IT UNIVERSITY OF COPENHAGEN

In-depth evaluation of cross-domain language identification methods

Project management (GEP)
Final Assignment

Víctor Cabré Guerrero

Bachelor Thesis
Specialization in Computing

Supervisor: Rob van der Goot

December 1st, 2024

Contents

1 Introduction and contextualization

Language identification is the process of automatically determining the language of a given piece of text. It is one of the most common Natural Language Processing (NLP) tasks, and it has a wide variety of applications. For instance, it is used in machine translation to determine the language of an input, and in preprocessing pipelines for data used in training Large Language Models (LLMs) (Penedo et al., 2023). Existing models perform well within specific domains (Cavnar, Trenkle, et al., 1994), but their performance often degrades when applied to texts from different domains. This is particularly important because the training data available for language identification models is often limited to specific domains, which constrains their ability to generalize. Some recent developments have introduced new methods, such as the use of deep neural networks, which are trained on labeled data from multiple domains. The objective of this research project is to perform an in-depth analysis of different language identification methods on cross-domain data, evaluate their relative performance, and assess their capabilities and limitations, while also providing a holistic view of the current state of language identification technology.

As language continually evolves and transforms, technological solutions must likewise adapt to meet new demands. Whereas traditional domains such as news and rights have not changed significantly, recent domains like social media have created new challenges for existing language identification technology. The use of *bad language*, defined as "text that defies our expectations about vocabulary, spelling, and syntax" (Eisenstein, 2013), introduces some phenomena which can degrade performance in current methods. Code-switching, non-standard spelling, abbreviations and unconventional characters such as emoji are some examples. Understanding the current open challenges in language identification is crucial for developing more robust and adaptable systems. The dynamic nature of modern communication demands a paradigm shift in how we approach language identification. Future solutions must be more adaptable, context-aware, and proficient in handling cross-domain data.

1.1 Context

This thesis corresponds to the Specialization in Computing of the Bachelor's degree in Informatics Engineering taught at Faculty of Informatics of the Polytechnic University of Catalonia (*Facultat d'Informàtica de Barcelona, Universitat Politècnica de Catalunya*) (UPC-FIB). The thesis is being de-

veloped at the IT University of Copenhagen (*IT-Universitetet i København*) (ITU) under the supervision of Rob van der Goot, as part of an Erasmus+ exchange financed by the European Union. The thesis defense and grading will take place in Denmark, following ITU's rules and regulations.

1.2 Concepts

In order to understand the project, the reader should be familiar with general machine learning concepts, such as supervised and unsupervised learning, model evaluation metrics, and feature representation. Familiarity with text processing techniques, including tokenization, vectorization, and the handling of multilingual data, is also beneficial.

1.2.1 Language identification

Language identification is the task of automatically determining the language or languages that a text is written in. Figure 1 illustrates this task with some example texts. Language identification has been thought to be a solved problem for a long time, with traditional methods such as N-gram overlap reporting strong metrics on in-domain data (Cavnar, Trenkle, et al., 1994). However, recent changes to the use of language in social media and other non-traditional domains have caused performance of traditional language identification technology to degrade (Baldwin and Lui, 2010).

There are many techniques for language identification, from simple rule-based methods to complex neural networks. Some of the most known methods will be briefly explained below.

1.2.2 N-gram overlap: TextCat

Cavnar, Trenkle, et al. (1994) propose using N-Gram frequency profiles for text categorization, such as language or domain identification. We will focus on its language identification capabilities. The approach consists of generating language profiles for each language in the training data, by computing a ranked list of the most frequent character N-grams. For every new document, a document frequency profile is computed. Then, an Out-Of-Place measure is calculated between the language profiles and the document profile, and the language with the lowest distance is selected as the output language.

This approach achieves almost perfect performance in some of the tests conducted, although Baldwin and Lui (2010) show that a higher number of languages, less training data or documents of lower length can impact performance.

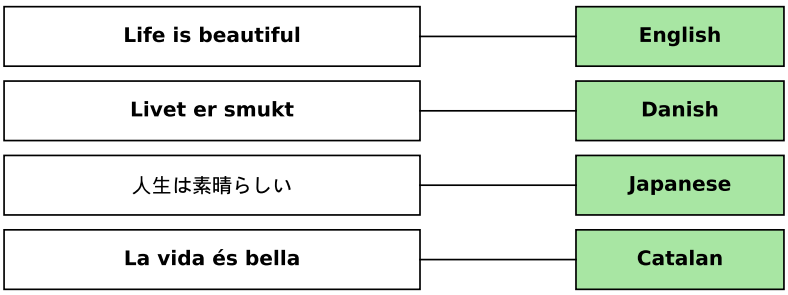


Figure 1: Texts and their corresponding languages as classified by `langid.py`.

We use TextCat, an implementation of the method, developed by van Noord (1997).

1.2.3 Naïve Bayes: `langid.py`

Lui and Baldwin (2012) present `langid.py`, a language identification tool that uses a Naïve Bayes classifier trained over byte n-grams. Their evaluation shows that the model is more robust on cross-domain data compared to TextCat, while outperforming it in terms of speed. Furthermore, an approach for cross-domain feature selection in language identification is proposed in Lui and Baldwin (2011).

Additionally, we provide our own implementation of a Naïve Bayes-based language identification method and attempt to extract common features from multiple domains.

1.2.4 Transformer-based language models: Glot500

Imani et al. (2023) present a language model that covers over 500 languages, including low-resource ones, which can be used for language identification and other NLP tasks. Glot500 outperforms most other language identification methods, at the cost of speed.

