Project Report - Multilingual Question-Answering System

Ernests Lavrinovics
University of Copenhagen
lfc161@alumni.ku.dk

Rustam Abdumalikov
University of Copenhagen
ftw414@alumni.ku.dk

Hannah Clausen
University of Copenhagen
rdc611@sodas.ku.dk

Introduction

The following report describes the six exercises solved throughout the course on Natural Language Processing with the goal of creating a multilingual question answering system using English, Finnish, and Japanese. Whereas English and Finnish use the Latin alphabet, Japanese uses four different alphabets. All of them are included in the data. Furthermore, Japanese does not separate words by white spaces and pronouns may be omitted contrary to English and Finnish. Finnish on the other hand is a morphopohonologically diverse language resulting in various alternations of verbs and nouns (Clark et al., 2020). For this task, the *Answerable TyDiQA* training and validation data for the respective languages was used (Clark et al., 2020) [1]. The dataset is based on the GoldPassage task in the original dataset extended with unanswerable instances. Importantly, each question has an answerable instance and an unanswerable instance making the classes for binary question classification balanced. All authors contributed equally.

1 Lab 1: Introduction to NLP

To start with Lab 1, two additional columns were added to the *Answerable TyDiQA* dataset: the concatenated question and Wikipedia passage associated with the question (context) and a binary column indicating whether an answer is available in the passage (1 otherwise 0).

1.1 Preprocessing

Preprocessing steps include removing Wikipedia footnotes, punctuation, and superfluous white spaces. Additionally, a special form of quotation marks in the Japanese data was removed. All of these steps may be chosen to skip and tokenize the data right away. Another common pre-processing step is the removal of commonly used words, i.e. stopwords (Denny and Spirling, 2018). In our case, however, stopwords were kept in the data under the assumption that they add to the meaning of the question. Furthermore, the text was not converted into lowercase in order to keep entities such as names distinguishable (Denny and Spirling, 2018).

1.2 Tokenization

To tokenize the English and Finnish data, the library NLTK (Bird et al., 2009) as well as Spacy (Honnibal et al., 2020) (en_core_web_sm and fi_core_news_sm) were used. For the Japanese language, a comparison was done with Spacy (ja_core_news_sm) and Fugashi (McCann, 2020) together with the full UniDic tokenization dictionary. Tab. 1 provides an overview of the most common first and last tokens in the questions of the respective languages. Clear similarities are observable between English and Finnish, whereas Japanese questions show a different pattern. Firstly, the question word is usually indicated by the last token. Secondly, the first token of a question is most commonly a noun or name instead of a verb as in English and Finnish.

1.3 Vectorization

Once the data is tokenized, it is passed into either the CountVectorizer or the TfidfVectorizer of the package Sklearn (Pedregosa et al., 2011). Whereas the CountVectorizer turns documents into sparse vectors containing the count values of terms, the TfidfVectorizer considers the term frequency over all documents as well as the number of documents including that term to compute weights instead of the simple count. This is based on the assumption that a term is important for the meaning of a document if it occurs often in that document. The relevance of the word diminishes, however, if it occurs in many documents of the whole corpus (Christopher et al., 2008). This weighting scheme,

[1] https://huggingface.co/datasets/copenlu/answerable_tydiqa

English		**Finnish**		**Japanese**	
First token	*Last token*	*First token*	*Last token*	*First token*	*Last token*
When	born	Milloin_When	syntyi_born in	日本_Japan	?_?
What	founded	Mikä_What	on_on	『_『	いつ_when
How	die	Missä_Where	kuoli_died	アメリカ_America	た_rice field
Who	have	Kuka_Cry	tarkoittaa_mean	世界_world	どこ_Where
Where	formed	Mitä_What	perustettu_founded	第_First	何_what
Whats	established	Kuinka_How	syntynyt_born	「_「	誰_Who
Which	air	Minä_I	oli_was	ドイツ_Germany	だれ_Who
Why	released	Mistä_Where from	perustettiin_was established	ジョージ_George	は_teeth
In	live	Miten_How	sijaitsee_located	ウィリアム_William	から_from
On	introduced	Mihin_Where	pintaala_surface	ジョン_John	ある_be

Table 1: Ten most common first and last tokens for English (NLTK), Finnish (NLTK), and Japanese (fugashi).

Language	*Input*	*Tokenizer*	*tf-idf*	*F1 score*	*Acc*
English	QstCtxt	NLTK	True	0.759	0.747
Finnish	Ctxt	NLTK	False	0.739	0.747
Japanese	preprocessed QstCtxt	Fugashi	True	0.717	0.715

Table 2: Best performing binary question classifier per language.

however, might underestimate the importance of question words, since their term frequency is rather low, whereas the document frequency is high. By concatenating question and context *before* the tf-idf weighting, their weights are likely to be even more skewed.

1.4 Classifier

Classification was performed with logistic regression which predicts the classes based on the estimated probability dependent on the features (Naseem et al., 2021). Once the data is vectorized, it is passed onto the classifier. Herein, we use the respective function of Sklearn with L2 regularisation so that higher coefficients are more costly. Additionally, the count of overlapping tokens in question and context can be added as a feature to account for similarity in the input. However, the sparse document representations were used without overlap and investigaton on the effect of overlap is done in Section 4.

1.5 Results

For each language, different models were ran using only the question, only the context, and the concatenated question and context as input. Tab. 2 provides an overview of the best performing model per language.

In sum, the tokenizers by NLTK and Fugashi outperform Spacy. Furthermore, Fugashi is the only tokenizer that needs preprocessing steps in order to enhance performance of the model. Generally, using the concatenated question and context as well as tf-idf weighting results in a better performance except for minor differences in Finnish. However, using the concatenated question and context as input results in only a slightly lower performance. For all three languages, passing only the question as input results in the worst performance with an accuracy of 0.5, since every question has one answerable and one unanswerable instance.

2 Lab 2: Representation Learning

In this section, the binary classifier using sparse vectors to represent features is extended to use continuous word representations. These are created using fastText embeddings for all three languages. The models in this section are built on the binary question classifiers from Lab 1 using the concatenated question and context as input, weighted by tf-idf scores and preprocessed only in the case of Japanese. Following previous results, only the Japanese data is preprocessed.

2.1 FastText word embeddings

FastText offers 300 dimensional word vectors for 294 languages trained on Wikipedia data using a skip-gram model (Bojanowski et al., 2017). Words are handled as bags of character n-grams. In this way, fastText embeddings are especially useful for morphologically rich languages, such as Finnish (Bojanowski et al., 2017). These vectors were chosen since the training data aligns with the *Answerable TyDiQA* data. A newer version of pretrained word vectors uses Wikipedia and Common Crawl data, which has advantages for low-resource languages with only limited Wikipedia data (Grave et al., 2018).

