Research and Innovation Action (RIA) H2020 – 957017

SELMA

Stream Learning for Multilingual Knowledge Transfer

https://selma-project.eu

D2.4 Intermediate Progress Report on Continuous Massive Stream Learning

Work Package	2
Main Author	Afonso Mendes
Co-Authors	Diogo Pernes, João Figueira, João Santos, Tugtekin Turan
Reviewer	Yannick Estève
Version	1.0
Contractual Date	December 31, 2022
Delivery Date	December 22, 2022
Dissemination Level	Public

Version History

Version	Date	Description
0.1	03/11/2022	Initial Table of Contents (ToC)
0.2	15/11/2022	Initial Input
0.3	19/12/2022	Internal Review
1.0	22/12/2022	Publishable version

Executive Summary

This report is an incremental version of the previously delivered document describing the scientific advances of our natural language tasks under work package two (WP2), in the SELMA project. WP2 is aimed to enable the SELMA system to learn automatically from a large-volume live multilingual stream of data. This document is structurally identical to the previous version, with incremental changes wherever applicable. As such, we report the advancements of our work on each of the sub-tasks required to achieve WP2 targets: story segmentation, entity linking, named entity recognition, news classification, clustering, and summarization. The first two chapters will introduce the general framework and overview of the WP2, particularly introducing each sub-task separately. In the following chapters, we define our methodologies for each task in Section 3 and present our experimental results in Section 4. The last section will conclude this report by emphasizing future studies and user feedback ideas as well.

Table of Contents

Table of Figures

Table of Tables

1. Introduction

Continuous learning aims to enable information systems to learn from a continuous data stream across time. We, as human beings, can learn by building on our memories and applying past knowledge to understand new concepts. However, it is not easy for existing deep learning architectures to learn a new task without forgetting previously acquired knowledge. Unlike humans, existing machine learning ideas are primarily trained in an isolated environment and can be used effectively only for a limited time. Therefore, the produced models become less accurate over time due to the changing distribution or nature of the data. With the recent advancements in deep learning, the problem of continuous learning in natural language is becoming even more critical, as current approaches cannot effectively keep previously learned knowledge and adapt to new information simultaneously.

The SELMA continuous learning platform specifically targets multilingual broadcast monitoring and production. With the exponential growth of online news content in several languages, the challenge is to avoid a language and cultural bottleneck. Hence, this work package eventually brings together many sources and makes information accessible to users in multiple languages, yet keeping relevant knowledge present in the original multilingual data sources.

Multilingualism supports the opportunity of sharing valuable knowledge across languages. We, therefore, aim to propose a unified approach to multilingual media monitoring and content production by contributing to recent advances in deep learning, particularly breakthroughs in knowledge and language transfer and fine-tuning of task models from user feedback. High-quality and up-to-date cross-lingual text and entity representations are vital components of this work package. Computing and updating these representations via user feedback is an important research direction in the context of natural language on news data, as relevant entities, which have a defining role in news stories, take part in ever-evolving story contexts.

To this end, this work package will create a high-performant modular platform for the ingestion and processing of data streams with the goal of training and maintaining multilingual natural language components. In this deliverable, we present our initial results based on the defined targets. Our proposed methods will finally create a distinctive setting for integrating high-quality user feedback

with massive amounts of data using stream learning techniques. Low-resource languages will also be addressed owing to the multilingual data context combined to transfer learning approaches.

2. Architecture

This work package enables the SELMA platform which learns automatically from a large-volume live multilingual stream of documents and continuously incorporate knowledge to update the models. Moreover, transfer learning will be investigated to improve scarce languages with knowledge from high-resource languages.

The multilingual stream will be combined with the SELMA processing pipeline. A collection of news sources will serve as a reference to guide the natural language downstream tasks executed on the user-supplied data. We mainly research novel approaches to jointly extract named entities from the reference stream and link them to a knowledge base to enable the proposed methods. We also employ current practices to learn up-to-date contextual cross-lingual embedding representations for text/entities and efficiently search on these representations.

In summary, it is possible to define these main goals for this work package,

- Learning a representation for text and entities from the input reference stream
- Identifying named entities and linking them to a knowledge base
- Incorporating the user feedback into training and improvement of our models
- Transferring knowledge between languages, whereas benefiting low-resourced languages

To achieve these goals, we can define the primary tasks of this work package as in the following:

Cross-lingual Stream Representations

This task focuses on learning contextual word and entity representations captured from a live news article stream. Note that the extensive data scale makes this task particularly challenging, in addition to the emphasis on serving across several languages simultaneously. Hence, to enable knowledge transfer from higher- to lower-resourced languages, we aim to learn a cross-lingual representation space, i.e., a representation where word contexts from different languages are mapped into a shared space, to enable knowledge transfer from higher- to lower-resourced languages. Furthermore, unlike

other approaches that rely on dictionaries' cross-lingual training contexts, we seek to learn and incrementally update a cross-lingual representation specially geared towards the current and most relevant news content, focusing on the changing named entity representations.

Named Entity Recognition and Linking

This task aims to develop statistical models for detecting entities within news article streams and learning a mapping of these entities to a knowledge base link. This step is fundamental to performing content enrichment on the data stream. Therefore, we focus on deep contextualized representations and approach this problem under end-to-end architecture where we perform entity disambiguation and obtain the correct link. It is also essential to investigate the novel issue of incorporating a multi-task learning approach over recent neural models. This sub-problem emphasizes context-dependent entity linking, which shares some ambiguity due to the polysemous nature of the entity, and primarily due to time-dependent context. Thus, we will focus on discovering new entities from the news stream, attribute unique knowledge-based IDs, and link further mentions of these entities together. A significant contribution of this work package will utilize zero-shot and transfer learning approaches to disambiguate and link new entities. We draw inspiration from relation extraction models and explore the entity co-location approach.

Story Segmentation

This task aims to segment long audio segments into meaningful units, providing speaker clustering, speaker recognition, and topic segmentation. For speaker clustering, the identity of the speakers is unknown, and the system provides only labels for segments of the same speaker appearing multiple times in one file. This is useful, e.g., for interviews in which only statements of a particular user are of interest to the journalist. Moreover, the human voice can contain personal attributes of unique pronunciation (vocal tract shape) and speaking manner (accent and rhythm). Therefore, speaker recognition is defined as the task of identifying persons from their voices. We approach this problem with an end-to-end framework for the recognition of specific speakers from a known speaker database. On the other hand, we investigate the speaker diarization task to label news content with classes that correspond to speaker identity in order to address "who spoke when".

Online News Classification and Clustering

News classification targets categorizing a given text sequence with one (or multiple) pre-defined class label(s) describing its semantic content. To this end, we will follow recent research on cross-lingual representations for topic labeling across various languages, which uses deep contextualized models. One of our primary concerns addresses the problem of learning a shared space for different label sets on multilingual data. Therefore, it is crucial to focus on different shared space architectures and attention mechanisms to cope with multilingual datasets—moreover, online news clustering groups semantically similar text streams without supervision or manually assigned class IDs. Similar to the online classification task, we plan to automatically cluster input documents in the cross-lingual space, such that documents from different languages can be aggregated together according to the story topic.

News Summarization

This task focuses on summarizing news content using state-of-the-art abstractive neural approaches. The biggest challenge for this task is the presence of factual inconsistencies in the generated summary. Both the automatic detection and the mitigation of factual inconsistencies are open research problems on which we will focus. Additionally, and since current research and available models are mostly English-centric, we will extend our research to the multilingual and cross-lingual scenarios.

In a parallel line of work, we are also addressing the challenging problem of end-to-end speech-to-text summarization. This task has been seldom explored in prior work, which mostly addresses it using a cascade of transcription and text summarization modules. We believe an end-to-end approach may yield better results by avoiding error propagation and being computationally more efficient.

At the final stage, we will evaluate our methods in the context of summarizing video audios by using the video teasers as the target summaries. We will compare the end-to-end speech-to-text approach with a cascaded model using a publicly available ASR module followed by our abstractive (text-to-text) approach.

3. Scientific Approach

This section presents an introduction to our proposed methodologies employed for this deliverable. The problems presented in the introduction section will be explained in detail under the following sub-chapters. We will give experimental results and their discussions in the next section.

3.1. Named Entity Recognition

For the named entity recognition (NER) task, we investigate two ideas: hierarchical NER and example-based NER. In the following subsections, we present a summary of these approaches. During the second reporting period we investigated the behavior of the proposed models in a multilingual scenario and their ability to zero-shot to unseen languages during training.

Hierarchical Nested NER

The task of recognizing entities can take different forms. We focus on the hierarchical nested approach, where a given sequence of words can correspond to more than one entity, e.g., “gpe” and “gpe → city”, with “city” being a more fine-grained entity type, with the added possibility of including nested entities. This subsection reports two approaches related to the task of hierarchical nested NER: improvements made to Marinho et al. (2019) (Stack-LSTM), and a new biaffine approach, heavily based upon Yu et al. (2020).

Stack-LSTM work, proposed by Marinho et al. (2019), models hierarchical and nested entities via four main actions: transitions, shifts, reduction, and outs. These actions modify the system's state by interacting with the words in an input sentence over a series of "stacks", which model different aspects using LSTMs. All words are represented by concatenating their corresponding fixed-word lookup embedding and learned character sequence embedding representations. We propose replacing the original word representations by contextual embedding representations, using existing models based on architectures such as BERT (Devlin et al. (2019)), coupled with an extensive study of pooling approaches and fine-tuning strategies. The main advantage is to use more powerful pre-trained embedding models, which can leverage the context of a word within its sentence. Several works highlight the excellent performance of applying pre-trained multilingual contextual embedding to languages other than English.

Biaffine model follows the work of Yu et al. (2020). This model scores pairs of start and end tokens in a sentence to explore all spans so that the model can predict named entities accurately. We propose using a biaffine classifier model, initially capable of identifying flat and nested entities. It uses token-level representations based on a combination of character and pre-trained contextual embeddings coupled with a biaffine model. This returns a score tensor of every possible class of start-end span combinations. It has dimensions *n* x *n* x *c*, where *n* is the number of tokens in the input, and *c* is the number of classes plus one, the no-entity class. We introduce three changes to make this approach capable of modeling hierarchical entities: (i) the score tensor, which is an output of the biaffine model, is now *n* x *n* x *m*, where *n* corresponds to the number of tokens in the input, and *m* corresponds to a span embedding dimension; (ii) we add a classifier that predicts whether a span corresponds to an entity or not. The intuition is that since predicting multiple labels for each span will involve evaluating all possible spans sequentially, skipping as many spans as possible improves performance; (iii) using the score tensor, we use an LSTM model to predict entities for a given span at a time, until the "end of the sentence" token is predicted. At each step, the LSTM model input becomes the concatenation of different intermediate representations.

Example-Based NER

Current research in text generation has shown that combining a traditional generation model with a k-nearest neighbors (kNN) approach improves performance (Khandelwal et al. (2020), Khandelwal et al. (2021)). We explore the possibility of extending these approaches to the NER task. In particular, for each token of the input sentence, we find the closest *k* tokens on a set of similar sentences retrieved using sentence embeddings (SBERT) (Reimers et al. (2019)). Then, we follow either a single-*k* approach, where the kNN distribution for each token is obtained from a single *k* value, or a multi-*k* approach, where the kNN distribution for each token is the average of the distributions obtained for multiple *k* values. The remaining steps follow the works mentioned above.

We highlight the possibility of using this approach to leverage user feedback by continuously adapting the NER predictions with the data collected, avoiding re-training the model as often. We aim to use this approach to deal with user feedback for entity linking and the NER from speech.

Learning Cross-language NER

In SELMA, one of our main objectives is to obtain good models over a large number of languages. Additionally, one of our major concerns is scalability when moving the models to production, which means that we cannot deploy one model for each of the SELMA target languages. To attain these objectives, we researched the possibility of having only one model for all the target languages without losing performance. We also wanted to know if it is possible to improve each of the languages by using data from other languages.

To meet the above objectives, we researched the following approaches keeping in mind that our aim, to keep platform-needed resources to a minimum, is to have one model that covers all languages:

- Training one joint model using as training data the mixture of all language datasets annotated with the common ontology making no explicit difference between them.
- Introduce an additional language class, and their respective *transition and reductions,* on the stack model representing the language of the input document.
- Use of language adapters that can be trained and injected in the model for each language as proposed by Neil Houlsby et al. (2019).

In the evaluation section, we report the results of the first hypothesis above, which proved very good and more general than the remaining.

3.2. Entity Linking and Cross-lingual Stream Representations

Entity linking is the task of connecting a named entity in a document to an entry in a knowledge base (KB). One way to address this problem is to create a candidate set for each named entity with possible entities from the KB and then rank the candidates to choose the most likely entity to be linked. Our work follows this approach and employs a model inspired by dynamic context augmentation (DCA) by Yang et al. (2019), which is itself an improvement over the original model proposed by Ganea and Hoffmann (2017). This family of models has two main components: the pre-trained entity embeddings and the ranking model (based on the DCA) that uses those embeddings and scores candidates through a combination of independent scores. This formulation allows for an existing

subset of entities to be adapted or added without retraining the whole set of entities in the knowledge base, facilitating user feedback and stream learning scenarios.

Our entity embeddings are bootstrapped from a frozen set of word embeddings. Following the idea in Yang et al. (2019), we employ Wikipedia canonical pages and hyperlinks. The original articles leveraged English Word2Vec (Mikolov et al. (2013)) embeddings. We extended the model to use multilingual word embeddings such that non-English embeddings and texts can now be used. BPEmb embeddings (Heinzerling & Strube (2018)) contain sub-word embeddings based on Byte-Pair encoding (BPE) in 275 languages, trained on Wikipedia. BPE is a compression algorithm that, in an NLP context, allows the representation of words by the set of most common sub-words, removing the need for out-of-vocabulary tokens. This ability, in conjunction with the extensive language set available and its improvement on performance with relation to Word2Vec, led us to use these embeddings.