Glot500 is an extension of XLM-R-base (Conneau et al., 2020). It uses the transformers architecture introduced in Vaswani et al. (2017). We will use Glot500 as the state-of-the-art open source language identification method in terms of accuracy.

1.3 Problem to be solved

The recent advancements in language usage have caused traditional language identification methods' performance to decrease. Phenomena such as code-switching, non-standard spelling, abbreviations and non-standard characters are common examples of language usage that cause existing methods to perform significantly worse. To the best of our knowledge, an in-depth evaluation of the current state of cross-domain LangID hasn't been published. This research project aims to identify the open challenges in cross-domain language identification, provide some insights as to which setups negatively impact current methods, and propose some ways to tackle these problems.

1.4 Stakeholders

The parties directly involved in the project are Víctor Cabré Guerrero, the thesis researcher; Rob van der Goot, the thesis supervisor; and both universities, *IT-Universitetet i København* and *Universitat Politècnica de Catalunya*. Additionally, with less direct involvement, the NLPnorth research group and its members, including other students developing their thesis on NLP topics. Other stakeholders that are not directly involved include beneficiaries of this work, such as academic or industry researchers, students, companies, nonprofit organizations, policymakers, community members, educational institutions, and technology developers. Each of these groups can leverage the outcomes of this work for various purposes, including enhancing their research and developing new technologies and applications.

2 Justification

2.1 Previous studies

Language identification has been a topic of research in NLP for a long time. High performances have been obtained with relatively simple methods, like N-gram based classification (Cavnar, Trenkle, et al., 1994) and Naïve Bayes classifiers (Lui and Baldwin, 2012); and with more complex solutions, such as the use of static word embeddings (Mikolov et al., 2017), Long Short-Term Memory (Toftrup et al., 2021) or transformers (Imani et al., 2023). Baldwin and Lui (2010) identified some challenges that arise when the experimental setups are less constrained. A larger number of languages, shorter or less informative documents, and an imbalance in the distribution of language classes can negatively impact performance. Lui and Baldwin (2011) show

that cross-domain setups are particularly challenging, and propose a solution based on feature selection.

2.2 Open challenges

While language identification systems have shown almost perfect results in controlled settings, their performance often deteriorates significantly when applied across different domains. Text from different domains, such as social media, legal documents, and wiki data present distinct characteristics that challenge existing models. This cross-domain adaptation problem is particularly noticeable when dealing with informal language, non-standard terminology, and domain-specific abbreviations. This cross-domain scenario represents the most realistic and relevant setup, as real-world applications rarely operate within the boundaries of a single domain.

Several factors contribute to the persistence of this challenge. First, the linguistic features that are relevant in one domain may not be equally useful in another. For example, models trained on text from rights or legal domains often struggle with social media content due to non-standard spellings, informal expressions, code-switching and non-standard characters or utterances like emojis, hashtags and usernames. Second, the distribution of languages varies significantly across domains, with some low-resource languages not being represented enough, leading to domain-specific biases in language identification systems.

This research aims to address these open challenges by tackling the root causes and investigating robust approaches to cross-domain language identification by focusing on domain-invariant features.

3 Scope

The main goal of the project is to identify the strengths and the shortcomings of existing language identification methods, and to provide a holistic view of the current state of language identification technology. Additionally, we believe that investigating ways to mitigate the issues in language identification would be a positive contribution to the NLP community. Therefore, we set that as an optional goal.

3.1 Objectives

Next, we lay out all the specific objectives of the project.

- Train and evaluate the most commonly used language identification methods on our own dataset.
- Create a comprehensive comparison of the methods' performance on cross-domain data.
- Identify problems in language classification, specifically on a per-language basis (or language pairs).
- Research ways to mitigate issues caused by cross-domain setups.
- Provide a holistic evaluation of the current state of language identification technology.

3.2 Requirements

These requirements are necessary in order to ensure the quality of the research.

- All the code used for testing and evaluating the models will be public when the research is concluded. As such, we must follow good programming practices. The code should be readable and easy to maintain.
- All experiments need to be reproducible. We will provide detailed instructions on how to set up the environment and run the code. This includes specifying all dependencies and their versions.
- The dataset used for the evaluation will be made publicly available, subject to appropriate licensing and citation requirements, where permitted by copyright law. If this is not possible, the scripts for automatically scraping the data will be provided instead.
- We will report comprehensive evaluation metrics across multiple language pairs and domains. Statistical significance of the results will be assessed.

3.3 Potential obstacles and risks

Some risks could prevent us from reaching a favorable outcome. The following are some obstacles that pose significant risk for the successful completion of the research project. This list is not extensive: unforeseen risks could arise during the execution of the project.

- **Deadline of the project.** The deadline for submitting the project to LearnIT (ITU's e-Learning platform) is December 16, 2024. The thesis defense occurs on January 16, 2025. These deadlines pose a risk in case of bad planning or unexpected obstacles.
- **Researcher's unfamiliarity with NLP.** Although I'm familiar with machine learning, I don't have any prior hands-on experience with NLP tasks.
- **Unavailability of ITU's HPC.** ITU provides its students and staff a High Performance Computer (HPC) for programs that require a lot of compute, but its availability is limited. We will need the HPC to train some of the computationally heavy models.

3.4 Methodology and rigor

3.4.1 Methodology

This section describes the methodology that we will follow during our research. Since our project is purely research based, and not focused on developing any particular applications or solutions, we will not specify any development methodologies.