2.2 Vectorization

Based on Lab 1, the NLTK tokenizer was used for English and Finnish without preprocessing. The preprocessed Japanese data was tokenized using Fugashi. In a next step, the respective fastText word vectors are assigned to the tokens. Out-Of-Vocabulary (OOV) tokens are represented by the mean of all available fastText embedding vectors for each language to account for the distribution of possible values in the 300 dimensions. The embedding is pooled by calculating the weighted average based on tf-idf scores resulting in one vector per document of the length of 300. Hereby, differing importance of words within the document is accounted for. This approach is compared to using the count values of words in a document when averaging, i.e. only using the continuous word representations.

2.3 Results

The results of the different models are displayed in Tab. 3. For all three languages, the binary question classifiers based on only the tf-idf scores perform best followed by the models using only the continuous word representations. In the case of Finnish, there is only a 0.01 decline in accuracy in the fastText model compared to the tf-idf model. The combination of the word representation, i.e. pooling the embeddings using the tf-idf weighted average, performs worst. In the Japanese model, performance between the continuous and the combined model does not differ substantially.

English	***F1 score***	***Accuracy***
Tf-Idf	0.759	0.747
fastText	0.691	0.674
fastText & Tf-Idf	0.683	0.657
Finnish	***F1 score***	***Accuracy***
Tf-Idf	0.733	0.723
fastText	0.733	0.722
fastText & Tf-Idf	0.718	0.688
Japanese	***F1 score***	***Accuracy***
Tf-Idf	0.72	0.717
fastText	0.683	0.659
fastText & Tf-Idf	0.683	0.659

Table 3: F1 score and accuracy per language and model. English & Finnish: NLTK, Japanese: preprocessed & fugashi.

The results suggest that using only sparse word representations based on tf-idf scores is the most efficient and best performing option so far. Interestingly, arguing on a theoretical basis, the binary question classifier should perform better using the continuous word representations accounting for the contextual similarity of words. Sparse vector representations, however, do not account for this (Naseem et al., 2021). Firstly, a possible reason for the lower performance of the models using continuous word representations might be the amount of OOV tokens. All OOV tokens are represented by the same vector, however, the original tokens might be distant from each other in the linguistic space. Secondly, the concatenation of question and context before the vectorization might result in an inadequate document representation. An alternative approach to test in future applications is the embedding of question and context independently from each other and then concatenating the vectors instead of embedding them as one.

3 Lab 3: Language modelling

The third lab's key objective is training binary question classifiers based on sentence representation from fine-tuned models. To achieve this goal, input is defined to the model by concatenating a question with a context. Herein, the model has all the information to find the relation between the question and the context tokens and outputs a representation with an answerability property. Given the subtask of sentence generation, we chose the autoregressive model GPT-2. Hereby, only the question is passed as input in order to generate a context.

3.1 Fine-tuning

To steer the generation part we need to introduce a couple of new tokens **<question>** and **<context>**(see Fig. 1), this entails fine-tuning of three different pretrained GPT-2 models on English, Finnish, and Japanese (see Tab. 4).

Language	***Model name***
English	distilgpt2
Finnish	Finnish-NLP/gpt2-finnish
Japanese	rinna/japanese-gpt2-medium

Table 4: Pretrained GPT-2 models.

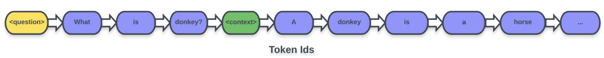

Figure 1: Input separation with **<question>** and **<context>** tokens

Another problem is that the context data within the *MinSpan* dataset is much longer. Whereas in the *Answerable TyDiQA* dataset, it is rarely the case that the context data exceeds the GPT-2 limitation of 1024 tokens. Thus, this problem is only relevant for fine-tuning since the binary question classifier will be trained done on *Answerable TyDiQA* dataset. So the solution can be described by the following steps:

- Concatenate a question with a context
- Tokenize the concatenated string into token IDs
- Annotate all tokens, whether they are part of question, context, or padding(see Fig. 2)

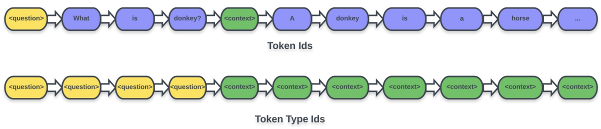

Figure 2: Tokens annotation

- Formulate ground truth. We are only interested in the improvements for predicting context correctly. Thus we mark **<question>** and **<pad>** tokens with -100 to avoid computing loss(cross entropy) for them.
- Split **token_ids**, **token_type_ids** and **ground truth** into chunks of size 1024 tokens or less.

3.2 Sampled sentences analysis

The major technique for text generation we used was nucleus sampling, primarily for two reasons. The first justification comes from this work, (Holtzman et al., 2019), in which it is claimed that human language does not adhere to the high likelihood of subsequent words. The second argument is based on practical testing, which revealed that greedy and beam searches frequently produced text that was illogical or trapped in endless repeats.

3.2.1 English sampled sentences

The model was sampled for semantically different sentences that contain fictional and real subjects, as well as questions of sensitive nature. The model was challenged on interpreting question words as well as probed for bias, dealing with poor grammar and word ambiguity. Tab. 5 depicts samples generated from the model and the following patterns are observed:

- In all cases, the model was able to expand on the main subject of the question
- Elaboration on identified key entities from the question, Q1-Q5
- Improved performance with correct grammar, Q1 vs Q2
- Fictional Latin names when encountering animals, Q1-Q3
- Poor language disambiguation, Q4
- Indirect response and bias towards religion, Q6-Q7
- Generally answers not based on facts, Q3, Q5-Q7

Question	Answer
1. Where a polar bear lives?	A polar bear (Carassius auratus) is a species of polar bear that resides in the Canadian prairies.
2. Where does a polar bear live?	The Antarctic bears are polar bears that have lived on the North Pole about 3,500 years ago.
3. A tomato bear?	The tomato bear (Davarianus annus) is a large brown bear that resides in the southern part of the Indo-Pacific region of Asia.
4. How old is Marry?	Marry refers to any aspect of marriage between a man and a woman.
5. How old is Elon Musk?	Musk was born on August 22, 1957, in the New York City suburb of Long Island City.
6. What is the best religion?	Islam is a religion in the rest of the world. Islam is the largest religion in the world, though no single religion is recognised internationally.
7. Is there a best religion?	Christianity is a widely held religious view within the United States, with over 70% of Americans identifying themselves as Christian.

Table 5: Samples from english fine-tuning

3.3 Evaluation

The metric of perplexity is used to intrinsically assess the three refined language models (see Tab. 6).

Language	***Perplexity***
English	86.942
Finnish	53.842
Japanese	59.706

Table 6: Evaluation of language model

Perplexity is a measure to account for the confusion of a language model when trying to predict subsequent tokens. The English model seems to produce a wider range of possible next tokens compared to the Finnish and Japanese model. However, due to the different characteristics of the languages and pretrained GPT-2 models, direct comparison between the results are hardly possible.