The ranking model DCA receives a pre-computed candidate set for each mention and yields a score for each candidate, choosing the highest scoring candidate as the linked entity. This score is a composition of independent scores. Yang et al. (2019) model considered only three scores: (i) prior probability, $P(E|m)$, computed using Wikipedia hyperlink count frequency; (ii) a local disambiguation score that calculates an attention score between a candidate embedding and word embeddings surrounding the mention to assign higher importance to certain context words; (iii) a global entity coherence score to produce an attention score between a candidate and previously disambiguated entities, under the assumption that there is consistency between document mentions. We extend this model to consider two other scores based on entity types and mention candidate similarity. The former generates the cosine similarity between the predicted type embedding of a mention and the type embedding of a candidate. The mentioned type is inferred using a classifier following Cardoso et al. (2020), trained alongside DCA. The latter is an alternative way to leverage global coherence by comparing candidates of mentions. We compute the cosine similarity between a given mention's candidate embedding and all the candidate embeddings of neighboring mentions and take the maximum similarity across all neighbors as the score for that candidate.

Our learned embeddings vocabulary can consider exclusively English, German, and Portuguese Wikipedia pages. In a multilingual scenario, for a given entity, we will sample positive words from

the English Wikipedia page and hyperlinks if that entity has an English Wikipedia page. Otherwise, we will sample positive words from the respective language from which it was obtained, either German or Portuguese. To train the DCA model, we used different configurations: training on the English portion of the CoNLL-2003 (Tjong Kim Sang & De Meulder (2003)) NER shared task data, containing news stories from Reuters news agency; training on an English Wikipedia page set where hyperlinks are considered mentions and their linked pages are the gold entities; training on both sets simultaneously.

During the second reporting period we started investigating how the previous approaches behaved when we extended the number of languages to a much bigger set of languages (currently 40). We found out that the models behave similarly when training the entity representations using these additional languages. Our current objective is to have a single model for EL and NER trained jointly or at least sharing the same base contextual model. For that purpose, we started researching the possibility of using the base contextual model fine-tuned on the NER data to learn the entity embeddings. This implies changing the Ganea model presented above to gather negative token embeddings from the Wikipedia multilingual dataset, posing a big optimization challenge due to the size of the dataset.

To train the contextual entity embeddings, we defined the following procedure, based on the procedure from Ganea and Hoffmann (2017):

1. Initialize the entity embeddings with the mean pooling of the individual token embedding of the entity title.
2. Obtain a first set of entity embeddings by training the maximum margin model using the Wikidata data for each entity in each of the languages available: label, description, title (we intended to extend this with other Wikidata properties, e.g subclass of, type, etc..). For this we will experiment using only the CLS token, the mean pooling of the token embeddings, or the individual embedding of each of the tokens. The negative samples are collected by a random permutation of all the positive embeddings.
3. Do the same procedure as above using the concatenation of all the Wikipedia language pages for each entity as positive examples.

4. Do the same procedure as above using as positives the Wikipedia contexts where there is a link to an entity.
5. A final optional step is to further train the entity embeddings using as positives all the entities that co-occur as link with another entity.

On a second phase, we will investigate if these embedding are suitable and how to use them for the downstream task of Entity Linking.

In the evaluation section, we report preliminary results obtained on the quality of the entity embeddings.

3.3. Story Segmentation

Human voice has a personal identity that may offer biometric security by combining physiological and behavioral characteristics (Lu et al. (2017)). Driven by a great deal of potential applications in story segmentation, automated systems have been developed to automatically extract the different pieces of information conveyed in the speech signal. Hence, several tasks could be defined under the speaker recognition problem. They differ mainly with respect to the decision type that is required for each task. In speaker identification, a voice sample from an unknown speaker is compared with a set of labeled speaker models (Tirumala et al. (2017)). The label of the best matching speaker is taken to be the identified speaker. In a speaker verification task, an identity claim should be provided or asserted along with the voice sample (Nagrani et al. (2020)). The unknown voice sample is compared only with the speaker model whose label corresponds to the identity claim.

A more challenging task is generally referred to as speaker diarization which is used to answer the question of "who spoke when?" (Wang et al. (2018)). Throughout the diarization process, the audio data would be divided and clustered into groups of speech segments with the same speaker identity/label. A complicating factor for this task is that the input news stream may contain speech from more than one speaker. Thus, speaker diarization is regarded as the combination of speaker segmentation and speaker clustering. The first aims at finding speaker change points in an audio stream and the second aims at grouping together speech segments on the basis of speaker characteristics.

In our initial experiments, we only investigate recognition tasks. Specifically, we focus on text-independent speaker recognition when the identity of the speaker is based on how the speech is spoken, not necessarily in what is being said. Typically, such a system operates on unconstrained speech utterances, which are converted into vectors of fixed length, called speaker embeddings.

Recently, x-vector-based architectures attained state-of-the-art results on speaker-related tasks (Snyder et al. (2018a)). The development of time-delayed neural networks (TDNNs) topology is still an active research area in speech processing. The preferred approach is to train neural networks on the speaker classification task. After the model convergence, low-dimensional embeddings are extracted from the bottleneck layer before the soft-max output. Speaker recognition can be completed by comparing the two embeddings over a cosine distance measurement to accept or reject a hypothesis that both samples contain the same speaker. Additional complex backend scoring can also be utilized for this task, such as probabilistic linear discriminant analysis (PLDA) (Ioffe (2006)).

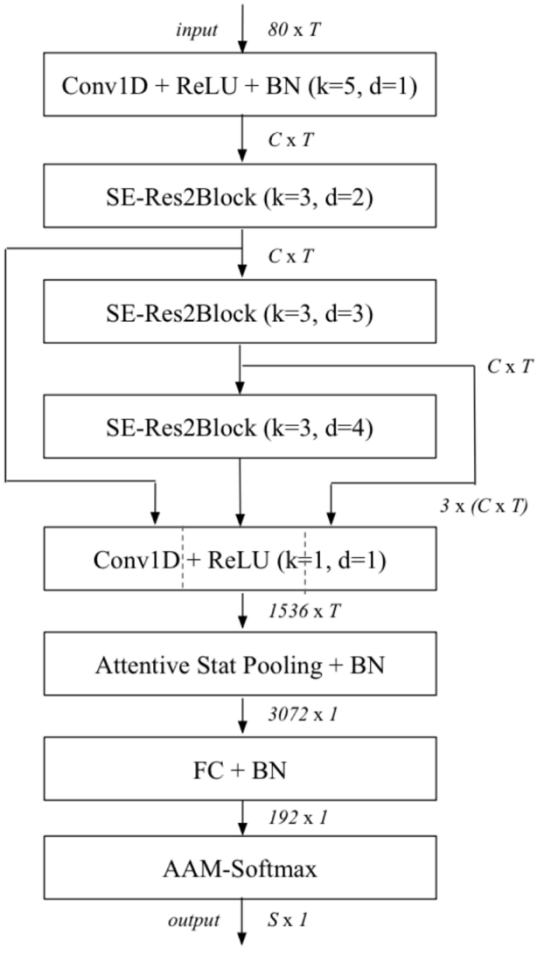

Figure 1 *Network topology of the ECAPA-TDNN (Desplanques et al. 2020) embedding extractor where BN stands for batch normalization and the non-linearities are rectified linear units (ReLU)*

The statistics pooling layer in the x-vector system can map the variable-length input into a fixed-length representation by gathering temporal statistics of hidden layer activations. Okabe et al. (2018) introduced a self-attention system to the statistical pooling, focusing more on essential frames. This model is then improved by adding elements of ResNet architecture (He et al. (2016)). The residual connections of ResNet between the frame-level layers can enhance the x-vector embeddings. Moreover, these residual connections improve the backpropagation in terms of faster convergence and prevent the vanishing gradient problem (Snyder et al. (2018b)).

In this deliverable, we follow ECAPA-TDNN (Desplanques et al. (2020)) architecture which can eliminate some limitations of the x-vector embeddings. This new model extends the temporal attention mechanism even further to the channel dimension. It enables the network to focus more on speaker characteristics that do not activate on identical or similar time instances. An overview of the complete architecture is given by Figure 1 where k and d represent kernel size and dilation spacing of the network layers. C and T correspond to the channel and temporal dimension of the intermediate feature maps, respectively, and S is the number of training speakers/users.

Channel- and context-dependent attention mechanism are implemented inside the pooling layer, which allows the network to attend different frames per channel. The temporal frame context in the original x-vector model is limited to *15* frames (Garcia-Romero et al. (2019)). As the model benefits from a broader temporal context, it is possible to rescale the frame-level features given global properties of the input sample, similar to the global context in the attention modules. Therefore, 1-D squeeze-excitation (SE) blocks (Hu et al. (2018)) rescale the channels of frame-level feature maps to insert global context information inside the locally operating convolutional blocks.

Regular residual blocks (ResBlocks) make it easy to incorporate advancements concerning computer vision architecture (He et al. (2016)). The recent Res2Net module enhances the central convolutional layer such that it can process multi-scale features by constructing hierarchical residual-like connections within (Gao et al. (2019a)). Thus, integrating 1-D SE-Res2Block improves performance while simultaneously reducing the total parameter count by hierarchically used grouped convolutions.

At the last stage, multi-layer feature aggregation (MFA) merges complementary information before the statistics pooling by concatenating the final frame-level feature map with intermediate feature maps of preceding layers (Gao et al. (2019b)). The overall network is trained by optimizing the AAM-soft-max (Deng et al. (2019)) loss on the speaker labels of the training data. The AAM-soft-max is an enhancement compared to the traditional soft-max loss in the context of fine-grained classification problems. It directly optimizes the cosine distance between the speaker embeddings. In this way, complex scoring backends, like PLDA, can be avoided.

3.4. Online News Classification

For the classification of online news, Priberam has worked with the taxonomy established by the International Press Telecommunications Council (IPTC), a consortium of the world's major news agencies. The IPTC Subject Codes vocabulary and the succeeding Media Topics vocabulary establish a hierarchical system of labels to describe the topics covered by any media document. In our experiments, the subject codes vocabulary has been used to classify news articles, and it covers 1404 labels of topics distributed over a hierarchy of three layers. Label names and descriptions are included in seven languages (English, German, French, Portuguese, Spanish, Italian, and Japanese).

Using a dataset of Portuguese news provided by the Lusa News Agency[1], Priberam has trained models for news classification in this taxonomy. The dataset includes over 700,000 news articles in Portuguese for training and testing and an additional 1,000 articles in Spanish and English, each provides a general sense of the cross-lingual performance of the model.

An additional dataset of news articles was acquired from the STT Finnish News Agency[2], the dataset includes over 900,000 news articles in Finnish, labelled with IPTC labels. We use this dataset along the Lusa dataset to train our models in a broader topic space, and to help multilingual models not overfit to a single language. As a Uralic language, Finnish is lexically very distant to Portuguese.

[1] Lusa Agency of Portugal: https://www.lusa.pt/lusanews

[2] STT Finnish News Agency: https://stt.fi/en/

Previous approaches to this task by Priberam used a model described in report D5.1 of the SUMMA project[3], which used convolutional neural networks (CNNs) to aggregate word embeddings to make a final decision through a fully connected layer. Separate versions of this model made decisions at each step of the label hierarchy. For the model to cover languages outside the training set, the FastText (Bojanowski et al. (2017)) multilingual word embeddings were used. The FastText word embeddings were initially published by Facebook research as separate sets of monolingual embeddings for 89 languages, these were later aligned by researchers at Babylon Health into a single set of multilingual embeddings. This allows the model to infer on zero-shot languages. These embedding vectors were not fine-tuned in training, which avoids corrupting the word embeddings of languages not seen during training.

One of the main focuses of the news classification task is to improve the performance of Priberam's news classifier. Firstly, by finding a lighter model that can predict the entire label hierarchy in a single forward pass. And secondly, by leveraging the new developments in NLP model architectures, namely models such as bidirectional encoder representations from transformers (BERT) (Devlin et al. (2019)), that can be pre-trained in a multilingual context and then fine-tuned for the specific task using the monolingual dataset.

Chalkidis et al. (2020) performed a thorough survey on the hierarchical multi-label classification of text and showed the outstanding performance of transformer type models. A significant drawback of these models is the limited input size that requires some news articles to be shortened.