- **Quantitative analysis.** We will use a range of established machine learning evaluation metrics to assess the performance of the language identification models, including accuracy, precision, recall and F1-score. Statistical significance testing will be used to determine if the observed performance differences between models are statistically meaningful.
- **Qualitative analysis.** We will perform case by case studies on misclassified samples to understand the underlying reasons for the models' errors in the cross-domain setup. Basic linguistic analysis will be conducted to identify the key language features that influence the models' predictions.
- **Reproducibility.** All code, data, and experimental details will be made publicly available to facilitate reproducibility and enable future research.

3.4.2 Monitoring tools and validation

We will use `git`, a version control system. This tool allows for collaborative development, tracking changes, and managing code repositories. Throughout the project, we will maintain a Git repository hosted on a platform like GitHub or GitLab.

A weekly or bi-weekly meeting will be scheduled between the researcher and the supervisor at ITU. The meetings will be scheduled using Outlook's meeting scheduling feature.

4 Temporal planning

This project should have a 504 hour workload, according to the amount of ECTS credits that this project has (18 ECTS in total, 15 ECTS for the project itself and 3 ECTS for the GEP course). The temporal planning of the project will be done according to this expected workload.

4.1 Task definition

In this section we describe all the tasks and subtasks that need to be completed.

Project planning (PP) - 90 hours

This section primarily relates to all the planning and preparation that needs to be done before the start of the research.

- **Scope (PP.1) - 10 hours.** Defining the objectives, requirements, potential risks and the methodology that will be used in the research. Additionally, defining the context and explaining some basic concepts in language identification.
- **Temporal planning (PP.2) - 15 hours.** Defining the tasks for the project, and, for each task, its start and end dates, and its requirements.
- **Budget analysis (PP.3) - 10 hours.** Analysis of the costs associated with the equipment and human resources involved in the project.
- **Sustainability assessment (PP.4) - 10 hours.** Analysis of the environmental, social and economic impact of the project.

- **Researcher-supervisor meetings (PP.5) - 20 hours.** A number of meetings between the researcher and the supervisor are scheduled.
- **Creation of the project management document (PP.6) - 25 hours.** Creation of a document that comprises all the information in this category.

PP.1, PP.2, PP.3, PP.4 and PP.6 are linear dependencies, because of the nature of project planning. First we need to define the scope and the objectives of the project, and from there we can identify the tasks that need to be completed. Once the tasks have been identified, we can estimate the temporal and budgetary cost of each task. Finally, we can compile all the project planning information in the document.

NLP familiarization (NF) - 160 hours

As part of the thesis, I will attend the *Advanced Natural Language Processing and Deep Learning* course, which is offered as part of the M.Sc. in Data Science at ITU (40 hours of lectures and 50 hours of course work). I will be attending as an auditing student, but will participate in activities and complete the deliverables. Additionally, I will conduct research on the topic of NLP for a total of 70 hours.

Literature review (LR) - 30 hours

Read literature in the world of NLP in order to understand the current state of language identification. This includes scientific papers, books and web resources from a wide variety of sources.

- Review classic and modern language identification methods.
- Understand challenges in language identification derived from new or non-standard language usage, and how low-resource languages are affected.
- Analyze existing cross-domain approaches and their effectiveness in different scenarios.
- Study evaluation methodologies used in previous research to establish best practices for our own evaluation framework.
- Identify current limitations in language identification research, particularly in cross-domain scenarios.

Experimentation (EX) - 175 hours

The experimentation phase constitutes the core of our research, where we will systematically evaluate and compare different cross-domain language identification methods.

- **Environment setup (EX.1) - 10 hours.** Setting up the virtual environment and all scripts, libraries and programs that are needed.
- **Model development (EX.2) - 50 hours.** Training and fine-tuning selected models using cross-domain data, using the high performance computer when needed. If necessary, selecting the best hyperparameters experimentally. The dependency for this task is EX.1, because the environment needs to be set up before being able to use all the libraries and programs that are necessary for developing the models.
- **Performance analysis (EX.3) - 65 hours.** Reviewing the performance of language identification on unseen cross-domain data, and investigating each model's strengths and weaknesses. The dependency for this task is EX.2, because the models need to be trained before they can be evaluated.
- **Enhancement strategies (EX.4) - 50 hours.** Based on evaluation results, explore and implement techniques to improve cross-domain generalization. The dependency for this task is EX.3, because a thorough analysis of the models' performance allows us to devise new strategies that mitigate the issues.

Paper and defense preparation (PD) - 105 hours

The final phase involves documenting our findings and preparing for the thesis defense.

- **Paper development (PD.1) - 60 hours.** Writing the thesis document following established academic writing conventions. Incorporate feedback from supervisors. Create clear and informative visualizations, tables and figures.
- **Code repository finalization (PD.2) - 15 hours.** Prepare the final version of the code repository with comprehensive documentation, usage instructions, and examples. Ensure all code is well-commented and follows best practices for reproducibility.

- **Defense preparation (PD.3) - 30 hours.** Create a presentation that highlights the key contributions and findings of the research.

4.2 Summary of tasks

Table 1 provides a comprehensive summary of all the tasks, the amount of time allocated per task and each task's dependencies. The total amount of hours closely resembles the amount of hours dictated by the amount of ECTS credits for the thesis.