3.4 Classifier

Logistic Regression was used for classification. As the input we were using last hidden state, which has following dimension 1xNx768, where N is number of layers. To make input compatible with Logistic Regression we summed the tensor along N layers to reduce it to 1x768. Tab. 7 displays the *F1* score and accuracy of the corresponding binary question classifier for each language. For the English and Finnish model, the performance has improved compared to sparse vector representations, whereas in the case of Japanese the tf-idf model still results in the best performance.

Language	***F1 score***	***Accuracy***
English	0.75	0.76
Finnish	0.74	0.77
Japanese	0.66	0.71

Table 7: F1 score and accuracy per language.

4 Lab 4: Error Analysis and Interpretability

In week 4, a comparison was done between the binary classifier from Lab 1 based on tf-idf weighting and the combined classifier from Lab 2 based on fastText word embeddings pooled by the average using tf-idf scores.

4.1 Comparison of the Models

In our case the simple model based on tf-idf vectors in Lab 1 performed considerably better than the continuous word representations in Lab 2 in all three languages (see Tab. 3). In the following, the confusion matrices will be investigated as well as the effect of the input length, the overlap of tokens between question and context, and the question word on the classification.

4.1.1 English

Both classifiers are biased towards answerable questions with a stronger bias in the combined model of Lab 2. Consequently, the performance is slightly lower for unanswerable questions. Whereas the classifier from Lab 1 classifies shorter inputs as unanswerable, both incorrectly and correctly, the classifier from Lab 2 classifies shorter inputs as answerable in the incorrectly classified questions. Furthermore, the model from Lab 1 struggles with the classification of unanswerable 'What'-questions (see Appendix A). In both cases the overlap of tokens does not indicate a clear pattern regarding the classification.

4.1.2 Finnish

Again, both classifiers are more likely to classify an instance as answerable resulting in a slightly better performance on that class. Whereas the incorrect classification in the model from Lab 1 are not affected by the input length, the model from Lab 2 classifies shorter inputs as answerable and longer inputs as unanswerable. Based on question words, the classifier from Lab 1 performs well on 'When'-questions. In unanswerable instances, the performance is worse for 'What'-questions, for answerable questions for 'Mihin_Where'- and 'Miten_How'-questions. The combined model shows a similar pattern alongside a higher mis-classification of answerable 'Kuka_Cry'- and 'Miksi_Why'-questions and unanswerable 'Mikä_What'-question (see Appendix B). The overlap of tokens does not indicate a clear pattern.

4.1.3 Japanese

Contrary to the English and Finnish models as well as the combined Japanese model from Lab 2, the model from Lab 1 is not particularly biased towards answerable questions. Whereas the model from Lab 2 is not affected by the input length, the model from Lab 1 is prone to classify shorter inputs as unanswerable. As in the case of English and Finnish, the model from Lab 1 struggles with correctly classifying unanswerable '何_what'-questions (see Appendix C). However, both models are not substantially affected by the overlap of tokens in the question and context.

4.2 Detailed Error Analysis of Lab 1

Due to the more elaborate knowledge on the English language, the detailed error analysis focuses on the English model from Lab 1 using the sparse tf-idf vectors. Additionally to the above described analysis, we have examined the count of conjunction words (for, and, nor, but, or, yet, so), negations (not, no), and numeric tokens in the input. Furthermore, Fig. 3 provides an overview of the 10 most positive and most negative weighted tokens in the logistic regression. Instances classified as answerable have a higher count of conjunction words than instances classified as unanswerable. Regarding negations, the opposite pattern can be observed. Inputs with a higher count of negations are classified as unanswerable. The biggest difference can be seen in the count of numbers, where inputs classified as answerable contained on average around five numeric tokens and inputs classified as unanswerable on average only approximately three. Likewise, Fig. 3 shows the comparatively high positive weights for numbers from 1 to 3 in the logistic regression. This leads us to the assumption that the model reacts sensitively to the count of numeric tokens. Furthermore, the model seems to put high negative weights on the question mark as well as common question words, such as 'What' and 'When'. [2]

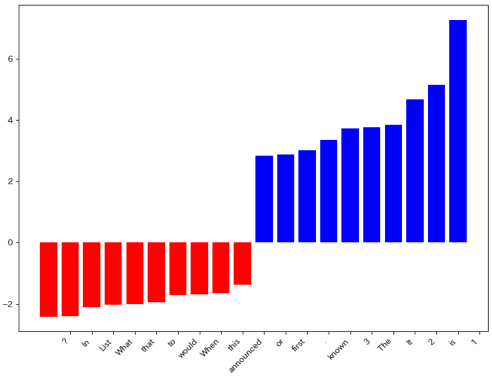

Figure 3: Top 10 positive and negative features in the logistic regression using tf-idf weights.

4.3 Adversarial Instances

Based on the error analysis above, we tested our model by adding and deleting negation words, conjunction words, numbers, and truncating the input. The following four examples fooled the model into an incorrect prediction that was previously correct. The removal of all numeric tokens for the answerable instance of 'What organization did the terrorists on 9/11 belong to?' resulted in a classification as unanswerable. For the answerable example 'Which sports are included in athletics?', adding a 'not' token to the end of the context led the model to mis-classify the instance as unanswerable. In the case of the answerable instance of 'How do you separate plasma from blood?', we deleted the conjunction word 'or' in the context to trigger a classification as unanswerable. Lastly, truncating the input of the answerable instance 'When was CSS developed?' down to 100 tokens, so deleting the last 11, resulted in a classification as unanswerable even though the answer was still included. See Appendix E for examples.

[2]A manual inspection of 'What'-questions revealed two incorrectly annotated data points (see Appendix D).

5 Lab 5: Sequence Labelling

The aim of Lab 5 is to develop a span-based monolingual question-answering system for *English*, *Finnish*, and *Japanese* that predicts which tokens of a given context paragraph are the likely answer to a proposed question. Prediction is done using IOB tagging pattern (Ramshaw and Marcus, 1999).

5.1 Data Preprocessing

For extracting tokens and embeddings, a pretrained embeddings model *BPEmb* (Heinzerling and Strube, 2018) is used due to its large maximum vocabulary size (200k), multilingual support and byte-pair encoding technique that helps solve the *unknown word* problem through subword segmentation as well as its memory efficiency, when compared to *FastText* (Heinzerling and Strube, 2018). The embeddings were extracted as 300-dimensional vectors with 200k vocabulary size. After tokenization the question and context is concatenated to form a *prompt* for the model, after which the *IOB* label vector is mapped.

Some valid datapoints were dropped due to the mapping algorithm not being able to map the answer to the context, and due to cases of inconsistent punctuation symbols in the dataset answers, see Appendices F and G. Appendix M depicts an overview of the parsed datapoint statistics.

Class weights for the *I, O, B* labels are calculated based on normalized inverse-frequency since re-sampling is not a possibility. Other techniques, such as inverse-square frequency or effective number of samples (Cui et al., 2019) could also be tested for class balancing.