[3] SUMMA Deliverable D5.1: http://summa-project.eu/wp-content/uploads/2017/08/SUMMA_D51_InitialNLU.pdf

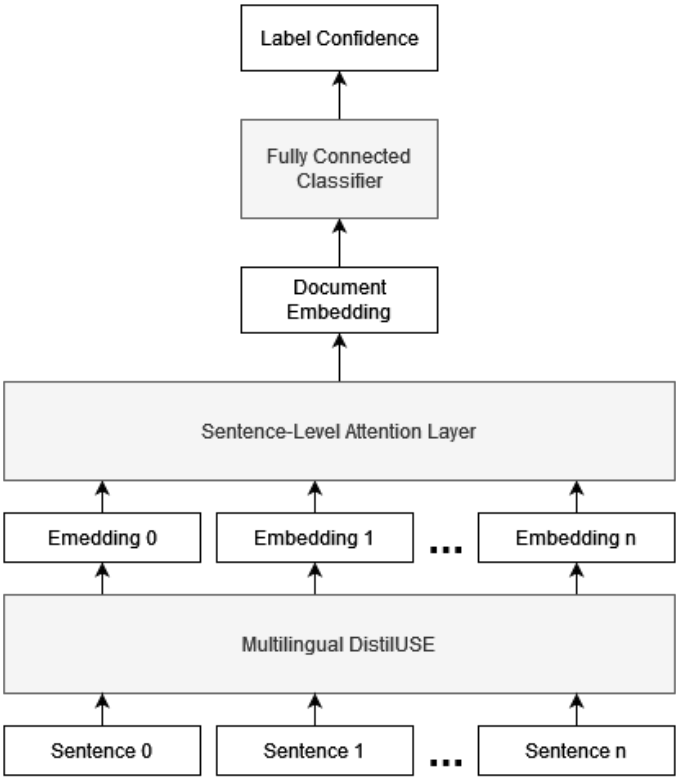

Figure 2 *Architecture of sentence embeddings-based classification models where the novel sentence-level attention layer can take queries from various sources, and outputs an embedding*

Our first proposed model uses multilingual sentence embeddings produced by a DistilUSE (Reimers et al. (2019)) model to represent an entire news article as a sequence of sentence embeddings. DistilUSE is a transformer-type model trained as a more lightweight multilingual counterpart to a monolingual teacher model (using knowledge distillation). This model is trained to generate sentence embeddings in a shared multilingual space. In our new proposed architecture, an attention layer is used to estimate the importance of each sentence embedding and aggregates them for a final decision in a fully connected layer. We further expanded on this practice by experimenting with separate attention queries for each label and particular attention queries for each hierarchy depth. The general architecture of these models is shown in Figure 2.

Our second proposed model is based on an attention-aware model called AttentionXML (You et al. (2019)), which has shown remarkable performance in use-cases of extreme multi-label classification. AttentionXML works by allowing each candidate label to query directly on the word embeddings.

The result of this attention layer is fed to a fully connected binary classifier that is shared between all labels.

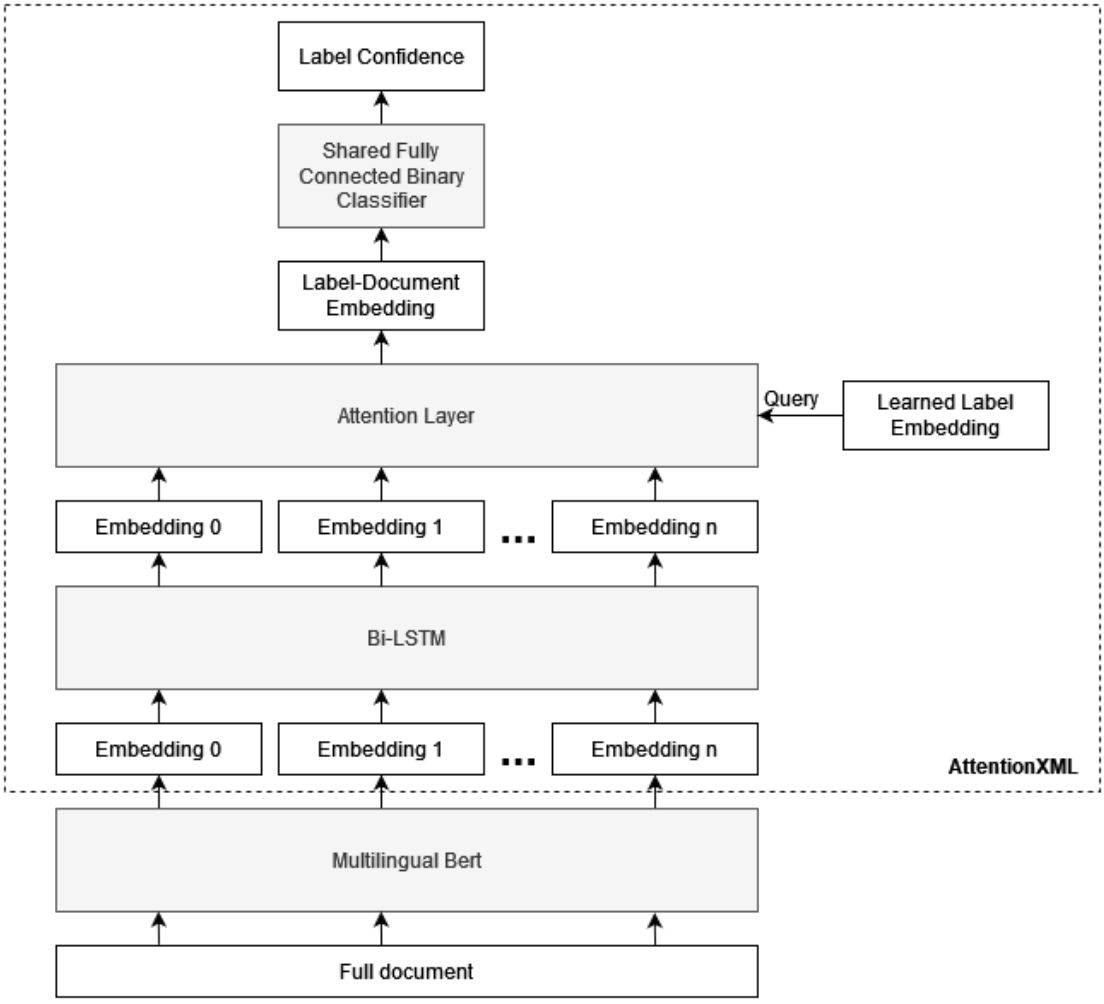

Figure 3 *Architecture overview of mBERT and AttentionXML hybrid models, the top dashed box shows the architecture of a stock AttentionXML*

Each label learns its own query, which finds the most relevant words. The final classification layer is trained on identifying if the document has the most attention on the words that are relevant to its topics. The major drawback of this model is its non-reliance on pre-training and the lack of multilingual support. We explore two modifications to this model, both aimed at making it multilingual. Firstly, we experiment replacing the word embeddings with pre-trained multilingual word embeddings, and we chose the BPEmb (Heinzerling & Strube (2018)) embeddings for this.

These are subword embeddings trained with byte pair encoding that outperform FastText in some scenarios. The authors have open-sourced BPEmb embeddings and tokenizers for 275 languages, along with a multilingual version that covers all 275 languages. Secondly, we try replacing the entire embedding layer with a transformer model. For this, we used a multilingual BERT to provide contextual embeddings for each token that serves as input to AttentionXML. This allows our embeddings to be more contextualized than what can be achieved with the default biLSTM and will enable us to partially finetune the mBERT model, improving its accuracy for the task without sacrificing the multilingual performance. We later run similar experiments with a pretrained multilingual Roberta-Large model (Liu et al. (2019)), which has shown great potential for multilingual NLP tasks. The architecture of these latter models is shown in Figure 3.

Improving Explainability of AttentionXML-Based Models

When initially proposing the architecture of AttentionXML, the original authors boast about how the model provides a simple explanation for model decisions since it has a single Token to Label attention layer. And that the attention values from this layer provide a score of how relevant each word token is for each label decision.

In our analysis of these attention distributions, we found many examples where the attention peaks were not on the relevant tokens, but instead other tokens in the neighborhood of these relevant tokens. We speculate that the BiLSTM of the model can aggregate information in the contextual embeddings near to the relevant text spans, and that some arbitrary embedding might be sufficient for the model to make a decision. To minimize this effect, we experiment with splitting the BiLSTM, into a separate Forward-LSTM and Backward-LSTM. The reasoning for this is that, since the LSTMs can accumulate relevant information onto arbitrary tokens in the neighborhood of the relevant sections, and that these tokens will later be favored by the attention layer, using separate LSTMs with different directions will restrict the positions at which these tokens will be found. With the Forward-LSTM, we can guarantee that the relevant information can only be accumulated on a token at the end of, or to the right of, the relevant section. Similarly, with the Backward-LSTM, we can guarantee that the relevant information can only be accumulated on a token at the start of, or to the left of, the relevant section. This way it is possible to find relevant spans delimited by the high attention tokens of these two models.

We also experimented with the attention layer of the mBert and AttentionXML hybrid model. Here we found that the attention weights were seemingly very random. Given the size of the transformer-type models, it is perhaps not valid to think of the output embeddings as contextual embeddings of the corresponding input tokens, in the same way that the output embedding of the [CLS] token is used as an embedding of the entire document.

Experiments with ICD coding

Multi-label classification of medical documents regarding diagnosis and procedures described within medical records is a popular task and benchmark for the models described here, due to its necessity and applicability in hospitals. The International Classification of Diseases (ICD)[4] is a globally used labelling schema, maintained by the World Health Organization (WHO). Due to the very large label space of ICD, and its somewhat hierarchical nature, ICD classification of medical documents is a similar task to IPTC classification. We experiment our models on the widely used MIMIC dataset (Johnson et al. (2016)), which is labelled according to the ICD9 version of the classification standard.

3.5. Online News Clustering

Our primary focus for the news clustering task is to build an online multilingual news clustering system that could process and organize articles from most SELMA languages[5]. In this task, a continuous stream of incoming news articles must be organized into clusters of events called stories. Miranda et al. (2018) approached this problem by processing the news documents stream into monolingual and cross-lingual clusters. Each document is first associated with a monolingual cluster using the term frequency-inverse document frequency (TF-IDF) sub-vectors of words, lemmas, and named entities. Then, cross-lingual clusters are computed by linking different monolingual clusters through cross-lingual word embeddings weighed with TF-IDF. While this approach obtained good results at the monolingual level, it had the following drawbacks: the cross-lingual word embeddings

[4] International classification of diseases (ICD): https://www.who.int/classifications/classification-of-diseases

[5] SELMA platform target languages: Albanian, Arabic, Bulgarian, Chinese, Croatian, English, French, German, Greek, Hindi, Indonesian, Macedonian, Persian, Polish, Portuguese, Romanian, Russian, Serbian, Spanish, Turkish, Ukrainian, Urdu.

did not take their neighboring words (and thus, the context of the sentence) into account, and the monolingual step required training a separate model for each language as well as extracting the entities from the given text, a task that can be problematic for low-resource languages.

For our approach (Santos et al. (2022)), we developed a system that can cluster news articles of any language without depending on language-specific features while being supported by pre-trained multilingual contextual embeddings. For a given document, our system is composed of four main steps: (i) obtaining its document representations, (ii) finding the best-ranked cluster for that document, (iii) deciding if the document accepts the best-ranked cluster and enters it, and (iv) merging clusters that pertain to the same story. A representation of our clustering system is depicted in the following figure.

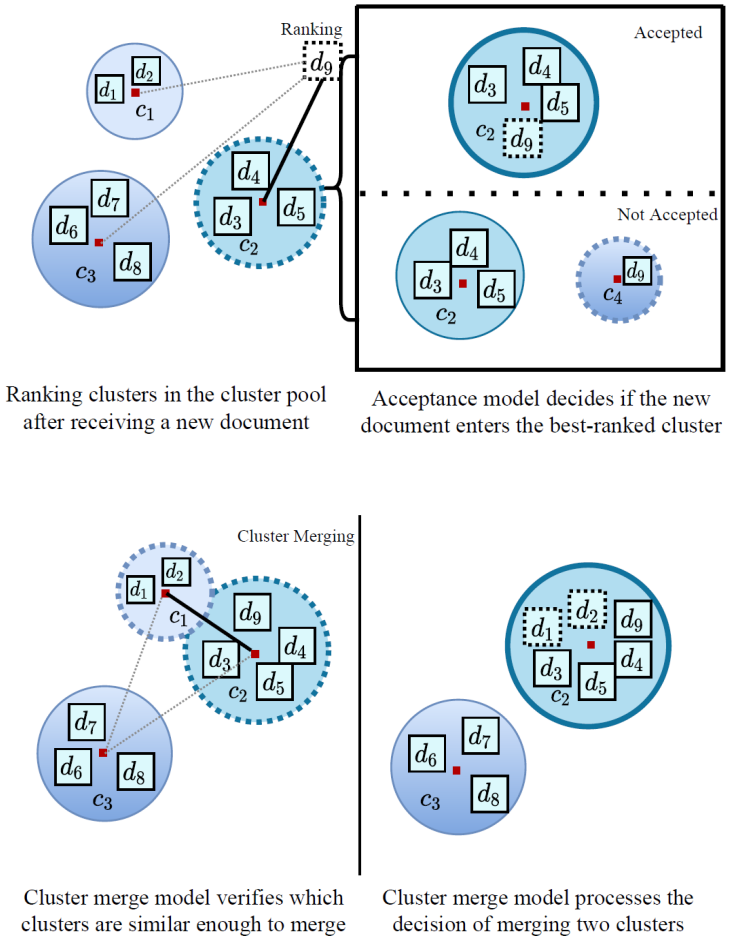

Figure 4 *Representation of the news clustering system's ranking, acceptance and merge steps*

To represent news documents and clusters, we focused our efforts on composing a contextual representation in a set of dense vectors. To that end, similarly to the news classification task, we use DistilUSE, a pre-trained model that aligns text at the sentence level into a shared semantic space, resulting in similar sentences being closely mapped in the vector space regardless of their language. This model supports over 50 languages and does not require the specification of the input language, providing a vectorial representation for the documents that can then be used to inference and group similar news articles. This is a significant change from previous approaches, as contextual information was not taken into account at a cross-lingual level in news clustering state-of-the-art (Miranda et al. (2018), Linger et al. (2020)). Additionally, this approach simplifies the clustering task by using a single cross-lingual representation for the documents, thus allowing for a fully dense clustering space.

Documents are comprised of two components: a set of dense vectors d^r corresponding to a contextual representation of the document, and a temporal representation (d^{ts}). For each document, d^r contains three dense representations: d^r_1 corresponds to its body and title, d^r_2 to its first paragraph, and d^r_3 to its first paragraph and title. Each of the output vector representations is obtained by mean pooling. Regarding the temporal representation, we follow previous approaches (Miranda et al. (2018)) and expose the temporal representation d^{ts} of a document as the value of its timestamp in days.