ID	Task	Time (h)	Dependencies
PP	**Project planning**	**90**	-
PP.1	Scope	10	-
PP.2	Temporal planning	15	PP.1
PP.3	Budget analysis	10	PP.2
PP.4	Sustainability assessment	10	PP.3
PP.5	Meetings	20	-
PP.6	Project management document	25	PP.4
NF	**NLP familiarization**	**160**	-
NF.1	Advanced NLP course lectures	40	-
NF.2	Advanced NLP course work	50	-
NF.3	Autonomous NLP research	70	-
LR	**Literature review**	**30**	-
EX	**Experimentation**	**175**	-
EX.1	Environment setup	10	-
EX.2	Model development	50	EX.1
EX.3	Performance evaluation	65	EX.2
EX.4	Enhancement strategies	50	EX.3
PD	**Paper and defense preparation**	**105**	-
PD.1	Paper development	60	-
PD.2	Code repository finalization	15	-
PD.3	Defense preparation	30	-
-	**Total**	**560**	**-**

Table 1: Task overview

4.3 Resources

A variety of both human and material resources will be needed for the thesis.

4.3.1 Human resources

Human resources include the student, supervisor and GEP tutor. For the purpose of this document, we identify different roles for the student, depending on the type of task being performed: the project manager, responsible for project planning and for overseeing all other roles; the researcher, leading the theoretical research; the programmer, tasked with coding and executing the practical application of the theoretical research done by the researcher; the tester, who will report any bugs or issues in the code; and the technical writer, charged with writing the final paper and preparing the defense. The supervisor is tasked with following the progress made by the researcher, reviewing the methodology used and suggesting improvements. The GEP tutor is tasked with providing feedback for the GEP assignments and grading the Final Assignment. Additionally, there are a lot of human resources that are not directly involved in the project, but without which the thesis could not be completed. This includes university administrators and HPC maintainers.

4.3.2 Material resources

Material resources refer to objects, tools or devices that we will use during the research. Some material resources include:

- **Laptop.** We will be using a Lenovo Ideapad 5 Pro, which has an AMD Ryzen 7 8845HS processor with 32 GB of LPDDR5x-6400 RAM. This laptop will be running Fedora, a Linux distribution.
- **High Performance Computer.** We will be using ITU's HPC in order to train some of the heavier models. This HPC allows students to use GPUs such as the Nvidia A100 or V100.

Table 2 shows a summary of the roles and the material involved in each of the tasks.

4.3.3 Software requirements

We will be using a wide array of programs and scripts.

- **Operating system.** We will use Fedora, a Linux distribution known for its community-driven development, stability and security. Particularly, we will be using version 41 Workstation.

ID	Task	Roles	Material
PP	**Project planning**		
PP.1	Scope	PM	Laptop
PP.2	Temporal planning	PM	Laptop
PP.3	Budget analysis	PM	Laptop
PP.4	Sustainability assessment	PM	Laptop
PP.5	Meetings	PM,P,T,R,TW	Laptop
PP.6	Project management document	PM,TW	Laptop
NF	**NLP familiarization**		
NF.1	Advanced NLP course lectures	R	-
NF.2	Advanced NLP course work	R	Laptop
NF.3	Autonomous NLP research	R	Laptop
LR	**Literature review**	R	Laptop
EX	**Experimentation**		
EX.1	Environment setup	P,T	Laptop
EX.2	Model development	R,P,T	Laptop, HPC
EX.3	Performance evaluation	R,P	Laptop
EX.4	Enhancement strategies	R,P	Laptop
PD	**Paper and defense preparation**		
PD.1	Paper development	PM,R,TW	Laptop
PD.2	Code repository finalization	P,T	Laptop
PD.3	Defense preparation	PM,R,TW	Laptop

Table 2: Summary of human and material resources per task. PM: Project Manager, P: Programmer, T: Tester, R: Researcher, TW: Technical Writer

- **Text editor.** We will use Visual Studio Code for the majority of coding, including all the document preparation using the LaTeX Workshop extension. Other minor documents and scripts will be edited using `vim`, a lightweight text editor.
- **Version control system.** We will use `git`, a commonly used version control system. The remote repository will be hosted on a platform like GitHub or GitLab.
- **Python interpreter.** We will be using the official Python interpreter (version 3.13) which comes pre-installed with Fedora 41 Workstation. We choose Python because of its syntax, rapid prototyping capabilities and because it's deeply established as the language of choice for a wide variety of machine learning tasks.

- **Python packages.** We will use multiple Python packages which we will install and manage using `pip` and a virtual environment. This includes Scikit-Learn, Pandas, PyTorch and Transformers. For convenience, we will use `virtualenvwrapper`, which allows for easy activation and deactivation of virtual environments using commands like `workon`.

4.4 Risk management

As seen in section 3.3, some obstacles may appear during the execution of the project. This section details the impact of each risk and the recommended approach to mitigate these risks.

Deadline of the project

The temporal planning may not be completely accurate, and unforeseen obstacles can delay some of the tasks. This can pose a threat towards the successful and on-time completion of the project.

- **Impact:** Medium
- **Buffer time:** 40 hours
- **Suggested mitigation:** Continuously adapt the temporal planning to account for new challenges, recalculating the amount of hours and ensuring that the deadline can be met.

Researcher's unfamiliarity with NLP

Being unfamiliar with natural language processing can pose a risk towards having a successful research project.