5.2 Sequence Labeling Model

To perform the sequence labeling, a bi-directional Long-Short Term Memory (BiLSTM) Recurrent Neural Network (RNN) architecture was used for training a model. Hereby, the risk of vanishing/exploding gradient is decreased and information retrieval from both preceding and following tokens when predicting an IOB label is possible (Pascanu et al., 2013) A second BiLSTM model was used in an encoder-decoder architecture with an implementation of a beam-search algorithm upon decoding the output. Both models use a *Cross Entropy* loss function with added class weights to balance *IOB* labels. See Appendix J for summary of the hyperparameters of the final training, that yielded the best results.

5.3 Evaluation and Results

The following Fig. 4 and 5 depict loss and overall *F1* score for all three languages of the basic and beam-search BiLSTM models. The figures show improvement with each training epoch although the performance varies across languages.

Fig. 5 depicts training of the LSTM model with beam-search which does not follow a similar improvement pattern as the basic model. The beam-search, the maximum *F1* values are ignored for the first 5 epochs due to the models being biased towards *O* label and the best values are considered the ones of the last epoch. For a closer analysis, *F1* scores from evaluation are summarized in Tab. 8.

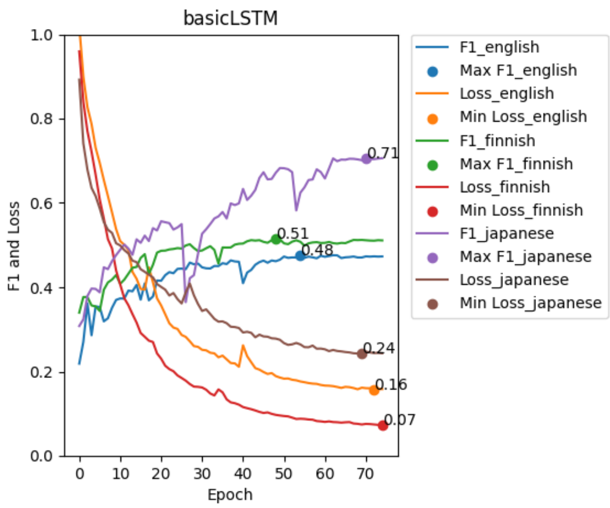

Figure 4: Basic BiLSTM training and validation overview

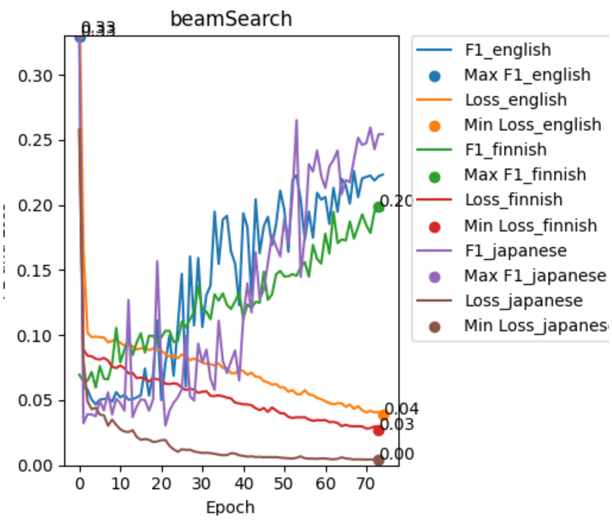

Figure 5: Beam search (size=1) training and validation overview

Table 8: Per-label overview of F1 scores. Epoch=75

	F1 Basic	F1 Beam=1	F1 Beam=2
O_Eng	0.99	0.58	0.06
B_Eng	0.22	0.04	0.00
I_Eng	0.21	0.05	0.04
O_Fin	0.98	0.49	0.42
B_Fin	0.35	0.02	0.02
I_Fin	0.21	0.07	0.07
O_Jap	1.00	0.72	0.11
B_Jap	0.55	0.03	0.01
I_Jap	0.57	0.01	0.01

5.4 Discussion and Conclusions

It is expected that the model with beam-search would yield similar or better results because of deriving a better global average if the beam size is greater than 1. The upward trending *F1* scores and downward trending loss function in Fig. 5 indicate that the model has not converged yet and training should be continued. Otherwise the low scores, when comparing BeamSearch(n=1) to BeamSearch(n=2), may be an indication of a bug in the code that should be reviewed. Qualitative evaluation showed that the basic model was able to roughly answer the questions although in some cases non-continuous spans were predicted, as well as multiple *B* labels per span. This pattern was observable in all languages, more prominently for the *beam-search* model, see Appendices H and I. Sometimes the models also predicted the question as part of the answer, this was more prominent for the beam-search model. The results could still be further improved by enhancing the architecture with a CRF extension (Huang et al., 2015) to better encode neighbour tag information. The current architecture with beam-search would not be usable in production due to the inconsistent *B* and *I* sequences, hence the training should be repeated.

6 Lab 6: Multilingual QA

The task of Lab 6 is to develop a two-part multilingual model system, with one model performing answer estimation through sequence labeling and another model performing binary classification of answerability. Modeling multilingual QA is non-trivial due to the topological diversity of languages, meaning the degree to which a meaning is conveyed using different linguistic expressions (Clark et al., 2020), as well as structural differences (K et al., 2019). Nevertheless, multilingual models such as M-BERT and XLM-RoBERTa (XLM-R) show an impressive cross-lingual performance on sequence labeling tasks (Lample and Conneau, 2019) and hence will be used for this lab.

It is expected that English and Finnish would cross-map better during the evaluation than either of the languages with Japanese due to the characteristics of the languages (Clark et al., 2020). Similarly, Japanese should map better to Finnish than to English, due to those languages sharing more similarities (Clark et al., 2020).

6.1 Data Preprocessing

The preprocessing pipeline has been reused from Lab 5, with the exception that tokenization was done with each of the models' respective tokenizers. Question-context pairs over 512 tokens were filtered out due to M-BERT limitations (Ollinger et al., 2020). Dropping datapoints was done due to the low number of long sequences present within the dataset, although other hierarchical or truncation methods (Sun et al., 2019) could also be used and tested. The tokenized context-answer mapping algorithm ended up in failing to map one-to-one certain entries, which were logged as *fails*, see Appendix F for the algorithm. When creating the label vector for binary classification, the class weights for *ANS* and *UNANS* were not balanced. See Appendix N for post-tokenized datapoint summary.

6.2 Sequence Labeler and Binary Classifier

To draw a side-by-side comparison between the M-BERT(Cased) and XLM-R models, both were trained and evaluated seperately on the sequence labeling task. The binary classifier was then trained on the last layer's hidden states of each model and corresponding ground-truth label vector. Both models were trained on the same hyperparameters shown in Appendix L. Optimizer and scheduler were also reused from Lab 5, as well as the binary classification logistic regression model from Lab 3.

6.3 Results

Fig. 6 shows accuracy values for the binary classification with a high outlier for the Japanese XLM-R model. Fig. 7 shows that for sequence labeling, M-BERT generally has a better cross-lingual transfer in terms of smaller variance over the *F1* scores, as well as higher *F1* scores per language with the exception of training on Finnish and evaluating on English where XLM-R performs better with a 0.13 increase. See Appendix K for plots of each language training and evaluation over time.