In order to find the best-ranked cluster for a given document, we trained a Rank-SVM model, which is a variant of the support vector machine (SVM) algorithm, using a news clustering dataset (Rupnik et al. (2016)) with dense and temporal features. Given the training partition of the dataset, each document generates a positive example corresponding to its gold cluster, and 20 negative examples for the 20 best-ranked clusters that are not the gold cluster.

These examples are then used in the Rank-SVM to obtain a set of fixed weights for each of the features. Temporal features are computed through the Gaussian similarity between two timestamps (represented by the $score^{ts}$ function, and the dense features are obtained through the computation of the cosine similarity ($score^{cos}$). The ranking score of a cluster *c* given a document *d* and the ranking model's fixed weights *u* is formalized as follows:

$$score^{rank}(d,c) = \sum_{i=1}^{3}\left(score^{cos}(d_i^r, c_i^r)\cdot u_i^r\right) + \sum_{j=1}^{2}\left(score^{cos}(d_{j+1}^r, c_1^r)\cdot u_{j+3}^r\right) + \sum_{k=1}^{3}\left(score^{ts}(d^{ts}, c_k^{ts})\cdot u_k^{ts}\right)$$

After computing the best-ranked cluster *c* for a given document *d*, a trained SVM model, which we refer to as the acceptance model, determines if the document enters the cluster by computing its acceptance score, represented as follows (*v* corresponds to the acceptance model's weights):

$$score^{accept}(d,c) = \sum_{i=1}^{3}\left(score^{cos}(d_i^r, c_i^r)\cdot v_i^r\right) + \sum_{j=1}^{2}\left(score^{cos}(d_{j+1}^r, c_1^r)\cdot v_{j+3}^r\right) + \sum_{k=1}^{3}\left(score^{ts}(d^{ts}, c_k^{ts})\cdot v_k^{ts}\right) + score^{rank}(d,c)\cdot v^{rank}$$

Finally, after receiving a new document, a cluster verifies its similarity with each cluster in the cluster pool using the ranking model described above. Each candidate cluster is then evaluated by a third SVM model, which we call cluster merge model, and the documents from each cluster that is evaluated as a positive match are inserted into the source cluster. The intuition for this model is to find separate clusters that pertain to the same story and subsequently merge them. This may happen throughout the clustering process; since few documents related to a given story have entered the system, the acceptance model may mistakenly assign separate clusters to those documents initially. As more relevant documents enter the system, those clusters may end up in similar points in the vector space and thus should be merged.

3.6. News Summarization

Monolingual Text Summarization

Text summarization aims at producing a short text segment that preserves the essential information conveyed by a longer source document. The approaches for automatic summarization can be divided into two categories: extractive and abstractive methods. The former address the problem by identifying salient parts of the source document and directly copying those to the summary (e.g.,

Kupiec et al. (1995), Dorr et al. (2003), Nallapati et al. (2017)). The latter produce the summary by generating new text that paraphrases the most relevant parts of the source document (e.g., See et al. (2017), Guo et al. (2018), Lewis et al. (2020)).

In SELMA, we will focus on summarizing video transcriptions using current neural approaches. Since extractive methods produce weak summaries over automatic transcriptions (given the low quality of the generated sentence boundaries), we shift toward abstractive summarization methods. Nonetheless, abstractive summaries often contain factual inconsistencies that hamper the adoption of these approaches in practical applications (Kryściński et al. (2019a)). For this reason, our main goal is to develop techniques to enhance the factual consistency of the generated summaries.

Our work (Pernes et al. (2022)) builds upon the state-of-the-art methodologies for abstractive summarization, namely those based on transformer sequence-to-sequence architectures, like BART (Lewis et al. (2020)), a pre-trained encoder-decoder transformer that can be finetuned in a wide range of text generation tasks, including summarization. At the same time, automatic evaluation metrics such as CTC scores (Deng et al. (2021)) have been recently proposed that exhibit a higher correlation with human judgments than traditional lexical-overlap metrics such as ROUGE. In our work, we close the loop by leveraging the recent advances in summarization metrics to create quality-aware abstractive summarizers. Namely, we proposed an energy-based model that learns to re-rank summaries according to one or a combination of these metrics. An overview of the proposed framework is presented in Figure 5. As suggested by the picture, the energy-based re-ranking model (EBR) is presented with a set of candidate summaries for a given source document and assigns a score to each candidate. The EBR is trained to mimic the ranking induced by a pre-specified gold-metric, so that the scores it provides should indicate which candidate is the best one according to that metric. We experiment using several metrics to train our energy-based re-ranker and show that it consistently improves the scores achieved by the predicted summaries. Nonetheless, human evaluation results show that the re-ranking approach should be used with care for highly abstractive summaries, as the available metrics are sometimes not sufficiently reliable for this purpose.

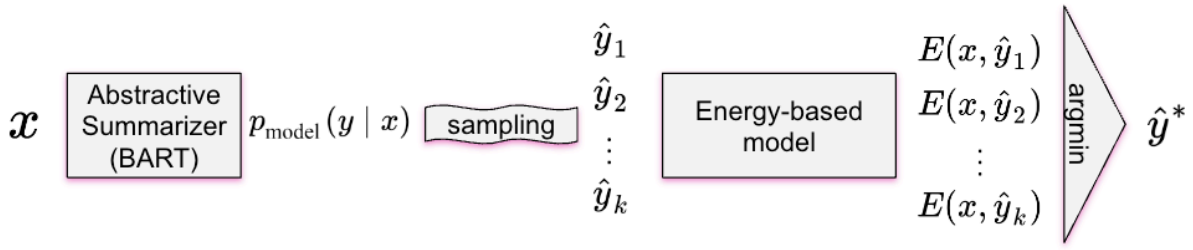

Figure 5 *Representation of the proposed energy-based re-ranking approach for abstractive summarization.*

Cross-lingual Text Summarization

The previous work has only focused on abstractive summarization of English documents. Multi-lingual resources for summarization are scarce, not only regarding the availability of trained models, but also in terms of public datasets and evaluation metrics. Moreover, SELMA proposes to address multilingual data streams and therefore following solely monolingual approaches for summarization would not be appropriate. Hence, we are addressing the challenging cross-lingual summarization problem, in which a summary in a given target language is generated from a source document in a different language. Currently available datasets for cross-lingual summarization are either English-centric (Nguyen and Daumé III (2019)) or were built by pairing documents from multi-lingual datasets using automatic similarity metrics (Hasan et al. (2021)), a process that inevitably leads to many false pairs. Thus, as a starting point, we have collected and curated a large-scale non-English-centric cross-lingual dataset for abstractive summarization with high-quality parallel. It contains 675k+ articles scraped from the EuroNews websites. The articles are written in 12 languages and the dataset includes document-summary pairs for all possible 144 source-target language pairs. This dataset is inherently cross-lingual since each article webpage contains links for the same article written in other languages. We are currently using this dataset to fine-tune and evaluate large multi-lingual language models, like mBART (Liu et al. (2020)) or mT5 (Xue et al. (2021)) on this task. As future work, we will research improved training strategies to enhance the performance of these models, mitigating hallucinations and promoting the generation of more relevant summaries.

Speech Summarization

This task is conceptually identical to text summarization except from the input modality, which is now raw speech instead of text. In the context of SELMA, speech summarization plays a central role as our efforts on abstractive summarization have the ultimate purpose of summarizing videos.

This task is traditionally divided into two subtasks addressed independently: automatic speech recognition (ASR), which generates the audio transcripts, and text summarization, that produces the summary given the transcript. However, ASR step leads to error propagation and to loss of the information provided by the speaker intonation. To tackle these problems, we are currently developing an abstractive summarization system capable of performing end-to-end speech-to-text summarization, i.e., without an intermediate transcription step.

Our approach uses a pre-trained wav2vec 2.0 model (Baevski et al. (2020)) to extract audio embeddings from the raw waveform and a transformer decoder to generate the summary text. This decoder is taken from a transformer that had previously been trained on the task of text-to-text summarization, so the decoder is expecting to receive textual token representations rather than the audio embeddings provided by wav2vec. Thus, we train a cross-modality adapter on the task of converting sequences of audio embeddings into the corresponding sequences of textual embeddings. Specifically, given an audio and the corresponding transcript, we extract the embeddings from the audio using the word2vec model and the corresponding textual embeddings using the encoder of the transformer trained for textual summarization. Then, we use these pairs of audio and textual embedding sequences to train the cross-modality adapter. By cascading the wav2vec encoder, the cross-modality adapter, and the transformer decoder, we obtain an end-to-end model for speech summarization. We are currently testing different architectures (LSTM-based and transformer-based) for the cross-modality adapter, as well as different training strategies. Data-wise, we are using French news audios extracted from Deutsche Welle and EuroNews websites and the corresponding teaser texts as the gold summaries.

We remark that surpassing the performance of the cascaded (ASR + text summarization) approach will be difficult given the relatively limited amount of data available Nonetheless, we intend to establish a first end-to-end baseline for this task and to narrow the performance gap to the cascaded model as much as possible.

4. Experimental Results

This section includes the experimental analysis of the previously defined problems, alongside their discussions, and our future plans for the rest of the project. Sub-section titles are arranged in accordance with the previous section.

4.1. Named Entity Recognition

Hierarchical Nested NER

We report named entity recognition and classification (NERC) F1 scores obtained for all entities. For each level of the hierarchy, we utilize two internal datasets related to media content: i) MediaPT, containing *42,000* training examples in Portuguese; and ii) MediaDE, containing *85,000* training examples in German. Both datasets have the same set of 61 labels, including hierarchy levels, e.g., "gpe → administrative_region → municipality", where "gpe" is the top-level (L0), "administrative_region" corresponds to L1, and "municipality" to L2. The obtained results can be seen in Table 1. It is possible to observe that both models achieve similar scores for both languages, with a slight advantage of the stack-LSTM model in MediaPT and the biaffine model in MediaDE. When comparing these models in terms of computational performance, the biaffine approach offers a clear advantage when decoding on CPU or when the sentences are short, with stack-LSTM performing similarly on GPU and slightly better for longer sentences.

Approaches \ Datasets	**MediaPT**	**MediaDE**
Development Set - NERC F1 - ALL (L0 / L1 / L2)		
Stack-LSTM	85.8 (86.5 / 85.4 / 64.5)	80.8 (80.5 / 82.4 / 59.2)
Biaffine	85.6 (86.3 / 85.2 / 64.4)	81.0 (80.2 / 83.4 / 58.9)
Test Set - NERC F1 - ALL (L0 / L1 / L2)		
Stack-LSTM	85.2 (86.0 / 84.6 / 42.0)	81.7 (81.7 / 82.8 / 53.4)
Biaffine	84.7 (85.7 / 84.0 / 48.4)	81.8 (81.8 / 82.7 / 55.6)

Table 1 *Stack-LSTM and biaffine results for MediaPT and MediaDE development and test sets*

The results presented in Table 1 highlight the previously mentioned advantage of working with pretrained multilingual contextual embedding models, which allows us to train models for different languages, as we did for MediaDE and MediaPT, and to train a single model for several languages. This allowed us to participate in the SlavNER shared task, part of the 8th Balto-Slavic NLP, where our biaffine approach was able to outperform all the other submissions for the NER subtask (Ferreira et al. (2021), Piskorski et al. (2021)), which included nested non-hierarchical entities for six different languages.

Cross-lingual Hierarchical Nested NER

In the second reporting period, datasets for additional languages became available, allowing us to further investigate the multilingual capacities of our models. Currently, the project has created NER datasets for English, French, German, Latvian, Spanish, and Portuguese. Additionally, smaller datasets to evaluate language transfer are also available for Ukrainian, Dutch, and Turkish. Table 2 contains the description of the datasets annotated with our ontology as described in D6.1 - Initial Data Management Plan. The number of annotations shown in the table are counted one for each level of the ontology so the number of different annotated text spans is much smaller than the number presented. English, French, German, and Spanish have a very good coverage, Latvian proved to have sufficient training data, and Dutch, Ukrainian, and Turkish are used only for evaluation proposes.

Language	#documents	#tokens	#annotations*
English	4500	5584365	3103202
French	3003	3086488	1771902
German	3122	2934042	1625036
Latvian	741	573731	321594
Spanish	2576	2855692	1556536
Portuguese	3199	2317747	1283964
Dutch	50	41193	22966
Ukrainian	211	160163	90865
Turkish	100	70815	40967

Table 2 *Multilingual NER datasets, *number of annotations counting with the hierarchy*

In order to compare the multilingual model performance against the monolingual baseline, we retrained all models using xml-roberta-base, except for French where we kept using camembert-base. This option permits verifying whether the multilingual model would outperform a good pure monolingual model. All the models were trained using the stack-LSTM approach with the hyper-parameters selected in our initial experiments. The monolingual results are presented in Table 3, we report F1 values for each of the ontology levels and a global F1 over the complete hierarchy. The global F1 includes the detection of the modifier tags (e.g. *nominal*, *function* and *relation),* which make this dataset much harder than other datasets publicly available.

Language	**F1**			
	All	L1	L2	L3
English	81.0	82.1	79.7	64.4
French*	85.7	87.2	83.9	0.77
German	82.2	81.9	84.0	71.5
Latvian	84.2	85.9	82.4	51.1
Spanish	83.5	85.4	81.0	54.5
Portuguese	84.4	85.5	83.4	50.4

Table 3 *Results on test sets training monolingual. *was trained using camembert instead xlm-roberta-base*

For training the multilingual model, we selected English, French, German, Latvian, Portuguese, and Spanish, because they have a good amount of training data. Table 4 reports the F1 values obtained when training the stack-LSTM model with the same hyper-parameters as in the monolingual setting. In this experiment we did not use any artifact to distinguish the languages when training or testing, because this is the simplest, less costly in resources and the most language-independent of the approaches that we researched. Table 4 shows that by training with all languages together we achieve robust improvements in most of the languages except for French, where in the monolingual setting

we used a base monolingual model (camembert-base). Although the drop of 0.6 is significant, it is not enough to justify the overhead of using a different model in a production scenario.