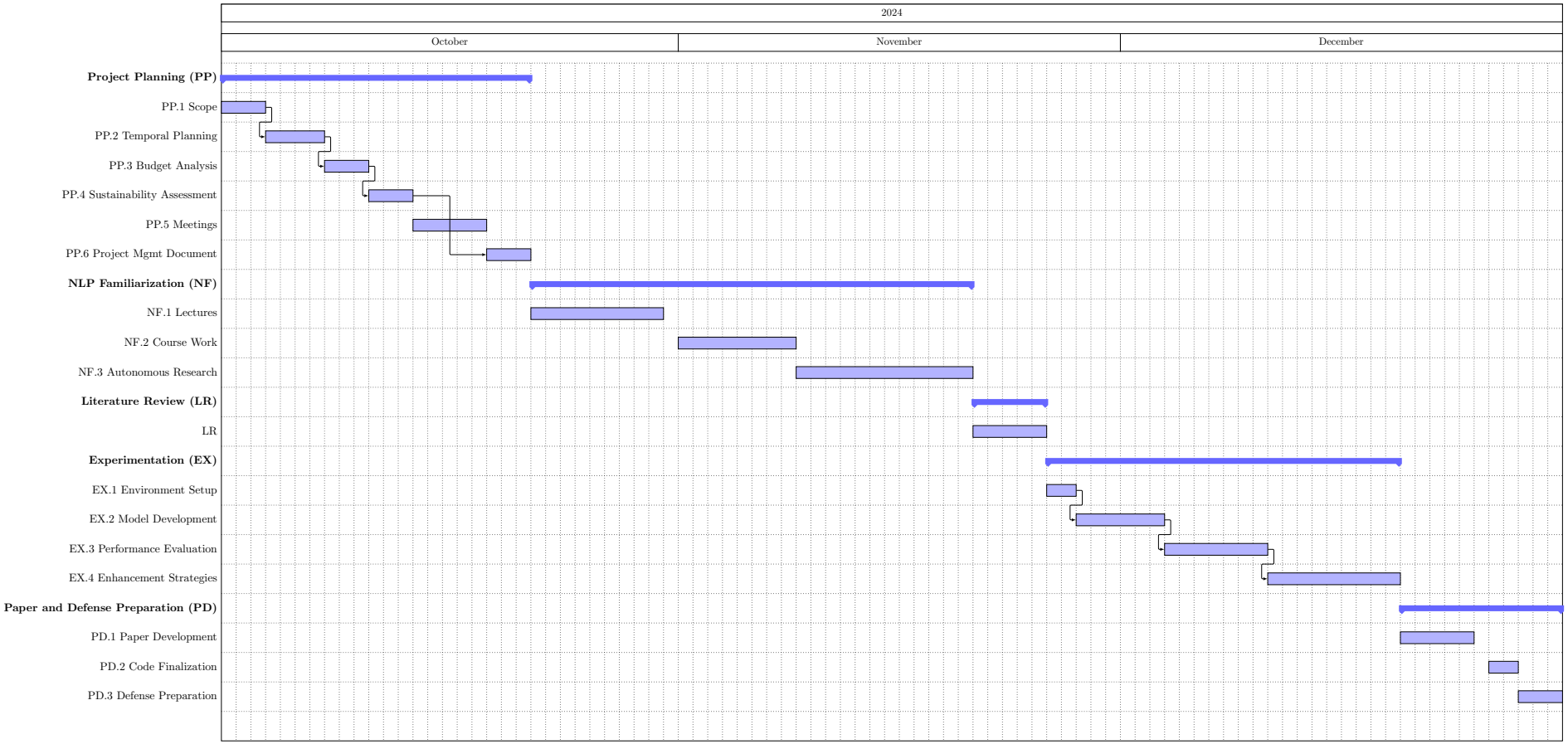
- **Impact:** High
- **Buffer time:** 50 hours
- **Suggested mitigation:** I will attend the *Advanced Natural Language Processing and Deep Learning* course, which is offered as part of the M.Sc. in Data Science at ITU. Additionally, I will carry out independent research beyond the scope of the literature review.

Unavailability of ITU's HPC

Not being able to use the High Performance Computer could mean that some of the heavier models, such as Glot500, would not be used in the thesis.

- **Impact:** Medium
- **Buffer time:** 50 hours
- **Suggested mitigation:** By default, ITU gives students access to the brown queue. Additionally, there is a red queue, which can be accessed by staff and researchers. If necessary, we could ask for permission to use the red queue.

4.5 Gantt chart



5 Budget

The project has some costs associated with it. There are staff (personnel) costs and material costs. All amounts are in Danish crowns (kr./DKK).

5.1 Staff costs

Realistically, a project would involve people with different roles. In a project like this, we would have a project manager, a programmer, a tester, a researcher and a technical writer. Table 3 provides a summary of each role's yearly salary and price per hour[1]. Table 4 shows the amount of hours each role is going to dedicate to each task. Finally, using the hourly cost of each role and the number of hours per role and task, we compute the total cost per task in table 5, where we also add the social security contribution paid by the employer.

Role	**Annual salary (kr.)**	**Price per hour (kr.)**[2]
Project manager	600,000	288.46
Programmer	300,000	144.23
Tester	250,000	120.19
Researcher	280,000	134.62
Technical writer	250,000	120.19

Table 3: Personnel costs per role

[1]The amounts are approximated to what a person working a specific role would be earning in a Danish university. All the positions are junior/little experience. The reference amounts were provided by the supervisor.

[2]Assuming an 8 hour workday, 52 week year with no vacation.