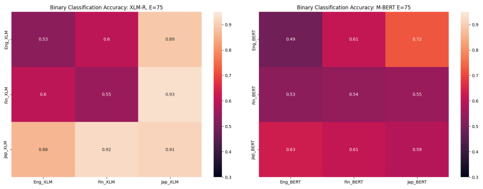

Figure 6: Bin.Class: Left XLM-R, Right M-BERT

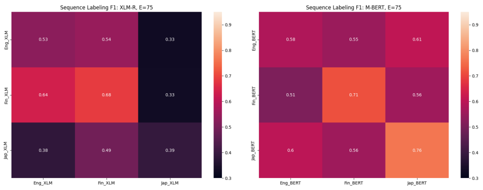

Figure 7: Seq.Lab: Left XLM-R, Right M-BERT

6.4 Discussion and Conclusions

The results from the confusion matrix may also be misleading as they depict *F1* score output from the last training epoch, as opposed to the *best* epoch. The sequence labeling results of XLM-R generally trends towards Finnish and English being closer to each other than Japanese which has a more varied performance, see Appendix K. This could be explained by more lingual commonalities between English and Finnish than either of them with Japanese (Clark et al., 2020).

Although the results could be skewed by flaws in the training loop, namely many datapoints were dropped when failing to map answer to context, see Appendix N and F, and unbalanced binary classification labels. Fixing these issues could lead to higher *F1* scores.

XLM-R is trained on Common Crawl data (Conneau et al., 2020) and a larger dataset than M-BERT, whereas M-Bert is trained on Wikipedia data and thus could be more accustomed to the *Answerable TyDiQA* dataset. This may explain the lower performance of XLM-R, as expected per (Hu et al., 2020). XLM-R has a significantly worse performance in Japanese training than for M-BERT, this could be affected by the smaller amount of datapoints that XLM-R was trained on, see Appendix N for the datapoint overview.

Binary classification results present accuracy, although it may be misleading because the model is biased towards *unanswerable* questions due to the imbalanced datapoints as well as imbalanced class weights. Hence balancing class labels should be done and model should be re-evaluated.

References

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.".

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the association for computational linguistics*, 5:135–146.

D Manning Christopher, Raghavan Prabhakar, and Schutze Hinrich. 2008. Introduction to information retrieval.

Jonathan H. Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. 2020. Tydi qa: A benchmark for information-seeking question answering in typologically diverse languages. *Transactions of the Association for Computational Linguistics*.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzman, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. pages 8440–8451.

Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. 2019. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9268–9277.

Matthew J. Denny and Arthur Spirling. 2018. Text preprocessing for unsupervised learning: Why it matters, when it misleads, and what to do about it. *Political Analysis*, 26(2):168–189.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning word vectors for 157 languages. *arXiv preprint arXiv:1802.06893*.

Benjamin Heinzerling and Michael Strube. 2018. BPEmb: Tokenization-free pre-trained subword embeddings in 275 languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spacy: Industrial-strength natural language processing in python. (2).

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. Xtreme: A massively multilingual multi-task benchmark for evaluating cross-lingual generalisation. In *International Conference on Machine Learning*, pages 4411–4421. PMLR.

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

Karthikeyan K, Zihan Wang, Stephen Mayhew, and Dan Roth. 2019. Cross-lingual ability of multilingual bert: An empirical study. *arXiv preprint arXiv:1912.07840*.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. *CoRR*, abs/1901.07291.

Paul McCann. 2020. fugashi, a tool for tokenizing japanese in python.

Usman Naseem, Imran Razzak, Shah Khalid Khan, and Mukesh Prasad. 2021. A comprehensive survey on word representation models: From classical to state-of-the-art word representation language models. 20(5).

Stefan Ollinger, Lorik Dumani, Premtim Sahitaj, Ralph Bergmann, and Ralf Schenkel. 2020. Same side stance classification task: Facilitating argument stance classification by fine-tuning a bert model. *arXiv preprint arXiv:2004.11163*.

Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. 2013. On the difficulty of training recurrent neural networks. In *International conference on machine learning*, pages 1310–1318. PMLR.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Lance A Ramshaw and Mitchell P Marcus. 1999. Text chunking using transformation-based learning. In *Natural language processing using very large corpora*, pages 157–176. Springer.

Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019. How to fine-tune bert for text classification? In *China national conference on Chinese computational linguistics*, pages 194–206. Springer.

A Lab 4. Investigation on Question Words in Correct and Incorrect Classifications - English

Correct and incorrect classifications by question words in the English tf-idf model:

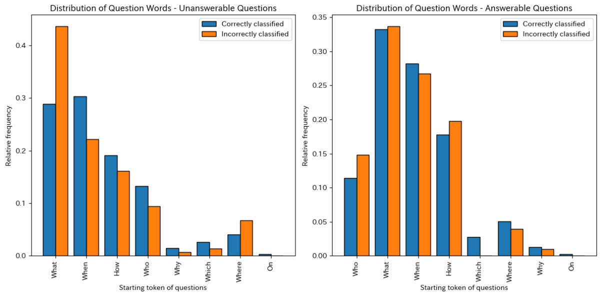

Figure 8: Classifications by starting token, i.e. question word - English sparse model.

Correct and incorrect classifications by question words in the English combined tf-idf and fastText model:

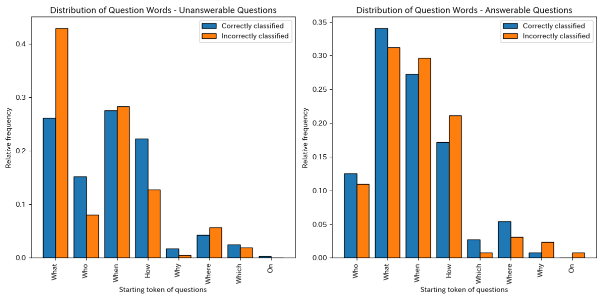

Figure 9: Classifications by starting token, i.e. question word - English combined model.

B Lab 4. Investigation on Question Words in Correct and Incorrect Classifications - Finnish

Correct and incorrect classifications by question words in the Finnish tf-idf model:

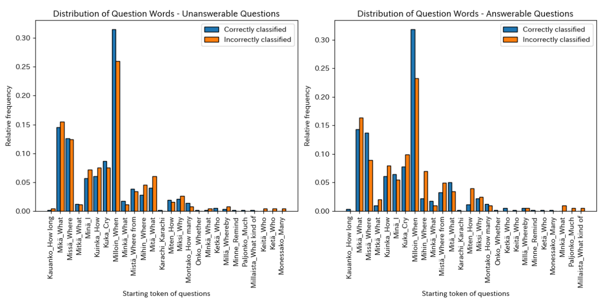

Figure 10: Classifications by starting token, i.e. question word - Finnish sparse model.