Language	F1				
	All	Diff to monolingual	L1	L2	L3
English	81.7	**+0.7**	82.5	80.8	64.1
French	85.1	-0.6	86,4	83.4	80.9
German	82.2	**+0.0**	81.7	83.4	72.3
Latvian	85.2	**+1.0**	86.0	84.6	75.0
Spanish	84.4	**+0.9**	86.1	82.3	59.3
Portuguese	85.1	**+0.7**	85.9	84.3	51.8

Table 4 *Results on test sets training multilingual.*

We evaluated our model zero-shot capabilities on languages present in the base model but for which we did not have NER training data. Surprisingly and against our best expectations, the multilingual model performs very well on unseen languages. To evaluate the zero-shot setting we asked the annotators to correct, remove, and add to the annotations proposed by the multilingual model. We are aware that this procedure will impose a bias on the annotators leading them to probably keep the annotations of the model but the cost and feasibility of the task imposes a pragmatic approach. Using those corrected datasets, we evaluated F1 results of the model when seeing the corrected data. Table 5 shows F1 results on the evaluation datasets for Dutch, Ukrainian, and Turkish showing that the annotators did not change much of the annotations proposed by the model for Dutch and Ukrainian. Turkish results have a considerable drop when comparing with the other two languages. This can be justified either by: the quality of the base model (xlm-roberta-base) for Turkish; a real difference in the language itself; or a different criterion was used by the Turkish annotator. If we arrive to the conclusion that the annotation is sound then we will extend the Turkish dataset and include it in the training data. To further validate these results, we will ask Priberam linguists team to validate each of these datasets with the annotators, making sure that the applied criteria were the same between these annotators and the original guidelines used for the other languages.

Language	F1			
	All	L1	L2	L3
Dutch	91.4	89.8	93.5	100
Ukrainian	90.8	88.1	94.5	100
Turkish	74.6	71.9	79.9	33.3

Table 5 *Zero-shot results after correcting the annotations predicted by the multilingual model by human annotators*

Lastly, on table 6 we present the aggregated F1 values with their support on the test dataset for each class on the ontology.

Class	Support	Precision	Recall	F1
animal	31	0.5588	0.6129	0.5846
currencies	409	0.9500	0.9756	0.9626
disciplines	127	0.5938	0.5984	0.5961
event	1599	0.7642	0.7073	0.7347
event->festivity	87	0.8144	0.9080	0.8587
event->happening	50	0.8049	0.6600	0.7253
event->organized_event	869	0.7908	0.7089	0.7476
facility	684	0.7147	0.6667	0.6899
gpe	8659	0.8820	0.9210	0.9011
gpe->address	86	0.6500	0.7558	0.6989
gpe->administrative_region	1494	0.7221	0.7289	0.7255
gpe->administrative_region->municipality	65	0.4921	0.4769	0.4844
gpe->administrative_region->parish	48	0.6970	0.4792	0.5679
gpe->city	2257	0.7613	0.8520	0.8041
gpe->continent	298	0.7975	0.8591	0.8271
gpe->country	4199	0.9208	0.9586	0.9393
gpe->non_administrative_region	498	0.5365	0.4137	0.4671
gpe->union_of_countries	100	0.8990	0.8900	0.8945
human_group	536	0.6115	0.4552	0.5219
human_group->ethnicity	67	0.6735	0.4925	0.5690
human_group->religion	77	0.7083	0.8831	0.7861
human_work	1116	0.7137	0.6478	0.6792
internet_address	193	0.8585	0.9119	0.8844

internet_address->email	14	0.8750	1.0000	0.9333
internet_address->url	80	0.7500	0.8250	0.7857
location	417	0.7419	0.7098	0.7255
location->astronomical_object	83	0.9067	0.8193	0.8608
location->geographical_feature	99	0.6893	0.7172	0.7030
location->river	63	0.7800	0.6190	0.6903
location->sea/ocean	34	0.8286	0.8529	0.8406
mod-collective	496	0.5718	0.4819	0.5230
mod-function	1190	0.7974	0.8832	0.8381
mod-negation	6	0.0000	0.0000	0.0000
mod-nominal	3954	0.6746	0.6396	0.6566
mod-relation	2449	0.8660	0.9212	0.8928
mod-sentiment_negative	10	0.8000	0.4000	0.5333
mod-sentiment_positive	6	0.0000	0.0000	0.0000
number	45	0.7381	0.6889	0.7126
number->license_plate	5	0.0000	0.0000	0.0000
number->telephone	40	0.7381	0.7750	0.7561
organization	9431	0.8397	0.8657	0.8525
organization->commercial_company	2323	0.7506	0.7891	0.7694
organization->commercial_company->brand	890	0.6762	0.6640	0.6701
organization->cultural_institution	32	0.4865	0.5625	0.5217
organization->education_institution	139	0.7325	0.8273	0.7770
organization->educational_institution	160	0.6966	0.6312	0.6623
organization->governmental_institution	2348	0.8176	0.8113	0.8145
organization->healthcare_institution	89	0.7100	0.7978	0.7513
organization->intergovernmental_organization	98	0.9255	0.8878	0.9062
organization->media	1562	0.8440	0.8867	0.8648
organization->non_governmental_organization	912	0.6660	0.7632	0.7113
organization->political_organization	396	0.8620	0.8359	0.8487
organization->religious_organization	4	0.5000	0.5000	0.5000
organization->sports_organization	1701	0.8418	0.8601	0.8508
other	725	0.6624	0.5683	0.6117
pathology	671	0.8967	0.9314	0.9137
pathology->disease	434	0.9057	0.9516	0.9281
pathology->pathogen	242	0.8659	0.8802	0.8730
people	11937	0.8889	0.9109	0.8998
people->alias	149	0.7857	0.4430	0.5665
people->job	3211	0.7494	0.7991	0.7735
quantity	2555	0.9128	0.9346	0.9236
quantity->age	652	0.9109	0.9095	0.9102
quantity->currency	660	0.9207	0.9682	0.9439

quantity->measure	349	0.8015	0.8911	0.8440
quantity->percentage	872	0.9479	0.9599	0.9538
quantity->temperature	47	0.8125	0.8298	0.8211
temporal_expression	4725	0.7821	0.8167	0.7990
temporal_expression->date	1909	0.9177	0.9466	0.9319
temporal_expression->datehour	185	0.8870	0.8486	0.8674
temporal_expression->frequency	142	0.6757	0.7042	0.6897
temporal_expression->hour	372	0.9096	0.9462	0.9275
temporal_expression->period	1116	0.6403	0.6747	0.6571
temporal_expression->time	1897	0.7110	0.7612	0.7352
time	1689	0.7824	0.7981	0.7902
time->date	680	0.9137	0.9338	0.9236
time->datehour	2	0.0000	0.0000	0.0000
time->frequency	43	0.6364	0.4884	0.5526
time->hour	154	0.8805	0.9091	0.8946
time->period	323	0.5851	0.7028	0.6385

Table 6 *Full NER results for all languages for each ontology level*

Example-Based NER

The results of example-based NER can be seen in Table 2, where we show the performance for both single-*k* and multi-*k*, for 7 datasets, including different domains, number of training examples, and number of labels. We perform hyperparameter tuning for each dataset using its development set. Few-NERD is the dataset that is more positively impacted by this approach. We hypothesize this could be due to the fact that this dataset is the only one that uses an IO-encoding, which could make it simpler to retrieve the correct tag, as it has to match only the I tag and not the B/I-tags.

Approach \ Dataset	Few-NERD	Onto Notes	Co NLL	WNUT	MIT-R	MIT-M	ATIS	Avg
Domain	*Generic*	*Generic*	*News*	*Soc. Media*	*Reviews*	*Reviews*	*Dialogue*	-
Trn. Examples	131,000	60,000	14,000	3,400	6,900*	6,700*	6,500	-
# of Labels	66	18	4	6	8	12	68	-
Development Set - NERC F1								
Class. Model	68.31	88.26	95.86	64.75	81.96	73.43	98.19	81.54
+ single-*k*	68.64	88.5	95.86	64.62	82.02	73.61	98.39	81.66
+ multi-*k*	68.75	88.53	95.87	64.74	81.9	73.6	98.33	81.67
Test Set - NERC F1								
Class. Model	67.83	90.11	92.28	57.53	80.05	71.22	95.88	79.27
+ single-*k*	68.18	90.04	92.35	57.61	80.06	71.26	95.86	79.34
+ multi-*k*	68.23	90.08	92.35	57.41	80.22	71.31	95.86	79.35

Table 7 *Example-based NER approach results with single k and multi k for different datasets (*original training data was split into training/validation splits)*

There are cases where development set improvements do not result in test set improvements (OntoNotes and ATIS), or where the improvements in the test set are rather small (remaining datasets). Regarding the possibility of using this approach as a way of incorporating user feedback, we report an experiment where we plot the performance of the linear classifier, the performance of the linear classifier plus kNN using all the available data, and the previous best linear classifier at a certain point plus kNN using the available data (i.e., at point 0.8 we interpolate the predictions made by a linear classifier trained on 60% of the training data, leveraging 80% of the training data as support data). As we can observe in Figure 5, the more support data available for the Few-NERD dataset, the clearer the benefits of using the kNN approach. In particular, it is also possible to observe the slight benefit from continuously collecting data (e.g., by comparing the point 0.8 of the line "Linear Classifier" and the point 1.0 of the line "Previous Linear Classifier + kNN", which only differ in the amount of available support data).

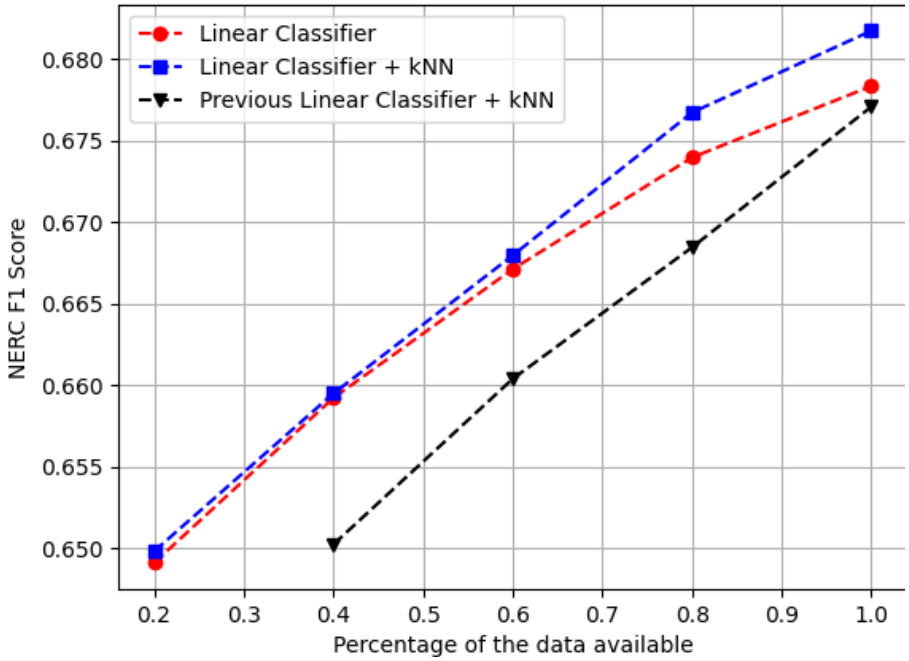

Figure 6 *Impact of increasing support data on example-based NER for the FewNERD dataset*

4.2. Entity Linking and Cross-lingual Stream Representations

We compare Yang et al. (2019) DCA model with our extended version using multilingual embeddings. We report the in-knowledge-based accuracy (i.e., accuracy disregarding predictions that do not exist in the knowledge base) for several datasets: (i) the English CoNLL 2003 shared task data, containing one development set (Aida-A) and a test set (Aida-B) with news stories from Reuters; (ii) WNED, a collection of English datasets containing news reports and newswire from various agencies (MSNBC, ACE2004, and AQUAINT) or varied English texts such as web pages or Wikipedia pages (CLUEWEB, WIKIPEDIA); (iii) sVoXel (Rosales-Méndez et al. (2018)), a collection of 15 manually annotated news articles, each available in 5 different languages.

Table 3 shows improvements across CoNLL for our base English-only model, but performance on the WNED datasets does not always improve, where the model achieves lower scores, particularly in the CLUEWEB and WIKIPEDIA datasets that are not news related.

Model	Aida-A	Aida-B	MSNBC	AQUAINT	ACE 2004	CLUE WEB	WIKI PEDIA
Original DCA	0.9003	0.8988	0.9334	0.8601	0.8773	0.7634	0.7623
Ours: EN - CoNLL	0.9195	0.9114	0.9395	0.8363	0.8853	0.7564	0.7383
Ours: All - CoNLL	0.9141	0.9157	0.9273	0.8000	0.8773	0.7206	0.7164
Ours: All - Wiki	0.8266	0.8606	0.9288	0.8965	0.8933	0.7515	0.7457
Ours: All - Both	0.8982	0.8921	0.9396	0.8769	0.8853	0.7539	0.7605

Table 8 *In-KB accuracy for English datasets for original DCA model and our embedding vocabulary - train data configurations*

Increasing the entity vocabulary leads to a small drop in performance in the WNED collection datasets. Finally, training on Wikipedia leads to a drop in CoNLL performance that can be countered by mixing both train datasets to obtain performance similar to the model using English entities only. This seems to indicate that having training data from different domains (news and Wikipedia) helps the model be more resistant to domain changes.

