9129	0.9134	0.9125	0.9129	0.00064
MLPS	0.9103	0.9110	0.9124	0.9112	0.00107
MLPM	0.9132	0.9125	0.9109	0.9122	0.00118
MLPL	0.9109	0.912	0.9109	0.9118	0.00086
Naïve Bayes	0.9125	0.9128	0.9112	0.9122	0.00085
SVM	0.9115	0.9099	0.9123	0.9112	0.0012
Random Forest	0.9591	0.96	**0.9611**	0.96	0.001
Extra Tree	**0.9595**	**0.9612**	0.9604	**0.9604**	**0.00085**
GB	0.7083	0.3607	0.514	0.5277	0.25
XGB	0.9139	0.9171	0.9133	0.9148	0.002

The F1 score is commonly used in studies that evaluate how well models perform in entity recognition tasks. The F1 values for each model, derived from the three trials, along with their mean and standard deviation, are presented in Table 6.

Table 6. F1 score for each model across the three trials, including the arithmetic mean and standard deviation.

Model	Trial 1	Trial 2	Trial 3	Arithmetic Mean	Standard Deviation
LR	0.8714	0.8721	0.8707	0.8714	0.0007
MLPS	0.8676	0.8688	0.8706	0.869	0.0015
MLPM	0.8717	0.8707	0.8685	0.8703	0.0023
MLPL	0.8684	0.8701	0.8708	0.8697	0.0012
Naïve Bayes	0.8708	0.8712	0.8689	0.8703	0.0055
SVM	0.8693	0.867	0.8704	0.8689	0.0017
Random Forest	0.9569	0.9577	**0.9588**	0.9578	0.0013
Extra Tree	**0.9573**	**0.9589**	0.9581	**0.9581**	**0.0008**
GB	0.7826	0.5082	0.6495	0.6467	0.137
XGB	0.8792	0.8823	0.8784	0.8799	0.002

The results in Table 6 show a similar trend, presenting the F1 score values achieved by each algorithm. The Extra Trees (ET) and Random Forest (RF) algorithms have the best scores, with 0.9581 and 0.9578, respectively, both exhibiting stability with low standard deviations. In contrast, the Gradient Boosting algorithm achieves a maximum F1 score of only 0.6467 and a high standard deviation of 0.137. On the other hand, the XGBoost algorithm performs better, achieving an F1 score of 0.8799. The Naive Bayes and SVM algorithms also demonstrate good results, with F1 scores of 0.8703 and 0.8689, respectively, both accompanied by low standard deviations. Logistic Regression shows an F1 score of 0.8714 with a standard deviation of just 0.0007. The MLP neural networks have closely aligned F1 scores, ranging from 0.869 to 0.8703, despite differences in the number of layers and nodes in their architectures.

Finally, training efficiency is a critical consideration for artificial intelligence models, with models requiring shorter training times being more desirable. In this experiment, we measured and calculated the average training durations for all ten models, as detailed in Table 7.

Table 7. Training time for each model (in seconds).

Model	Training time (s)
Naïve Bayes	0.3491
Extra Tree	24.37
LR	61.27095
Random Forest	72.47
XGB	97.2
MLPS	153.56266
MLPM	214.47567
GB	1,364.4
MLPL	1,842.75
SVM	3,373.4

The results in Table 7 show substantial disparities in training times across the algorithms. Naive Bayes stands out as the fastest, completing training in just 0.3491 seconds. In contrast, the MLPL algorithm is significantly slower, requiring 1,842.75 seconds. Significant differences are also observed among the three different MLP networks. The slowest algorithm, however, is SVM, with a training time of 3,375.4 seconds.

6 Conclusion and future work

This paper presents a Named Entity Recognition (NER) corpus for Albanian texts, manually annotated by two native speakers. We used a tagset consisting of 10 tags to identify entities such as persons, organizations, products, events, places, and dates. The corpus, containing 1,003,836 tokens (56,595 sentences) with 89,850 labeled tokens collected from well-known online news platforms in Albania, is the largest NER corpus for the Albanian language.

This corpus is used to train and evaluate 10 models by implementing algorithms such as Naïve Bayes, Logistic Regression, SVM, Random Forest, Gradient Boosting, Extreme Gradient Boosting, and Multi-Layer Perceptron (MLPS, MLPM, and MLPL), with performance evaluated based on accuracy, F1 scores, and training time.

The experimental results indicate that the Random Forest and Extra Trees algorithms have the highest performance in terms of both accuracy and F1 score, coupled with remarkable stability and low standard deviations. SVM and Naïve Bayes also achieve satisfactory results with high consistency, making them reliable choices for NER tasks. Conversely, the Gradient Boosting algorithm shows subpar performance, with lower values for both metrics and higher variability. While the Extreme Gradient Boosting algorithm outperforms Gradient Boosting, achieving satisfactory results in both metrics, it still lags the top-performing models. Overall, the findings underscore that among ensemble learning techniques, bagging algorithms, particularly Random Forest and Extra Trees, consistently outperform boosting algorithms in effectively identifying entities in Albanian texts.

Furthermore, the three neural networks have similar performance, although their results are slightly lower than those of the Logistic Regression, Random Forest, Extra Trees, and XGBoost algorithms. Notably, the depth of the MLP neural networks does not appear to have a substantial effect on their performance. However, increased depth significantly impacts training times, leading to longer durations for deeper networks.