ID	Task	Time (h)	Time per role (h)				
			Project manager	**Programmer**	**Tester**	**Researcher**	**Technical writer**
PP	**Project planning**	**90**	**54**	**4**	**4**	**4**	**24**
PP.1	Scope	10	10	0	0	0	0
PP.2	Temporal planning	15	15	0	0	0	0
PP.3	Budget analysis	10	10	0	0	0	0
PP.4	Sustainability assessment	10	10	0	0	0	0
PP.5	Meetings	20	4	4	4	4	4
PP.6	Project management document	25	5	0	0	0	20
NF	**NLP familiarization**	**160**	**0**	**60**	**0**	**100**	**0**
NF.1	Advanced NLP course lectures	40	0	20	0	35	0
NF.2	Advanced NLP course work	50	0	20	0	35	0
NF.3	Autonomous NLP research	70	0	20	0	30	0
LR	**Literature review**	**30**	**0**	**0**	**0**	**30**	**0**
EX	**Experimentation**	**175**	**0**	**70**	**5**	**100**	**0**
EX.1	Environment setup	10	0	10	0	0	0
EX.2	Model development	50	0	25	5	20	0
EX.3	Performance evaluation	65	0	25	0	40	0
EX.4	Enhancement strategies	50	0	10	0	40	0
PD	**Paper and defense preparation**	**105**	**25**	**10**	**5**	**30**	**35**
PD.1	Paper development	60	10	0	0	20	30
PD.2	Code repository finalization	15	0	10	5	0	0
PD.3	Defense preparation	30	15	0	0	10	5
-	**Total**	**560**	**79**	**144**	**14**	**264**	**59**

Table 4: Personnel hours per task

ID	Task	Cost (kr.)
PP	**Project planning**	**20057.56**
PP.1	Scope	2884.60
PP.2	Temporal planning	4326.90
PP.3	Budget analysis	2884.60
PP.4	Sustainability assessment	2884.60
PP.5	Meetings	3230.76
PP.6	Project management document	3846.10
NF	**NLP familiarization**	**22115.80**
NF.1	Advanced NLP course lectures	7596.30
NF.2	Advanced NLP course work	7596.30
NF.3	Autonomous NLP research	6923.20
LR	**Literature review**	**4038.60**
EX	**Experimentation**	**24159.05**
EX.1	Environment setup	1442.30
EX.2	Model development	6899.10
EX.3	Performance evaluation	8990.55
EX.4	Enhancement strategies	6827.10
PD	**Paper and defense preparation**	**17500.00**
PD.1	Paper development	9182.70
PD.2	Code repository finalization	2043.25
PD.3	Defense preparation	6274.05
-	**Social security contribution**[3]	**4038.46**
-	**Total**	**91909.47**

Table 5: Total cost per task and total staff cost

5.2 Generic costs

Generic costs refer to the physical objects, items and facilities that will be used during the project. For instance, a laptop, the High Performance Computer, the physical room where the researcher will be or the electricity that is going to be used.

In this section we will use the concept of *amortization*, which is the process of reducing the cost of an asset over time. In order to understand this concept, we propose the following scenario. We purchase a laptop for

[3]Calculated as the total amount of hours multiplied by the hourly social security contribution, rounded to 15000 kr., assuming an 8 hour workday, 52 week year with no vacation.: https://www.bdo.dk/en-gb/insights/tax-and-vat/danish-social-security-contributions

10,000 kr., and we expect to use it for the next five years. Amortization is calculated so that the cost is divided evenly across the laptop's useful life. In our case, 10,000 kr./5 years = 2,000 kr./year. This means that for each year, 2,000 kr. is the cost associated with using the laptop.

5.2.1 Hardware

We will be using a variety of hardware devices, including a laptop, peripherals, and the HPC. This section shows which devices and machines are going to be used, their total price and their amortized cost.

For each hardware device, we show the price, useful life, expected usage time during the project and the final amortization value, computed as:

$$\text{Amortization (kr.)} = \frac{\text{Price (kr.)}}{\text{Useful life (h)}} \times \text{Time used (h)}$$

Hardware	**Price (kr.)**	**Useful life (h)**	**Time used (h)**	**Amortization (kr.)**
Lenovo Ideapad 5 Pro	8000	20000	560	224.00
HPC (Nvidia A100)[4]	52000	40000	150	195.00
Computer mouse	300	15000	560	11.20

Table 6: Hardware information and amortized cost

5.2.2 Workspace

The project will be developed at ITU, in one of the common areas or inside one of the meeting rooms. We estimate the monthly cost of a room of similar characteristics to be 4000 kr. The total cost for 6 months is 24000 kr.

5.2.3 Electricity

The kWh price in Denmark is 2.8 kr.[5] Assuming the room has 10 LED bulbs of 8W each, the total wattage is 80 W. The energy consumption for the 560 hours of work is as follows:

$$E_{room} = \frac{80 \text{ W} \times 560 \text{ h}}{1000} = 44.8 \text{ kWh}$$

[4]Only the GPU is being considered here. There are other components such as CPU or memory that we will skip since their price is very low compared to the GPU.

[5]Data for the first half of 2024, obtained from Danmarks Statistik, a Danish governmental organization: https://www.dst.dk/en/Statistik/emner/miljoe-og-energi/energiforbrug-og-energipriser/energipriser

The total cost of electricity in the room, assuming no other costs such as heating, is:

$$\text{Room electricity cost} = 44.8 \text{ kWh} \times \frac{2.8 \text{ kr.}}{1 \text{ kWh}} = 125.44 \text{ kr.}$$

Additionally, we compute the electricity cost of the HPC. We focus only on the Nvidia A100 GPU and ignore all other components, since their wattage is comparatively not very significant. This GPU has a Thermal Design Power (TDP) of 300 W[6]. The GPU is not always going to be using max power, but we take 300 W as the average power in order to calculate an upper boundary of the real cost.

$$E_{HPC} = \frac{300 \text{ W} \times 150 \text{ h}}{1000} = 45 \text{ kWh}$$

$$\text{HPC electricity cost} = 45 \text{ kWh} \times \frac{2.8 \text{ kr.}}{1 \text{ kWh}} = 126 \text{ kr.}$$