Table 4 shows results for the multilingual scenario where our model can surpass the original DCA results with English entities. Adding German and Portuguese entities results in metric behavior similar to the English-only scenario where there is a slight reduction in scores using only CoNLL data and adding Wikipedia data helps to revert that reduction. Comparing the English results in sVoxEL with the previous table it is possible to observe a notorious increase in performance for sVoxEL. This is because this dataset contains a small set of documents. Moreover, a considerable subset of those documents deals with the news related to the European Union leading to recurrent entities across documents that inflate scores since they are repeatedly solved. Having common entities in the documents means that they will have an English Wikipedia page. The embeddings for these entities will thus only be trained on English data, which might be a reason for the marginal increases in other languages even for German whose entities we are using.

Model	sVoxEL-fr	sVoxEL-de	sVoxEL-it	sVoxEL-es	sVoxEL-en
Original DCA	0.9200	0.8434	0.9173	0.8750	0.9327
Ours: EN - CoNLL	0.9500	0.8737	0.9523	0.9100	0.9625
Ours: All - CoNLL	0.9300	0.8737	0.9474	0.9050	0.9277
Ours: All - Wiki	0.9300	0.8789	0.9373	0.9100	0.9476
Ours: All - Both	0.9300	0.8789	0.9474	0.9100	0.9526

Table 9 *In-KB accuracy in a multilingual scenario for original DCA model and our embedding vocabulary - train data configurations*

With regards to future work, we are currently exploring a new approach based on De Cao et al. (2021), that solves entity linking through an autoregressive formulation. This new approach directly generates the linked entity's name with a highly parallel formulation that is able to use transformer-based contextual embeddings while boasting a considerable training and inference speedup over previous methods. Moreover, the original work by De Cao achieved state-of-the-art results in the CoNLL dataset. We are currently adapting the English-only original model to work in a multilingual scenario. Our initial results in CoNLL reveal a 4-point drop in micro F1 score for the multilingual approach. Despite the performance drop, our multilingual model is still competitive with past state-of-the-art English-only approaches and can work with a large set of 100 languages. Current efforts are geared towards evaluating the model with multilingual datasets. Provided the model works well, we intend to understand how to adapt either DCA or this model, depending on which is better, to work with user feedback and in stream learning scenarios.

Contextual entity representations

In this subsection, we report preliminary results on the quality of the contextual entity representations. With this aim, we follow the same procedure as Ganea and Hoffman 2017 by computing entity relatedness scores on the dataset from Ceccarelli et al. 2013. We use the same evaluation metrics: normalized discounted cumulative gain (NDCG) and mean average precision (MAP). Table 5 shows our results both for the initial multilingual scenario with mbpe embeddings and using the contextual embeddings of the fine-tuned xml-roberta-base on the multilingual SELMA NER dataset.

	NDCG@1	NDCG@5	NDCG@10	MAP
Yamada (2016) English only	0.59	0.56	0.59	0.52
Ganea and Hoffman English only	0.632	0.609	0.641	0.578
Ours multilingual mbpe	0.641	0.604	0.635	0.572
Ours multilingual contextual, mean pool	0.649	0.603	0.629	0.569

Table 5 *Entity relatedness on the test set of Ceccarelli et al. 2013*

The results above are promising since they are our baseline approach and achieved results on par with our previous multilingual mbpe entity embeddings. We will keep experimenting the other described approaches to see if we can improve the results.

4.3. Story Segmentation

We investigate a speaker embedding extractor model that shows superior performance on speaker recognition tasks. ECAPA-TDNN architecture, adopted from Desplanques et al. (2020), presents a state-of-the-art model, which combines channel- and context-dependent attention mechanism, multilayer feature aggregation, as well as squeeze-excitation and residual blocks together. Owing to its carefully designed neural architecture, this model has recently shown impressive performance in the speaker tasks. We utilized a pre-trained model developed by NVIDIA[6]. As opposed to the original work which only uses the development part of the VoxCeleb2 dataset (Chung et al. 2018) with *5,994* speakers as training data, NVIDIA's pre-trained model is trained with VoxCeleb1 (Nagrani et al. 2017) and VoxCeleb2 data together. It is a known that neural networks can benefit from data augmentation which generates extra training samples. Thus, the RIRs2 (reverb) (Ko et al. 2017) and

[6] NVIDIA's pre-trained model: https://catalog.ngc.nvidia.com/orgs/nvidia/teams/nemo/models/ecapa_tdnn.

MUSAN datasets (babble, noise) (Synder et al. 2015) are then used for data-augmentation purposes. In total, the training data contains *7,205* speakers with *1,234,651* utterances. We use the VoxCeleb1 cleaned test split to evaluate our speaker recognition experiments.

The performance of our speaker recognition systems is evaluated by the two most common metrics known as equal error rate (EER) and decision cost function (DCF). EER is a biometric security algorithm used to determine the threshold values for its false acceptance rate and its false rejection rate. When these rates are equal, the resulting value is referred to as the EER. This value indicates that the proportion of false acceptances equals the proportion of false rejections (i.e., when *Type I* error is equal to *Type II* error). The lower the equal error rate value, the higher the accuracy of the speaker system. Alternatively, the decision cost function takes the prior probabilities of the target speaker occurrences, the proportion of target and non-target speakers into consideration. The detection cost function is a simultaneous measure of discrimination and calibration. In our experiments, we prefer to report the minimum value of the DCF curve that is called minDCF.

Three types of speaker recognition architectures will serve as baselines to measure the impact of our proposed model: i-vector, x-vector, and ResNet-based system, which currently provides state-of-the-art performance on several recognition tasks such as VoxSRC (Chung et al. 2019).

Dehak et al. (2011) proposed the i-vector model, which is a combination of speaker space and channel space. A new low-dimensional space defined as the total factor space represents each utterance with a low-dimensional feature vector termed i-vector. In other words, each utterance is projected onto the entire factor space and is characterized by an i-vector. The input features are *20* MFCCs with a frame length of *25ms* that are mean-normalized over a sliding window of up to *3* seconds. Delta and acceleration are appended to create *60*-dimensional feature vectors.

Snyder et al. (2018a) presented an x-vector system that is based on the neural network-based embeddings described in Section 3.3 with a greater detail there. The features are *24*-dimensional filter banks with a frame length of *25ms*, mean-normalized over a sliding window of up to *3* seconds.

Extended TDNN (E-TDNN) x-vector architecture improves the original x-vector system where the initial frame layers consist of 1-D dilated convolutional layers (Zeinali et al. 2019). Residual connections are also introduced in all frame-level layers, followed by an attentive statistical pooling

that calculates the mean and standard deviations of the final frame-level features (Garcia-Romero et al. 2020). After the statistical pooling, two fully-connected layers are introduced, with the first one acting as a bottleneck layer to generate the low-dimensional speaker characterizing embedding.

A performance overview of these baseline systems and the proposed ECAPA-TDNN system are given in the table below. Speaker embeddings are extracted from the final fully-connected layer for all systems. Trial scores are produced using the cosine distance between embeddings. We use *512* convolutional layers with the input features of 80-dimensional MFCCs from a *25ms* window with a 10ms frameshift. As a final augmentation step, SpecAugment (Park et al. 2019) on the log-mel spectrogram is applied. The model randomly masks *0 to 5* frames in the time domain and *0 to 10* channels in the frequency domain. ECAPA-TDNN architecture significantly outperforms all baselines and gives an average relative improvement of 32% in EER and 25% in MinDCF over the E-TDNN.

Model	EER [%]	MinDCF
i-Vector (Dehak et al. 2011)	5.32	0.49
x-Vector (Snyder et al. 2018a)	3.14	0.33
E-TDNN (Zeinali et al. 2019)	1.49	0.16
***This Work:* ECAPA-TDNN**	1.01	0.12

Table 10 *EER and MinDCF performances of all systems on the standard VoxCeleb1 test split*

In the following experiments, we will continue with an ablation study to gain a deeper understanding of how each of the components affects the performance, such as SE-block, MFA, or Res2Net-block. Apart from that, we will investigate three crucial future directions in the rest of the project: (i) extending the ECAPA-TDNN model to speaker diarization problem, (ii) implementing domain adaptation scenarios to transfer recognition model from one language to another, (iii) incorporating the user feedback mechanism over the speaker adaptation idea such that users can explicitly provide new data from an unknown speaker to achieve better recognition performance.

4.4. Online News Classification

We compared the results of our new approaches on the News Classification problem to the model previously described in report D5.1 of the SUMMA project, hereafter referred to as "multi-CNN". We report micro-F1 scores of our models, trained on the Lusa Portuguese news dataset with IPTC subject labels. We also report zero-shot cross-lingual results on the smaller English and Spanish datasets. Table 6 shows the results of our sentence embedding attention-based models. We compare the results of using a single query to generate a single representation of the model; Three queries, corresponding to the three depths of the label hierarchy, to develop three representations; And having each label learn its own query. As a baseline, we also present the results of averaging all sentence embeddings in a document and using the resulting vector for classification.

Model	Portuguese F1	English F1	Spanish F1
Multi-CNN	64.33%	49.32%	52.61%
DistilUSE + average	65.08%	54.24%	49.16%
DistilUSE + global attention	66.77%	53.19%	60.05%
DistilUSE + hierarchy depth attention	67.40%	52.13%	61.30%
DistilUSE + label attention	66.48%	54.52%	60.63%

Table 11 *F1 performance of sentence embedding attention-based models on Portuguese, English, and Spanish testing datasets (English and Spanish are zero-shot languages)*

Table 7 shows the results of our AttentionXML based models. We compare the results of using a traditional AttentionXML with a word embedding layer using the multilingual BPEmb embeddings and using a multilingual mBERT model to generate the contextual word embeddings that are fed into the biLSTM of AttentionXML. Our current results and incremental improvements of F1 scores over previous models show the promise of the current direction of work. As future work we intend on leveraging the information in the label descriptions available in the IPTC vocabulary to generate better label embeddings. This is similar to work done in the past by Mittal et al.

Model	Portuguese F1	English F1	Spanish F1
Multi-CNN	64.33%	49.32%	52.61%
AttentionXML + BPEmb	68.63%	33.26%	55.29%
AttentionXML + mBERT	70.10%	52.88%	64.36%

Table 12 *F1 performance of sentence embedding attention-based models on Portuguese, English, and Spanish testing datasets, for models trained on the Lusa dataset (English and Spanish are zero-shot languages)*

A drawback to be tackled is the limited input size of 512 tokens on the AttentionXML+mBERT model. Approaches to this issue include using BERT style models that are pretrained for longer inputs, such as the Longformer (Beltagy et al. 2020). Alternatively, we intend on experimenting with training AttentionXML's biLSTM to join the concatenated outputs of consecutive mBERT forward passes.

As expected of models that are fine-tuned on a monolingual Portuguese dataset, the best results are obtained on the Portuguese language test sets. This suggests that some multilingual performance of the pretrained models is lost in our experimental setup. We have also trained some of the described models on the multilingual dataset created from joining the Finnish and Portuguese news datasets from STT and Lusa, respectively. It should be noted that these results cannot be fairly compared to the ones shown on the previous tables because the label space changed to include labels that were added for being present in the Finnish dataset. The results for these models are shown in Table 8, along with the scores of the old Multi-CNN model, which has not been retrained on the Finnish dataset, and along a hybrid model of AttentionXML with a multilingual Roberta-Large.

Model	Portuguese F1	Finnish F1	English F1	Spanish F1
Multi-CNN*	64.33%	14.93%	49.32%	52.61%
AttentionXML + BPEmb	67.15%	67.86%	29.04%	43.37%
AttentionXML + mBERT	67.69%	66.28%	55.44%	64.04%
AttentionXML + Roberta	70.32%	68.58%	56.26%	62.81%

Table 13 *F1 performance of sentence embedding attention-based models on Portuguese, Finnish, English, and Spanish testing datasets, for models trained on the Lusa+STT dataset (*excluding Multi-CNN) (English and Spanish are zero-shot languages)*

4.5. Online News Clustering

We follow previous work on this task and evaluate our system on a news clustering dataset (Rupnik et al. (2016)). Besides the three main languages (English, Spanish, and German), this dataset also provides a significant number of documents in Chinese and Russian, as well as documents in Slovenian, Croatian, French, and Italian.

Systems	BCubed			Standard			Clusters
	F1	***P***	***R***	***F1***	***P***	***R***	
Miranda et al. (2018)	-	-	-	84.00	83.0	85.00	-
Linger et al. (2020)	82.06	80.25	83.97	86.49	85.11	87.92	606
4-F Rank + Accept.	88.02	91.31	84.95	92.34	97.26	87.09	957
8-F Rank + Accept.	89.24	92.62	86.11	93.76	97.66	90.15	1023
8-F Rank + Accept. + Merge	90.10	89.70	90.51	97.21	97.01	97.42	812

Table 14 *Cross-lingual clustering performances on the news clustering test dataset where P and R represent the precision and recall respectively*

The samples allow us to roughly preview the system's performance in other languages besides the ones it was trained in. The dataset is composed of 34,687 news documents, and it is divided into two sets: a training set comprised of 20,813 articles and a test set that contains 13,874 articles. For cross-

lingual clustering, as shown in Table 9, our system achieves state-of-the-art performance on BCubed F1 (Amigó et al. (2009)) (+8.04) and the standard F1 (+11.33) despite producing a larger number of clusters. We also perform an ablation study that shows the relative importance of system components. 4-F Rank+Accept. refers to the clustering system with a 4-feature ranking and acceptance model. Adding the other features, such as 8-F Rank+Accept., improved both standard (+1.42) and BCubed F1 (+1.22). Finally, the cluster merge model is added to our system, which results in gains for both standard (+3.35) and BCubed F1 (+0.86).