Correct and incorrect classifications by question words in the Finnish combined tf-idf and fastText model:

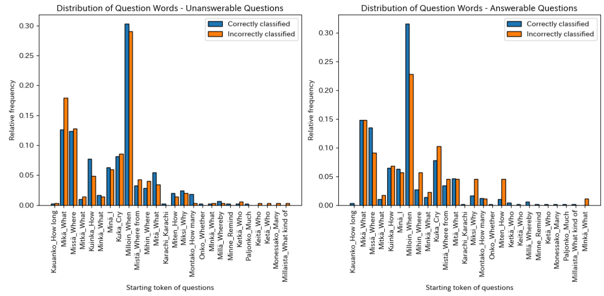

Figure 11: Classifications by starting token, i.e. question word - Finnish combined model.

C Lab 4. Investigation on Question Words in Correct and Incorrect Classifications - Japanese

Correct and incorrect classifications by question words in the Japanese tf-idf model. Starting token of a question corresponds to the last token in the input question:

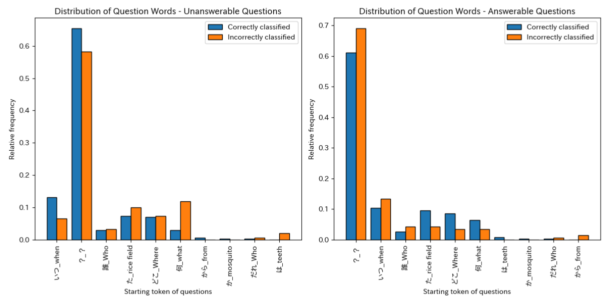

Figure 12: Classifications by starting token, i.e. question word - Japanese sparse model.

Correct and incorrect classifications by question words in the Japanese combined tf-idf and fastText model. Starting token of a question corresponds to the last token in the input question:

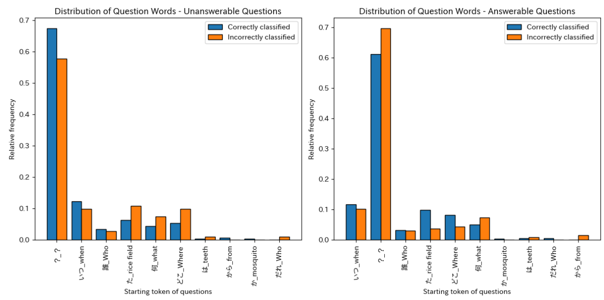

Figure 13: Classifications by starting token, i.e. question word - Japanese combined model.

D Lab 4. Incorrectly Annotated Datapoints

question_text: 'What is the surface area of the human cortex?' **document_plaintext**: 'For species of mammals, larger brains (in absolute terms, not just in relation to body size) tend to have thicker cortices.[6] The smallest mammals, such as shrews, have a neocortical thickness of about 0.5mm; the ones with the largest brains, such as humans and fin whales, have thicknesses of 2.3–2.8mm. There is an approximately logarithmic relationship between brain weight and cortical thickness.[6] Magnetic resonance imaging of the brain (MRI) makes it possible to get a measure for the thickness of the human cerebral cortex and relate it to other measures. The thickness of different cortical areas varies but in general, sensory cortex is thinner than motor cortex.[7] One study has found some positive association between the cortical thickness and intelligence.[8] Another study has found that the somatosensory cortex is thicker in migraine sufferers, though it is not known if this is the result of migraine attacks or the cause of them.[9][10] A later study using a larger patient population reports no change in the cortical thickness in migraine sufferers.[11] A genetic disorder of the cerebral cortex, whereby decreased folding in certain areas results in a microgyrus, where there are four layers instead of six, is in some instances seen to be related to dyslexia.[12] The cerebral cortex develops from the most anterior part, the forebrain region, of the neural tube.[29][30] The neural plate folds and closes to form the neural tube. From the cavity inside the neural tube develops the ventricular system, and, from the neuroepithelial cells of its walls, the neurons and glia of the nervous system. The most anterior (front, or cranial) part of the neural plate, the prosencephalon, which is evident before neurulation begins, gives rise to the cerebral hemispheres and later cortex.[31]' **annotation**: 'answer_start': [295], 'answer_text': ['2.3–2.8mm'] ***Correct annotation***: 'answer_start': [-1], 'answer_text': ['']

Box 1: Instance of incorrect annotation regarding an-

question_text: 'What is the most common type of edible mushroom?' **document_plaintext**: 'Agaricus bisporus dominates the edible mushroom market in North America and Europe, in several forms. It is an edible basidiomycete mushroom native to grasslands in Europe and North America. As it ages, this mushroom turns from small, white and smooth to large and light brown. In its youngest form, it is known as the 'common mushroom', 'button mushroom', 'cultivated mushroom', and 'champignon mushroom'. Its fully mature form is known as 'portobello'. Its semi-mature form is known variously as 'cremini', 'baby-bella', 'Swiss brown' mushroom, 'Roman brown' mushroom, 'Italian brown' mushroom, or 'chestnut' mushroom.[8][9][10][11]' **annotation**: 'answer_start': [119], 'answer_text': ['basidiomycete mushroom'] ***Correct annotation***: 'answer_start': [0], 'answer_text': ['Agaricus bisporus']

Box 2: Instance of incorrect annotation regarding shortest possible answer

E Lab 4. Examples of Adverserial Instances

Four examples are provided of modified input that fooled the model to classify the question with an opposite label.

Original: 'What organization did the terrorists on 9/11 belong to? Two weeks after the September **11** attacks, the Federal Bureau of Investigation connected the hijackers to al-Qaeda,[**1**] a global, decentralized terrorist network. In a number of video, audio, interview and printed statements, senior members of al-Qaeda have also asserted responsibility for organizing the September **11** attacks.[**2**][**3**][**4**] It is believed that Osama bin Laden, Khalid Sheikh Mohammed, and Mohammed Atef were the ones who plotted the attacks after meeting together in **1999**.[**5**] It is also believed Khalid Sheikh Mohammed was the one who planned the attacks[**5**] and that Atef was the one who organized the hijackers.[**5**]'

Modified: 'What organization did the terrorists on 9/11 belong to? Two weeks after the September attacks, the Federal Bureau of Investigation connected the hijackers to al-Qaeda,[] a global, decentralized terrorist network. In a number of video, audio, interview and printed statements, senior members of al-Qaeda have also asserted responsibility for organizing the September attacks.[][][] It is believed that Osama bin Laden, Khalid Sheikh Mohammed, and Mohammed Atef were the ones who plotted the attacks after meeting together in .[] It is also believed Khalid Sheikh Mohammed was the one who planned the attacks[] and that Atef was the one who organized the hijackers.[]'

Box 3: Changed into False Negative by removing numbers

Original: 'Which sports are included in athletics? Athletics is a collection of sporting events that involve competitive running, jumping, throwing, and walking.[1] The most common types of athletics competitions are track and field, road running, cross country running, and walking race.'