Finally, we compute the electricity cost of the laptop. We estimate the average power usage of the Lenovo Idepad 5 Pro to be 65 W.

$$E_{laptop} = \frac{65 \text{ W} \times 560 \text{ h}}{1000} = 36.4 \text{ kWh}$$

$$\text{Laptop electricity cost} = 36.4 \text{ kWh} \times \frac{2.8 \text{ kr.}}{1 \text{ kWh}} = 101.92 \text{ kr.}$$

Usage	**Time (h)**	**Power usage (W)**	**Cost (kr.)**
Lenovo Ideapad 5 Pro	560	65	101.92
HPC (Nvidia A100)	150	300	126.00
Room lighting	560	80	125.44
Total			**353.36**

Table 7: Summary of electricity costs

[6]According to the specifications section in the product page (PCIe version): https://www.nvidia.com/en-us/data-center/a100/

5.2.4 Summary of generic costs

Table 8 shows a summary of all the generic costs of the project. In total, we plan to spend around 26000 kr. on materials and energy.

Concept	**Cost (kr.)**
Hardware	430.20
Workspace	24000.00
Electricity	353.36
Total	**24783.56**

Table 8: Summary of generic costs

5.3 Budget deviations

5.3.1 Contingency

We plan for unexpected events that may happen during the execution of the project by setting up a contingency fund. We add 15% to the total budget, as shown in table 9.

Concept	**Cost (kr.)**
Staff costs	91909.47
Generic costs	24783.56
Contingency (15%)	17503.96
Total	**134196.99**

Table 9: Total budget with contingency

5.3.2 Incidental costs

We also plan for the risks and obstacles defined in section 3.3, following the mitigation plans defined in section 4.4. Executing these alternative plans has an intrinsic cost, which we consider in table 10.

[7]Calculated as 40 extra hours of project management.

[8]Calculated as 50 extra hours of research.

[9]Calculated as 50 hours of A100 usage on AWS EC2, using a p4d.24xlarge instance: https://aws.amazon.com/ec2/capacityblocks/pricing/

Incident	**Estimated cost (kr.)**	**Risk (%)**	**Cost (kr.)**
Deadline	11538.40[7]	35	4038.44
Unfamiliarity	6731.00[8]	70	4711.70
HPC unavailability	650.50[9]	70	455.35
Total			**9205.49**

Table 10: Summary of incidental costs

5.3.3 Total cost

The total cost of the project is 143402.48 kr., as shown in table 11. This cost has been computed adding the staff costs, generic costs, the 15% contingency fund and the incidental costs.

Concept	**Cost (kr.)**
Staff costs	91909.47
Generic costs	24783.56
Contingency (15%)	17503.96
Incidental costs	9205.49
Total	**143402.48**

Table 11: Summary of total cost of the project

5.4 Management control

Due to the complexity of estimating a task's cost beforehand, it is probable for real costs to be different from the expected values. In order to keep track of the deviations from the budget while executing the project, we will calculate each task's deviation after completion. This allows us to continuously evaluate if we are meeting the budget estimate for each task or if we are exceeding the budget. To do so, we will use the following formula, where t is the total number of tasks:

$$\text{Task deviation} = \text{Real cost} - \text{Estimated cost}$$

$$\text{Total deviation} = \sum_{i=0}^{t} \text{Task deviation}_i$$

Note that if the total deviation is positive, we have exceeded the estimated costs and are spending more money that we anticipated. If it's

negative, we have not reached the budget. Additionally, note that some tasks may have positive deviations, while others have negative deviations. This is not a problem as long as they offset each other and we remain in budget.

6 Sustainability

6.1 Self assessment

The world is evolving towards a more sustainable future, where all human activity is designed to minimize its environmental impact. Lots of companies are striving to achieve economic processes that are carbon neutral, and there is a strong focus on renewable energy generation and responsible energy usage. Other types of sustainability are also important to consider, such as the economic and social impact that a project can have on the population.

In this section, we perform a holistic evaluation of the impact of our work on the environment, our communities and the economy.

6.2 Economic impact

Regarding PPP: Reflection on the cost you have estimated for the completion of the project

All the information regarding cost estimations can be found in section 5, where we estimated the staff and material costs of the project. For staff costs, we considered the average yearly salary for each one of the roles (project manager, programmer, tester, researcher and technical writer), computed the hourly salary, and calculated the total cost for each one of the roles according to the amount of hours that each role was going to be working. A breakdown of assigned hours per role and task can be found in table 4, and a summary of costs per task and the total staff costs can be found in table 5. We also computed the material or generic costs of the project, taking into account devices such as the laptop and the High Performance Computer, for which we compute the amortized cost; and additional costs such as electricity, commute costs and workspace rent. Finally, we add a 15% contingency fund to the budget, and add some extra funds according to the expected impact of the identified potential obstacles and risks in section 3.3.

From my standpoint, I'm impressed at the very high costs, especially when it comes to the staff. Human costs account for more than 85% of the total cost of the project, which I was not expecting.

Regarding Useful Life: How are currently solved economic issues (costs...) related to the problem that you want to address (state of the art), and how will your solution improve economic issues (costs ...) with respect other existing solutions?

As far as we know, a standardized benchmark to compare language identification methods using cross-domain data doesn't exist. Multiple studies have used their own evaluations, but it is often difficult to compare across benchmarks because their setups and datasets are different. A systematic comparison of different methods will allow companies and researchers to use the model that best fits their needs without the need to compare them from scratch, which will save resources. Additionally, we also consider the energy and time costs of the models, which means that there is further potential for savings.

