Machine learning specialist task

Description of the task:
We want you to build a text classification service.
The model on which the service should be based on may be binary or multiclass/multilabel depending on the dataset of your choice.

Expected artifacts:

1. Model training code and description. It might be a script accompanied with description of data, training parameters and discussion on validation results OR Jupyter Notebook including everything aforementioned.
2. Web service code that has at least one endpoint '/predict' that accepts POST request with list of texts and returning list of corresponding classes.

Additional info:
The task description was intentionally made high-level so you are flexible in selecting dataset, frameworks, programming languages etc.
You can even use pretrained models but still please include data exploration and model validation analysis in the case.
It is highly recommended to use Docker for packaging the web service so we can run the application without dependencies hustle.
Expected completion time for the task is 4-8 hours (excluding model training time) depending on your experience with related technologies.

TF-IDF

Если документ содержит 100 слов, и слово[3] «заяц» встречается в нём 3 раза, то частота слова (TF) для слова «заяц» в документе будет 0,03 (3/100). Вычислим IDF как десятичный логарифм отношения количества всех документов к количеству документов содержащих слово «заяц». Таким образом, если «заяц» содержится в 1000 документах из 10 000 000 документов, то IDF будет равной: log(10 000 000/1000) = 4. Для расчета окончательного значения веса слова необходимо TF умножить на IDF. В данном примере, TF-IDF вес для слова «заяц» в выбранном документе будет равен: 0,03 × 4 = 0,12.

Мера TF-IDF часто используется для представления документов коллекции в виде числовых векторов, отражающих важность использования каждого слова из некоторого набора слов (количество слов набора определяет размерность вектора) в каждом документе.

monogram, n-grams

Selecting Dataset
Text classification can be applied to a wide range of tasks.

Text is one of the most common types of unstructured data. Analyzing, understanding, organizing, and sorting through text data is hard and time-consuming to extract value from that.
The fundamental tasks in Natural Language Processing (NLP) can be divided into

- sentiment analysis (determining whether a text is positive, negative, or neutral),
- topic labeling,
- spam detection
- intent detection.

A few typical applications of text classification technology for companies include:

- Social media monitoring.
- Brand monitoring.
- Customer service

At the company's web-page there are two text-input entry point:

- Contact form
- Application form

Classification of the topic in each of the entry point seems to create the most value to the company performance. For example, defining information from application form such as type of goods to deliver, conditions and term provided by customer in text format will help to better process the application.

Based on that, I was searching for a dataset for topic classification. Among other open-source datasets the "news classification" dataset that is available in a Python sklearn package can be the one to create a topic classification service.