Languages	**BCubed**			**Standard**			**Clusters**
	F1	***P***	***R***	***F1***	***P***	***R***	
Chinese	96.18	100.00	92.65	99.07	100.00	98.16	28
Slovenian	76.92	100.00	62.50	79.67	100.00	66.21	12
Croatian	77.85	100.00	63.73	74.99	100.00	60.00	5
French	98.50	100.00	97.04	99.69	100.00	99.39	3
Russian	100.00	100.00	100.00	100.00	100.00	100.00	1
Italian	98.86	100.00	97.75	98.78	100.00	97.59	3

Table 15 *Clustering performances on other languages where P and R represent the precision and recall respectively*

Given the nature of our system, we evaluated it on the remaining languages of the dataset, as shown in Table 10. Our ranking, acceptance, and cluster merge models were not trained on any data from these languages (except for Chinese), making this a zero-shot clustering scenario. Chinese, French, Russian, and Italian document clustering had high F1 scores, with results above 95%, and both Slovenian and Croatian had initial clustering scores above 70%.

Regarding future work, a relevant approach to follow is the implementation of high-performance vector search in order to improve clustering speed and scalability, which takes advantage of the current fully dense clustering space. Taking the feedback of users into account on the clustering process in order to fine-tune the models is also a pertinent direction. Regarding the improvement of the current evaluation scores, following work on entity-aware contextual embeddings is also a

relevant approach, with the main obstacle being the need of said entity-awareness to cover all of the SELMA languages.

4.6. News Summarization

Monolingual Text Summarization

We evaluate our energy-based re-ranking model (EBR) described in Section 3.6 against a baseline BART system with the usual beam search decoding algorithm and against other improved summarization systems, namely: BRIO (Liu et al. (2022)), which employs a ranking loss as an additional term on the training of the abstractive system; CLIFF (Cao & Wang (2021)), which uses data augmentation techniques and contrastive learning to enhance the factual consistency of the summaries; DAE (Goyal & Durret (2021)), which detects and discards non-factual tokens from the training data; FASum (Zhu et al. (2021)), which incorporates knowledge graphs also to enhance factual consistency; and SumRerank (Ravaut et al. (2022)), which employs a mixture of experts to train a re-ranker on the combination of various metrics. For our model and for SumRerank, we sample 8 candidate summaries from BART using diverse beam search (Vijayakumar et al. (2016)). The models are evaluated on two benchmark datasets for abstractive summarization: CNN/DailyMail (Hermann et al. (2015)) and XSum (Narayan et al. (2018)), both containing news articles paired with their respective reference summaries. In XSum each summary consists of a single sentence, while in CNN/DailyMail it can comprise three sentences or more. Regarding the automatic evaluation metrics, apart from the usual ROUGE scores, we also measured the QuestEval (Scialom et al. (2021)) and CTC scores (Deng et al. (2021)), which are transformer-based metrics that exhibit a stronger correlation with human judgment. The results of the baselines and of our EBR trained with the CTC metric are in Table 11.

Models	CNN/DailyMail				XSum			
	R2	***QE***	***Cons***	***Rel***	***R2***	***QE***	***Cons***	***Rel***
BART	20.75	43.28	95.01	61.75	19.42	28.27	83.18	52.23
BRIO	**24.06**	43.49	89.61	60.75	-	-	-	-
CLIFF	20.88	43.28	94.68	60.38	**21.41**	**29.34**	82.57	51.92
DAE	-	-	-	-	14.19	29.20	79.45	51.05
FASum	17.68	42.87	94.30	57.91	9.97	24.35	75.45	39.42
SumRerank	21.73	43.61	95.07	62.49	21.40	28.76	83.00	52.75
EBR [Ours]	20.87	**43.79**	**96.15**	**63.32**	19.72	28.66	**86.03**	**54.74**

Table 16 *Results of our model and baselines on each of the automatic evaluation metrics. (R2: ROUGE-2, QE: QuestEval, Cons: CTC consistency, Rel: CTC relevance)*

We see that our model outperforms or is competitive with the remaining in all the metrics except ROUGE, which is known to correlate poorly with human judgment. Interestingly, despite the fact that our model was trained with the CTC scores only, it yields improvements over BART in ROUGE and QuestEval metrics as well.

Even though the results of automatic evaluation are promising, directly optimizing for a metric is risky as none of these metrics correlate perfectly with human judgment. For this reason, it is crucial to conduct a human evaluation. Specifically, we asked the judges to do pairwise comparisons between the summaries generated by three models: BART, CLIFF, which was the strongest published baseline at the time we conducted this study, and our EBR trained with the CTC scores. For each source document, we presented three pairs of summaries consecutively, which correspond to all the pairwise combinations of the summaries generated by the three systems. Then, we asked the judges to rank the summaries in each pair according to three criteria: factual consistency, relevance, and fluency. For each criterion, the judges had to evaluate whether the first summary was better than, tied with, or worse than the second. We randomly sampled 30 source documents from the test set of CNN/DailyMail and another 30 from the test set of XSum, so each judge was asked to compare 180 pairs of summaries. The results are presented in Table 12. The first observation is that our EBR model

succeeds at improving the quality of the candidates sampled from BART on the CNN/DailyMail dataset in all three criteria. On XSum, the improvements are marginal or even absent, except on the fluency dimension. Surprisingly, the comparison of our model with CLIFF contradicts the results of the automatic evaluation (Table 11), especially on the XSum dataset. Further analysis conducted in our work shows that the primary cause for this contradiction are flaws in the CTC metrics that our model was trained to mimic. Specifically, the CTC consistency metric often fails at detecting factual inconsistencies, especially when the summaries are highly abstractive as is the case in XSum.

Despite the improvements obtained by our approach, the lack of more reliable metrics to automatically assess summary quality, in particular its factual consistency, spoils its effectiveness in more abstractive settings. We reemphasize the difficulty of evaluating summary quality automatically and therefore this is a topic that should deserve our attention in future work. Moreover, most of the aforementioned transformer-based metrics (e.g. CTC scores and QuestEval) are only available for English and therefore the applicability of our method to non-English data is not straightforward. This observation also motivates us to pursue methods to address abstractive summarization in the multi-lingual and cross-lingual settings.

Models	**CNN/DailyMail**			**XSum**		
	FC	***R***	***F***	***FC***	***R***	***F***
CLIFF is better	.17	.33	**.33**	**.25**	**.32**	**.27**
Tie	.65	.24	.40	.63	.63	.68
BART is better	**.18**	**.43**	.27	.12	.05	.05
EBR is better	**.13**	**.30**	**.24**	**.15**	.12	**.30**
Tie	.80	.52	.58	.72	.77	.63
BART is better	.07	.18	.18	.13	.12	.07
EBR is better	.12	**.45**	**.32**	.10	.08	.07
Tie	.68	.20	.42	.63	.63	.88

CLIFF is better	**.20**	.35	.27	**.27**	**.28**	**.08**
Agreement	.50	.63	.54	.56	.58	.87
Strong disag.	.01	.11	.08	.01	.00	.00

Table 17 *Proportion of times that each model was considered the best for the human judges in each pairwise comparison according to three criteria: factual consistency (FC), relevance (R), and fluency (F). Rows "Agreement" and "Strong disag." show, respectively, the proportion of times that the two judges agreed and chose opposite options on the pairwise comparisons.*

Cross-lingual Text Summarization

As mentioned in Section 3.6, we will present a large-scale dataset for cross-lingual summarization in 12 languages, comprising document-summary examples in all possible 144 language pairs. In addition, we will also show the performance of mT5 trained end-to-end on this task using our dataset. Some preliminary results are presented in the following. For mT5, two special tokens identifying the source and target languages are prefixed to every input sequence at the encoder side. At the decoder side, we use language-specific start-of-sequence tokens identifying the target language. The performance of this model is compared with a more conventional approach that treats translation and summarization as separate steps, using English as the pivot language. Specifically, given a document in a source language *X* and a target language *Y*, we use a machine translation (MT) model to translate the document from *X* to English, then we obtain a summary in English using a monolingual abstractive summarization model, and finally we use MT again to translate the English summary to *Y*. Obviously, when *X* (resp. *Y*) is English, the initial (resp. final) translation step is not necessary. In our experiments, we used the M2M100 1.2B model (Fan et al. (2021)) for MT and a BART fine-tuned on the English split of our data for abstractive summarization.

The ROUGE-2 scores of the end-to-end and cascaded approaches are presented in Tables 13 - 16. For conciseness, we only show results for a high-resource language, German (Tables 13 and 14), and a low-resource language, Persian (Tables 15 and 16).

Models	de → *											
	ar	de	el	en	es	fa	fr	hu	it	pt	ru	tr
Cascaded	1.64	7.26	**2.13**	**5.62**	5.11	1.21	**6.09**	2.17	3.36	3.95	2.79	4.60
mT5	**1.66**	**10.56**	2.04	5.30	**5.18**	**1.88**	6.06	**2.83**	**3.55**	**4.53**	**3.40**	**6.85**

Table 18 *ROUGE-2 scores of the mT5 model and cascaded approach on cross-lingual summarization from German to 12 languages*

Models	* → de											
	ar	de	el	en	es	fa	fr	hu	it	pt	ru	tr
Cascaded	2.51	7.26	2.62	**3.43**	3.03	2.15	2.88	2.86	2.98	2.97	2.76	2.90
mT5	**3.35**	**10.56**	**3.39**	3.13	**3.63**	**3.20**	**3.57**	**3.46**	**3.53**	**3.55**	**3.33**	**3.61**

Table 19 *ROUGE-2 scores of the mT5 model and cascaded approach on cross-lingual summarization from 12 languages to German*

Models	fa → *											
	ar	de	el	en	es	fa	fr	hu	it	pt	ru	tr
Cascaded	**1.54**	2.15	3.13	4.31	**5.24**	4.58	**5.41**	2.31	**3.67**	3.76	3.16	3.81
mT5	1.31	**3.20**	**4.00**	**4.98**	4.71	**7.21**	4.98	**2.55**	3.61	**4.43**	**3.87**	**6.02**

Table 20 *ROUGE-2 scores of the mT5 model and cascaded approach on cross-lingual summarization from Persian to 12 languages*

Models	* → fa											
	ar	de	el	en	es	fa	fr	hu	it	pt	ru	tr
Cascaded	1.07	1.21	**1.07**	1.23	1.42	4.58	1.23	1.34	1.20	1.30	1.23	0.73
mT5	**1.74**	**1.88**	1.00	**1.46**	**2.19**	**7.21**	**1.71**	**1.96**	**1.97**	**1.91**	**1.88**	**1.26**

Table 21 *ROUGE-2 scores of the mT5 model and cascaded approach on cross-lingual summarization from 12 languages to Persian*

The first observation is that the end-to-end cross-lingual summarization with mT5 surpasses the performance of the cascaded approach in most cases, both in the high-resource and low-resource scenarios. However, the obtained scores are still very low for a few language pairs, so further improvement is required. Moreover, a preliminary human inspection of the generated summaries showed that, in many situations, the information presented in the summary changed radically depending on the target language. The presence of hallucinations or mistranslated entities were also very frequent. These are problems we plan to address in future work.

5. Conclusions

In this report, we present the current research and development undertaken in the SELMA work package, WP2. In particular, we present our latest advances in named entity recognition, entity linking, story segmentation, news summarization, online news classification, and clustering. Significant progress was made on the multilingual NER achieving very impressive zero shot results and allowing us to use only one model for all tested languages. The work done on multilingual summarization is very novel and the contribution of this new dataset will probably be much appreciated by the research community. Our improvements on the explainability of the classification models will enable the use of these models in other scenarios where human supervision is essential. Our work on clustering and summarization was accepted at the SIGIR and EMNLP conferences.

All the different components being developed in WP2 are the results of our ongoing research effort to find the systems that better suit the use-cases of SELMA. These components are being integrated on UC1 and UC2 as described in D2.5 and D2.6.

Bibliography

Baevski, A., Zhou, Y., Mohamed, A., & Auli, M. (2020). wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in Neural Information Processing Systems,* 33, 12449-12460.

Beltagy, I., Peters, M. E., & Cohan, A. (2020). Longformer: The Long-document Transformer. *ArXiv:2004.05150.*

Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics.*

Cao, S., & Wang, L. (2021). CLIFF: Contrastive Learning for Improving Faithfulness and Factuality in Abstractive Summarization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing* (pp. 6633-6649).