As future work, we aim to explore deep learning techniques by implementing models such as LSTM, BERT, and RoBERTa. Additionally, we plan to investigate the impact of news article topics on the training process.

References

1. Jehangir, B., Radhakrishnan, S., and Agarwal, R.: A survey on Named Entity Recognition— datasets, tools, and methodologies. Natural Language Processing Journal, (2023) 3, 100017
2. Kote, N., Rushiti, R., Çepani, A., Haveriku, A., Trandafili, E., Kajo Mece, E., Skenderi Rakipllari, E., Xhanari, L., Deda A.: Universal Dependencies Treebank for Standard Albanian: A new approach. Proceedings of the Sixth International Conference Computational Linguistics in Bulgaria (CLIB 2024), Sofia, Bulgaria, (2024) 80-89
3. Kote, N., Biba, M., Kanerva, J., Rönnqvist, S., and Ginter, F.: Morphological Tagging and Lemmatization of Albanian: A Manually Annotated Corpus and Neural Models. ArXiv, abs/1912.00991 (2019)
4. Haveriku, A., and Frashëri, N.: Building a sentiment analysis model for tweets in Albanian language. Proceedings of International Conference on Information Technologies and Educational Engineering (ICITEE 21), Tirana, Albania, (2021) 273 - 280
5. Kote, N., Biba, M.: A Neural Network Model for Opinion Mining in Albanian Language. Proceedings of International Conference on Information Technologies and Educational Engineering (ICITEE 21), Tirana, Albania, (2021) 293 - 299.
6. Kote, N., and Biba, M.: An Experimental Evaluation of Algorithms for Opinion Mining in Multi-Domain Corpus in Albanian. Proceedings Foundations of Intelligent Systems: 24th International Symposium, ISMIS 2018, Limassol, Cyprus (2018) 439–447.
7. Kote, N., and Biba, M.: Opinion Mining in Albanian: Evaluation of the Performance of Machine Learning Algorithms for Opinions Classification. International Journal of Innovative Science and Research Technology (2021) 6(6) 964 - 973.
8. Muraku, B., Xiao, L., Meçe, E.K.. Toward Detection of Fake News Using Sentiment Analysis for Albanian News Articles. In: Barolli, EIDWT 2024. Lecture Notes on Data Engineering and Communications Technologies, vol 193. Springer, Cham. https://doi.org/10.1007/978-3-031-53555-0_55 (2024) 575-585

9. Kote, N., Trandafili, E., and Plepi, G.: Question Classification for Albanian Language: An Annotated Corpus and Classification Models. Advances on P2P, Parallel, Grid, Cloud and Internet Computing - Proceedings of the 17th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC-2022) Tirana Albania (2022) 1 - 12
10. Trandafili, E., Kote, N., and Plepi, G.: Question Classification in Albanian Through Deep Learning Approaches. International Journal of Advanced Computer Science and Applications (2023) 14(3)
11. Çano, E.: AlbNER: A Corpus for Named Entity Recognition in Albania. arXiv:2309.08741 (2023)
12. Skënduli, Prifti, M., and Biba, M.: A Named Entity Recognition approach for Albanian. International Conference on Advances in Computing, Communications and Informatics (ICACCI). Mysore India (2013) 1532 - 1537.
13. Trandafili, E., Kajo Meçe, E., and Duka, E.: A Named Entity Recognition Approach for Albanian Using Deep Learning. Complex Pattern Mining: New Challenges, Methods and Applications (2020) 85-101.
14. Hoxha, K., and Kono, G.: Named Entity Recognition in Albanian Based on CRFs Approach. Proceedings of the 2nd International Conference on Recent Trends and Applications in Computer Science and Information Technology (RTA-CSIT) (2016) 18, 95–108
15. Hoxha, K., and Baxhaku, A.: An Automatically Generated Annotated Corpus for Albanian Named Entity Recognition. Cybernetics and Information Technologies (2018) 18(1) 95 - 108
16. Tkachenko, M., Malyuk, M., Holmanyuk, A., and Liubimov, N.: {Label Studio}: Data labeling software. https://github.com/HumanSignal/label-studio
17. F. Tjong, E., Sang, K., and De Meulder, F.: Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. arXiv:cs/0306050 (2003)
18. Singh, Thoudam Doren, Nongmeikapam, Kishorjit, Ekbal, Asif and Bandyopadhyay, Sivaji: Named Entity Recognition for Manipuri Using Support Vector Machine. Proceedings of the 23rd Pacific Asia Conference on Language, Information and Computation, 2 (2009)
19. Wild, S., Parlar, S., Hanne, T., Dornberger, R.: Naïve Bayes and Named Entity Recognition for Requirements Mining in Job Postings", 3rd International Conference on Natural Language Processing (ICNLP) Beijing China (2021) 155 - 161
20. Breiman, L.: Random Forests. Machine Learning. (2001) 45(1) 32-45.
21. Geurts, P., Ernst, D., and Wehenkel, L.: Extremely randomized trees. Machine Learning. (2006) 63(1) 3 - 42.
22. He, Z., Lin, D., Lau, T., and Wu, M.: Gradient Boosting Machine: A Survey. arXiv:1908.06951 (2019)
23. Jurafsky, D., H. Martin, J.: Logistic Regression. I Speech and Language Processing. (2024)
24. Murtagh, F.: Multilayer perceptrons for classification and regression. Neurocomputing. (1991) 2 (5-6).