6.3 Environmental impact

Regarding PPP: Have you estimated the environmental impact of the project?

We have not performed a study on the environmental impact of the project. However, the project should not generate any material waste, and the total energy used for training and inference, according to our own calculations, is 45 kWh. According to the International Energy Agency, 81.2% of electricity generation in Denmark comes from renewable sources, such as wind or solar, and 39.52% of final energy consumption comes from modern renewables.[10]

Regarding PPP: Did you plan to minimize its impact, for example, by reusing resources?

We are not using any material resources. However, we are using already developed open source models such as `langid.py`, which saves development time and, consequently, resources such as electricity. Additionally, we save on computation, and therefore electricity, by storing the models and other calculations in storage, instead of recalculating every time. This minimizes the impact of the project on the environment.

[10] As of 2021 and 2022, respectively: https://www.iea.org/countries/denmark/renewables

Regarding Useful Life: How is currently solved the problem that you want to address (state of the art)?, and how will your solution improve the environment with respect other existing solutions?

To the best of our knowledge a standardized and systematic comparison of language identification methods on cross-domain data doesn't exist. The existing solutions are setup-specific and do not generalize well across different methods or domains. Therefore, we consider the problem to be unsolved.

6.4 Social impact

Regarding PPP: What do you think you will achieve -in terms of personal growth- from doing this project?

I will learn about how to properly conduct scientific research and how to write academic papers. Additionally, I will improve my understanding of machine learning and artificial intelligence. I will also venture into Natural Language Processing, which is a topic that I have not explored previously, outside some very specific and simple problems in my Machine Learning course.

Regarding Useful Life: How is currently solved the problem that you want to address (state of the art)?, and how will your solution improve the quality of life (social dimension) with respect other existing solutions?

By creating a systematic comparison of cross-domain language identification methods, we will allow researchers and companies to save time which they would otherwise have to use evaluating the different methods themselves. This will allow them to quickly select the best method for the task, taking into consideration both the accuracy of the methods and their time and power requirements.

Regarding Useful Life: Is there a real need for the project?

Yes, since to the best of our knowledge a standardized benchmark to compare language identification methods using cross-domain data doesn't exist. Multiple studies have used their own evaluations, but it is often difficult to compare across benchmarks because their setups and datasets are different.

List of Tables

List of Figures

References

Baldwin, Timothy and Marco Lui (June 2010). "Language Identification: The Long and the Short of the Matter". In: *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Ed. by Ron Kaplan et al. Los Angeles, California: Association for Computational Linguistics, pp. 229–237. URL: https://aclanthology.org/N10-1027.

Cavnar, William B, John M Trenkle, et al. (1994). "N-gram-based text categorization". In: *Proceedings of SDAIR-94, 3rd annual symposium on document analysis and information retrieval*. Vol. 161175. Ann Arbor, Michigan, p. 14.

Conneau, Alexis et al. (2020). *Unsupervised Cross-lingual Representation Learning at Scale*. arXiv: 1911.02116 [cs.CL]. URL: https://arxiv.org/abs/1911.02116.

Eisenstein, Jacob (June 2013). "What to do about bad language on the internet". In: *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Ed. by Lucy Vanderwende, Hal Daumé III, and Katrin Kirchhoff. Atlanta, Georgia: Association for Computational Linguistics, pp. 359–369. URL: https://aclanthology.org/N13-1037.

Imani, Ayyoob et al. (July 2023). "Glot500: Scaling Multilingual Corpora and Language Models to 500 Languages". In: *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Ed. by Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki. Toronto, Canada: Association for Computational Linguistics, pp. 1082–1117. DOI: 10.18653/v1/2023.acl-long.61. URL: https://aclanthology.org/2023.acl-long.61.

Lui, Marco and Timothy Baldwin (Nov. 2011). "Cross-domain Feature Selection for Language Identification". In: *Proceedings of 5th International Joint Conference on Natural Language Processing*. Ed. by Haifeng Wang and David Yarowsky. Chiang Mai, Thailand: Asian Federation of Natural Language Processing, pp. 553–561. URL: https://aclanthology.org/I11-1062.

— (July 2012). "langid.py: An Off-the-shelf Language Identification Tool". In: *Proceedings of the ACL 2012 System Demonstrations*. Ed. by Min Zhang. Jeju Island, Korea: Association for Computational Linguistics, pp. 25–30. URL: https://aclanthology.org/P12-3005.

Mikolov, Tomas et al. (2017). *Advances in Pre-Training Distributed Word Representations*. arXiv: 1712.09405 [cs.CL]. URL: https://arxiv.org/abs/1712.09405.

Penedo, Guilherme et al. (2023). "The RefinedWeb Dataset for Falcon LLM: Outperforming Curated Corpora with Web Data, and Web Data Only". In: arXiv: 2306.01116 [cs.CL]. URL: https://arxiv.org/abs/2306.01116.

Toftrup, Mads et al. (Apr. 2021). "A reproduction of Apple's bi-directional LSTM models for language identification in short strings". In: *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop*. Ed. by Ionut-Teodor Sorodoc et al. Online: Association for Computational Linguistics, pp. 36–42. DOI: 10.18653/v1/2021.eacl-srw.6. URL: https://aclanthology.org/2021.eacl-srw.6.

van Noord, Gertjan (1997). *TextCat Language Guesser*. Accessed: 29 october 2024. URL: https://www.let.rug.nl/vannoord/TextCat/.

Vaswani, Ashish et al. (2017). *Attention Is All You Need*. arXiv: 1706.03762 [cs.CL]. URL: https://arxiv.org/abs/1706.03762.