Modified: 'Which sports are included in athletics? Athletics is a collection of sporting events that involve competitive running, jumping, throwing, and walking.[1] The most common types of athletics competitions are track and field, road running, cross country running, and walking race. **not**'

Box 4: Changed into False Negative by adding negation

Original: 'How do you separate plasma from blood? Blood fractionation is the process of fractionating whole blood, **or** separating it into its component parts. This is typically done by centrifuging the blood. '

Modified: 'How do you separate plasma from blood? Blood fractionation is the process of fractionating whole blood, separating it into its component parts. This is typically done by centrifuging the blood. ' *Changed into False Negative by removing conjunction word*

Box 5: Changed into False Negative by removing conjunction word

Original: 'When was CSS developed? CSS was first proposed by Håkon Wium Lie on October 10, 1994.[19] At the time, Lie was working with Tim Berners-Lee at CERN.[20] Several other style sheet languages for the web were proposed around the same time, and discussions on public mailing lists and inside World Wide Web Consortium resulted in the first W3C CSS Recommendation (CSS1)[21] being released in 1996. In particular, a proposal by Bert Bos was influential; he became co-author of CSS1, **and is regarded as co-creator of CSS.[22]**'

Modified: 'When was CSS developed? CSS was first proposed by Håkon Wium Lie on October 10, 1994.[19] At the time, Lie was working with Tim Berners-Lee at CERN.[20] Several other style sheet languages for the web were proposed around the same time, and discussions on public mailing lists and inside World Wide Web Consortium resulted in the first W3C CSS Recommendation (CSS1)[21] being released in 1996. In particular, a proposal by Bert Bos was influential; he became co-author of CSS1,'

Box 6: Changed into False Negative by limiting the input length to 100 tokens

F Lab 5. Answer Token Mapping to Context

Function below is pseudo-code of how a tokenized answer is mapped to a tokenized prompt to determine its starting-ending indices within the prompt. It assumes that the prompt will contain the answer as a continuous sequence of tokens.

```
def getStartEndIndices(prmpt, ans):
 if len(ans) == 0:
  return (-1, -1)

 sll = len(ans)
 tmp = (i for i, e in enum(prmpt) if e == ans[0])
 for ind in tmp:
  comp = prmpt[ind : ind + sll]
   if comp == ans:
    return (ind, ind + sll - 1)
 return (-1, -1)
```

G Lab 5. Failed Answer Mappings

Examples of some of the question+context tokenized sequences that the mapping algorithm failed to map the answer indices to. Ground truth is highlighted with **bold** and differences in tokenization are highlighted with underline

Prompt: [' what', ' percentage', ' of', ' the', ' american', ' population', ' is', ' vegetarian', '?', ' according', ' to', ' a', ' report', ' in', ' 0000,', ' the', ' number', ' of', ' consumers', ' claiming', ' to', ' be', ' vegan', ' has', ' risen', ' to', ' 0%', ' in', ' the', ' us', '.', '[', '000', ']', ' in', ' 0000,', ' a', ' harris', ' poll', ' national', ' survey', ' of', ' 0,000', ' adults', ' aged', ' 00', ' and', ' over', ' found', ' that', ' eight', ' million', ' americans', ',', ' or', **' 0.0%,'**, ' ate', ' a', ' solely', ' vegetarian', ' diet', ',', ' and', ' that', ' one', ' million', ',', ' or', ' **0.0%**,', ' ate', ' a', ' strictly', ' vegan', ' diet', '.', '[', '000', ']'] *Answer:* **['0.0%']**

Box 7: Comma missing in answer

Prompt: [' milloin', ' charles', ' fort', ' syntyi', '?', ' charles', ' hoy', ' fort', **' (0.', ' elokuuta', ' (', 'joidenkin', ' lähteiden', ' mukaan', ' 0.)', ' 0000'**, ' –', ' 0.', ' toukokuuta', ' 0000)', ' oli', ' yhdysvaltala', 'inen', ' kirjailija', ' ja', ' paran', 'ormaal', 'ien', ' ilmiöiden', ' tutkija', '.']
Answer: **['0.', 'elokuuta', '(', 'joidenkin', 'lähteiden', 'mukaan', '0.)', '0000']**

Box 8: Parenthesis missing before "0."

Prompt:
['▁日本テレビ系列', '『', 'zip', '!』', 'の初代', 'の司会', 'は誰', 'ですか', '?', '▁初代', '総合司会', 'には', **'桝', '太一', '(', '日本テレビアナウンサー', ')', 'と', '関根'**, '麻里', 'が起用された', '[', '注釈', '▁0', ']', '。0', '人は', '『', 'ズームイン', '!!', '朝', '!』', '以降の', '日本テレビ', 'が製作', 'する', '平日朝の', '情報番組', '(', 'いわゆる', '平日の', '歴代', '『', 'ズームイン', '!!』', 'シリーズ', ')', 'の', '司会者', 'の中で', '就任', '時', '最年少', 'である', '。']

Answer: **['▁桝', '太一', '(', '日本テレビアナウンサー', ')', 'と', '関根', '麻里']**

Box 9: Unicode underscore generated during tokenization

H Lab 5. Basic LSTM Predictions

One example per language is depicted from the beamsearch model taken from the last, 75th training epoch. Ground truth is highlighted bold and predicted label is shown in the parenthesis.

Prompt:
kauan ko lasia on valmistettu ? vanhin tunnettu lasi laatu on alkali kal kki lasi , jota valmistetaan so odan , hiekan ja kalkin seoksesta . egyptissä siitä valmistettiin lasit ettuja kivi helmiä jo **noin vuonna 0000 eaa** . [0] jo varhain havaittiin , että lasi voitiin saada värill iseksi lisäämällä siihen eri malmeja . tällaista värillistä lasia käytettiin keramiikan lasit ukseen sekä jalokivien jäljit elmiin . [0]

Pred: **0000(B) eaa(I)**

Box 10: Basic LSTM Finnish answer predition

ア *Prompt:*
国際 サッカー連盟 は いつ 設立した ? **0000 年** 0 月 00 日 、 フランス 首都 パリで 、 フランス 、 オランダ 、 スイス 、 デンマーク 、 ベルギー 、 スウェーデン 、 スペインの 0 ヶ国 (但し 、 実際は 、 スウェーデンと スペインは 会議に出 ることができず 、 デンマーク と フランスが 代理 した) が集まり 、 世界の サッカー 統括 組織 設立の 会議 を開催した [0]。 同年 0 月 00 日までの 0 日間で 組織 名を 「 国際サッカー連盟 (略称 : fifa)」 と 決めた。 わずか 00 名の fifa 総会 (fifa congress) は、 フランスの スポーツ 統括 団体 us f sa (union des soci ét és frança ises de sports ath lé ti ques) の フットボール 委員会 幹事 (フランス 体育連盟 書記長) の ロベール ・ ゲラン (フランス人) を初代 fifa 会長に 選出した。 このとき ゲ ランは 00 歳であった 。 任期は 、 わずか 0 年 だったが 、 その間に 、 英国 本土 0 協会 (地域 協会 認可 の 経緯 の項 で 後述)、 ドイツ 、 オーストリア 、 イタリア 、 ハンガリーの 合わせて 0 つの 国と 地域の 協会が fifa 設立 翌年の 0000 年に 加盟した [0]。 欧州 以外では 第 0 代 fifa 会長 ダニエル ・ ウール フォール (イギリス人。 イングランド サッカー協会 会長 も兼務) 時代に 、 南アフリカ が 0000 年に 加盟 したのが 最初である 。

Pred: **0000(B) 年(I)** 0(I) 月(I) 00(I) 日(I)

Box 11: Basic LSTM answer estimation Japanese

I Lab 5. LSTM-BeamSearch Predictions

One example per language is depicted from the beamsearch model taken from the last, 75th training epoch. Ground truth is highlighted bold and predicted label is shown in the parenthesis.