Cardoso, R., Marinho, Z., Mendes, A., & Miranda, S. (2021). Priberam at MESINESP Multi-label Classification of Medical Texts Task. *Proceedings of International Conference of the CLEF Association.*

Chalkidis, I., Fergadiotis, M., Kotitsas, S., Malakasiotis, P., Aletras, N., & Androutsopoulos, I. (2020). An Empirical Study on Large-scale Multi-label Text Classification Including Few and Zero-shot Labels. *Empirical Methods in Natural Language Processing (EMNLP).*

Chung, J. S., Nagrani, A., & Zisserman, A. (2018). Voxceleb2: Deep Speaker Recognition. *ArXiv:1806.05622.*

Chung, J. S., Nagrani, A., Coto, E., Xie, W., McLaren, M., Reynolds, D. A., & Zisserman, A. (2019). VoxSRC 2019: The First VoxCeleb Speaker Recognition Challenge. *ArXiv:1912.02522.*

De Cao, N., Aziz, W., & Titov, I. (2021). Highly Parallel Autoregressive Entity Linking with Discriminative Correction. *Empirical Methods in Natural Language Processing (EMNLP).*

Dehak, N., Torres-Carrasquillo, P. A., Reynolds, D., & Dehak, R. (2011). Language Recognition via i-Vectors and Dimensionality Reduction. *In INTERSPEECH.*

Deng, J., Guo, J., Xue, N., & Zafeiriou, S. (2019). ArcFace: Additive Angular Margin Loss for Deep Face Recognition. *IEEE/CVF Conference on Computer Vision and Pattern Recognition.*

Deng, M., Tan, B., Liu, Z., Xing, E. P., & Hu, Z. (2021). Compression, Transduction, and Creation: A Unified Framework for Evaluating Natural Language Generation. *ArXiv:2109.06379.*

Desplanques, B., Thienpondt, J., & Demuynck, K. (2020). ECAPA-TDNN: Emphasized Channel Attention, Propagation and Aggregation in TDNN-Based Speaker Verification. *ArXiv:2005.07143.*

Devlin, J., Chang, M., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *North American Chapter of the Association for Computational Linguistics: Human Language Technologies.*

Dorr, B., Zajic, D., & Schwartz, R. (2003). Hedge Trimmer: A Parse-and-trim Approach to Headline Generation. *Proceedings of the HLT-NAACL on Text Summarization Workshop.*

Fan, A., Bhosale, S., Schwenk, H., Ma, Z., El-Kishky, A., Goyal, S., ... & Joulin, A. (2021). Beyond English-Centric Multilingual Machine Translation. *Journal of Machine Learning Research*, *22*(107), 1-48.

Ferreira, P., Cardoso, R., & Mendes, A. (2021). Priberam Labs at the 3rd. Shared Task on SlavNER. *Proceedings of the Balto-Slavic Natural Language Processing Workshop.*

Ganea, O.-E., & Hofmann, T. (2017). Deep Joint Entity Disambiguation with Local Neural Attention. *ArXiv:1704.04920.*

Gao, S., Cheng, M. M., Zhao, K., Zhang, X. Y., Yang, M. H., & Torr, P. H. (2019a). Res2Net: A new Multi-scale Backbone Architecture. *IEEE Transactions on Pattern Analysis and Machine Intelligence.*

Gao, Z., Song, Y., McLoughlin, I. V., Li, P., Jiang, Y., & Dai, L. R. (2019b). Improving Aggregation and Loss Function for Better Embedding Learning in End-to-End Speaker Verification System. *In INTERSPEECH.*

Garcia-Romero, D., Snyder, D., Sell, G., McCree, A., Povey, D., & Khudanpur, S. (2019). X-Vector DNN Refinement with Full-Length Recordings for Speaker Recognition*. In INTERSPEECH.*

Garcia-Romero, D., McCree, A., Snyder, D., & Sell, G. (2020). JHU-HLTCOE System for the VoxSRC Speaker Recognition Challenge. *International Conference on Acoustics, Speech and Signal Processing (ICASSP).*

Goyal, T., & Durrett, G. (2021). Annotating and Modeling Fine-grained Factuality in Summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 1449-1462).

Guo, H., Pasunuru, R., & Bansal, M. (2018). Soft Layer-specific Multi-task Summarization with Entailment and Question Generation. *Proceedings of the Annual Meeting of the Association for Computational Linguistics.*

Hasan, T., Bhattacharjee, A., Islam, M. S., Mubasshir, K., Li, Y. F., Kang, Y. B., ... & Shahriyar, R. (2021). XL-Sum: Large-Scale Multilingual Abstractive Summarization for 44 Languages. *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021* (pp. 4693-4703).

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. *IEEE Conference on Computer Vision and Pattern Recognition.*

Heinzerling, B., & Strube, M. (2018). BPEmb: Tokenization-free Pre-trained Subword Embeddings in 275 Languages. *ArXiv:1710.02187.*

Hermann, K. M., Kočiský, T., Grefenstette, E., Espeholt, L., Kay, W., Suleyman, M., & Blunsom, P. (2015). Teaching machines to read and comprehend. In *Proceedings of the 28th International Conference on Neural Information Processing Systems-Volume 1* (pp. 1693-1701).

Houlsby, N., Giurgiu, A., Jastrzebski, S., Morrone, B., De Laroussilhe, Q., Gesmundo, A., ... & Gelly, S. (2019). Parameter-efficient transfer learning for NLP. In *International Conference on Machine Learning* (pp. 2790-2799). PMLR.

Hu, J., Shen, L., & Sun, G. (2018). Squeeze-and-Excitation networks. *IEEE Conference on Computer Vision and Pattern Recognition.*

Ioffe, S. (2006). Probabilistic Linear Discriminant Analysis. *European Conference on Computer Vision.*

Johnson, A., Pollard, T., Shen, L., Lehman, L., Feng, M., Ghassemi, M., Moody, B., Szolovits, P., Celi, L., Mark, R., (2016). MIMIC-III, a freely accessible critical care database. *Scientific Data.*

Khandelwal, U., Levy, O., Jurafsky, D., Zettlemoyer, L., & Lewis, M. (2020). Generalization through Memorization: Nearest Neighbor Language Models. *ArXiv:1911.00172.*

Khandelwal, U., Fan, A., Jurafsky, D., Zettlemoyer, L., & Lewis, M. (2021). Nearest Neighbor Machine Translation. *ArXiv:2010.00710.*

Ko, T., Peddinti, V., Povey, D., Seltzer, M. L., & Khudanpur, S. (2017). A Study on Data Augmentation of Reverberant Speech for Robust Speech Recognition. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).*

Kryściński, W., Keskar, N. S., McCann, B., Xiong, C., & Socher, R. (2019a). Neural Text Summarization: A Critical Evaluation. *ArXiv:1908.08960.*

Kryściński, W., McCann, B., Xiong, C., & Socher, R. (2019b). Evaluating the Factual Consistency of Abstractive Text Summarization. *ArXiv:1910.12840*

Kupiec, J., Pedersen, J., & Chen, F. (1995). A Trainable Document Summarizer. *Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.*

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., & Zettlemoyer, L. (2020). BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. *Proceedings of the Annual Meeting of Association for Computational Linguistics.*

Linger, M., & Hajaiej, M. (2020). Batch Clustering for Multilingual News Streaming. *Proceedings of the Text2Story Workshop.*

Liu, Y., Gu, J., Goyal, N., Li, X., Edunov, S., Ghazvininejad, M., ... & Zettlemoyer, L. (2020). Multilingual Denoising Pre-training for Neural Machine Translation. *Transactions of the Association for Computational Linguistics,* 8, 726-742.

Lu, L., Liu, L., Hussain, M. J., & Liu, Y. (2017). I Sense You by Breath: Speaker Recognition via Breath Biometrics. *IEEE Transactions on Dependable and Secure Computing.*

Liu, Y., Liu, P., Radev, D., & Neubig, G. (2022). BRIO: Bringing Order to Abstractive Summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 2890-2903).

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V. (2019). RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv:1907.11692*

Marinho, Z., Mendes, A., Miranda, S., & Nogueira, D. (2019). Hierarchical Nested Named Entity Recognition. *Proceedings of the Clinical Natural Language Processing Workshop.*

Mikolov, T., Chen, K., Corrado, G.S., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. *International Conference on Learning Representations (ICLR).*

Miranda, S., Znotins, A., Cohen, S.B., & Barzdins, G. (2018). Multilingual Clustering of Streaming News. *Empirical Methods in Natural Language Processing (EMNLP).*

Mittal, A., Dahiya, K., Agrawal, S., Saini, D., Agarwal, S., Kar, P., & Varma, M. (2021). DECAF: Deep Extreme Classification with Label Features. Proceedings of the 14th ACM International Conference on Web Search and Data Mining.

Nagrani, A., Chung, J. S., & Zisserman, A. (2017). VoxCeleb: A Large-scale Speaker Identification Dataset. ArXiv:1706.08612.

Nagrani, A., Chung, J. S., Xie, W., & Zisserman, A. (2020). VoxCeleb: Large-scale Speaker Verification in the Wild. Computer Speech & Language.

Nallapati, R., Zhai, F., & Zhou, B. (2017). SummaRuNNer: A Recurrent Neural Network Based Sequence Model for Extractive Summarization of Documents. *AAAI Conference on Artificial Intelligence.*

Narayan, S., Cohen, S. B., & Lapata, M. (2018). Don't Give Me the Details, Just the Summary! Topic-Aware Convolutional Neural Networks for Extreme Summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing* (pp. 1797-1807).

Nguyen, K., & Daumé III, H. (2019). Global Voices: Crossing Borders in Automatic News Summarization. *Proceedings of the 2nd Workshop on New Frontiers in Summarization* (pp. 90-97).

Okabe, K., Koshinaka, T., & Shinoda, K. (2018). Attentive Statistics Pooling for Deep Speaker Embedding. *ArXiv:1803.10963.*

Park, D. S., Chan, W., Zhang, Y., Chiu, C. C., Zoph, B., Cubuk, E. D., & Le, Q. V. (2019). SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition. *ArXiv:1904.08779.*

Pernes, D., Mendes, A., & Martins, A. F. (2022). Improving abstractive summarization with energy-based re-ranking. *Proceedings of the 2nd Workshop on Natural Language Generation, Evaluation, and Metrics (GEM 2022)*.

Piskorski, J., Babych, B., Kancheva, Z., Kanishcheva, O., Lebedeva, M.Y., Marcinczuk, M., Nakov, P., Osenova, P., Pivovarova, L., Pollak, S., Pribán, P., Radev, I., Robnik-Sikonja, M., Starko, V., Steinberger, J., & Yangarber, R. (2021). Slav-NER: the 3rd Cross-lingual Challenge on Recognition, Normalization, Classification, and Linking of Named Entities across Slavic Languages. *Proceedings of the Balto-Slavic Natural Language Processing Workshop.*

Ravaut, M., Joty, S., & Chen, N. (2022). SummaReranker: A Multi-Task Mixture-of-Experts Re-ranking Framework for Abstractive Summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 4504-4524).

Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. *Proceedings of the Empirical Methods in Natural Language Processing.*

Rosales-Méndez, H., Hogan, A., & Poblete, B. (2018). VoxEL: A Benchmark Dataset for Multilingual Entity Linking. *International Semantic Web Conference.*

Rupnik, J., Muhic, A., Leban, G., Skraba, P., Fortuna, B., & Grobelnik, M. (2016). News Across Languages—Cross-lingual Document Similarity and Event Tracking. *Journal of Artificial Intelligence Research.*

Santos, J., Mendes, A. & Miranda, S. (2022). Simplifying Multilingual News Clustering Through Projection From a Shared Space. *Proceedings of Text2Story - Fifth Workshop on Narrative Extraction From Texts* held in conjunction with the *44th European Conference on Information Retrieval (ECIR 2022)* Stavanger, Norway, April 10, 2022 (pp. 015-024)

Scialom, T., Dray, P. A., Lamprier, S., Piwowarski, B., Staiano, J., Wang, A., & Gallinari, P. (2021). QuestEval: Summarization Asks for Fact-based Evaluation. *In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing* (pp. 6594-6604).

See, A., Liu, P. J., & Manning, C. D. (2017). Get to the Point: Summarization with Pointer-generator Networks. *Proceedings of Annual Meeting of the Association for Computational Linguistics.*

Snyder, D., Garcia-Romero, D., McCree, A., Sell, G., Povey, D., & Khudanpur, S. (2018a). Spoken Language Recognition Using X-Vectors. *In Odyssey.*

Snyder, D., Garcia-Romero, D., Sell, G., Povey, D., & Khudanpur, S. (2018b). X-Vectors: Robust DNN Embeddings for Speaker Recognition. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).*

Tjong Kim Sang, E. F., & De Meulder, F. (2003). Introduction to the CoNLL-2003 Shared Task: Language-independent Named Entity Recognition. *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL.*

Tirumala, S. S., Shahamiri, S. R., Garhwal, A. S., & Wang, R. (2017). Speaker Identification Features Extraction Methods: A Systematic Review. *Expert Systems with Applications.*

Vijayakumar, A. K., Cogswell, M., Selvaraju, R. R., Sun, Q., Lee, S., Crandall, D., & Batra, D. (2016). Diverse beam search: Decoding diverse solutions from neural sequence models. *arXiv preprint arXiv:1610.02424*.

Wang, Q., Downey, C., Wan, L., Mansfield, P. A., & Moreno, I. L. (2018). Speaker Diarization with LSTM. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).*

Xue, L., Constant, N., Roberts, A., Kale, M., Al-Rfou, R., Siddhant, A., ... & Raffel, C. (2021). mT5: A Massively Multilingual Pre-trained Text-to-Text Transformer. *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 483-498).

Yang, X., Gu, X., Lin, S., Tang, S., Zhuang, Y., Wu, F., Chen, Z., Hu, G., & Ren, X. (2019). Learning Dynamic Context Augmentation for Global Entity Linking. *Empirical Methods in NLP and International Joint Conference on NLP (EMNLP-IJCNLP).*

You, R., Zhang, Z., Wang, Z., Dai, S., Mamitsuka, H., & Zhu, S. (2019). AttentionXML: Label Tree-based Attention-aware Deep Model for High-performance Extreme Multi-label Text Classification. *Advances in Neural Information Processing Systems*.

Yu, J., Bohnet, B., & Poesio, M. (2020). Named Entity Recognition as Dependency Parsing. *Proceedings of the Annual Meeting of the Association for Computational Linguistics.*

Zeinali, H., Wang, S., Silnova, A., Matějka, P., & Plchot, O. (2019). BUT System Description to VoxCeleb Speaker Recognition Challenge 2019. *ArXiv:1910.12592.*

Zhu, C., Hinthorn, W., Xu, R., Zeng, Q., Zeng, M., Huang, X., & Jiang, M. (2021). Enhancing Factual Consistency of Abstractive Summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 718-733).