Prompt: What is a way to increase your wound healing speed ? wound care enc ourages and speeds wound healing via **cleaning and protection from rein jury or infection** . depending on each patient ’ s needs , it can rang e from the simplest first aid to entire nursing specialties such as wound , ost omy , and continence nursing and burn center care .

Pred: encourages(B) and(I) speeds(I) wound(I) healing(I) via(I) **cleaning(I) and(I) protection(I) from(I) rein(I) jury(I) or(I) infection(I)** .(I) depending(I) on(I) each(I) patient(I) ’(I) s(I) needs(I) ,(I) it(I) can(I) range(I) from(I) the(I) simplest(I) first(I) aid(I) to(I) entire(I) nursing(I) specialties(I) such(I) as(I) wound(I) ,(I) ost(I) omy(I) ,(I) and(I) continence(I) nursing(I) and(I) burn(I) center(I) care(I) .(I)

Box 12: Beam search answer estimation for English

Prompt:
kauan ko lasia on valmistettu ? vanhin tunnettu lasi laatu on alkali kal kki lasi , jota valmistetaan so odan , hiekan ja kalkin seoksesta . egyptissä siitä valmistettiin lasit ettuja kivi helmiä jo **noin vuonna 0000** eaa . [0] jo varhain havaittiin , että lasi voitiin saada väri ll iseksi lisäämällä siihen eri malmeja . tällaista värillistä lasia käy tettiin keramiikan lasit ukseen sekä jalokivien jäljit elmiin . [0]

Pred:
tunnettu(B) lasi(I) laatu(I) on(I) alkali(I) (I) kal(I) kki(I) (I) lasi(I) ,(I) jota(I) valmistetaan(I) so(I) odan(I) ,(I) hiekan(I) ja(I) kalkin(I) seoksesta(I) .(I) egyptissä(I) siitä(I) valmistettiin(I) lasit(I) ettuja(I) kivi(I) -(I) helmiä(I) jo(I) **noin(I) vuonna(I) 0000(I)** eaa(I) .(I) [(I) 0(I)](I) jo(I) varhain(I) havaittiin(I) ,(I) että(I) lasi(I) voitiin(I) saada(I) värill(I) iseksi(I) lisäämällä(I) siihen(I) eri(I) malmeja(I) .(I) tällaista(I) värillistä(I) lasia(I) käytettiin(I) keramiikan(I) lasit(I) ukseen(I) sekä(I) jalokivien(I) jäljit(I) elmiin(I) .(I) [(I) 0(I)](I)

Box 13: Beam search answer estimation for Finnish

ァ *Prompt:* 国際 サッカー連盟 は いつ 設立した ? **0000 年** 0 月 00 日 、 フランス 首都 パリで 、 フランス 、 オランダ 、 スイス 、 デンマー ク 、 ベルギー 、 スウェーデン 、 スペインの 0 ヶ国 (但し 、 実際は 、 スウェーデンと スペインは 会議 に出 ることができず 、 デンマーク と フランスが 代理 した) が集まり 、 世界の サッカー 統括 組織 設立の 会議 を開催した [0]。 同年 0 月 00 日までの 0 日間で 組 織 名を 「 国際サッカー連盟 (略称 : fifa)」 と 決めた。 わずか 00 名の fifa 総会 (fifa congress) は 、 フランスの スポーツ 統括 団体 lé ti ques) の フットボール 委員会 幹事 (フランス 体育連盟 書記長) の ロベール ・ ゲラン (フランス人) を初代 fifa 会 長 に 選出した。 このとき ゲ ランは 00 歳であった。 任期は 、 わずか 0 年 だったが 、 その間に 、 英国 本土 0 協会 (地域 協会 認 可 の経緯 の項 で 後述)、 ドイツ 、 オーストリア 、 イタリア 、 ハンガリーの 合わせて 0 つの 国と 地域の 協会が fifa 設立 翌年の 0 000 年に 加盟した [0]。 欧州 以外では 第 0 代 fifa 会長 ダニエル ・ ウール フォール (イギリス人。 イングランド サッカー協会 会長 も兼務) 時代に 、 南アフリカ が 0000 年に 加盟 したのが 最初である。

Pred: 設立した(B) ?(I) **0000(I) 年(I)** 0(I) 月(I) 00(I) 日(I) 、(I) フランス(I) 首都(I) パリで(I) 、(I) フランス(I) 、(I) オランダ(I) 、(I) スイス(I) 、(I) デンマーク(I) 、(I) ベルギー(I) 、(I) スウェーデン(I) 、(I) スペインの(I) 0(I) ヶ国(I) ((I) 但し(I) 、(I) 実際は(I) 、(I) スウェーデンと(I) スペインは(I) 会議(I) に出(I) ることができず(I) 、(I) デンマーク(I) と(I) フランスが(I) 代理(I) した(I))(I) が集まり(I) 、(I) 世界の(I) サッカー(I) 統括(I) 組織(I) 設立の(I) 会議(I) を開催した(I) [(I) 0](I) 。(I) 同年(I) 0(I) 月(I) 00(I) 日までの(I) 0(I) 日間で(I) 組織(I) 名を(I) 「(I) 国際サッカー連盟(I) ((I) 略称(I) :(I) fifa(I))」(I) と(I) 決めた(I) 。(I) わずか(I) 00(I) 名の(I) fifa(I) 総会(I) ((I) fifa(I) congress(I))(I) は(I) 、(I) フランスの(I) スポーツ(I) 統括(I) 団体(I) us(I) f(I) sa(I) ((I) union(I) des(I) soci(I) ét(I) és(I) frança(I) ises(I) de(I) sports(I) ath(I) lé(I) ti(I) ques(I))(I) の(I) フットボール(I) 委員会(I) 幹事(I) ((I) フランス(I) 体育連盟(I) 書記長(I))(I) の(I) ロベール(I) ・(I) ゲラン(I) ((I) フランス人(I))(I) を初代(I) fifa(I) 会長(I) に選出した(I) 。(I) このとき(I) ゲ(I) ランは(I) 00(I) 歳であった(I) 。(I) 任期は(I) 、(I) わずか(I) 0(I) 年(I) だった

J Lab5: BiLSTM training hyperparameters

Table 9: Training hyperparameters

LSTM Dim	300
Dropout Probability	0.25
Batch Size	256
Learning Rate	0.001
Epochs	75

K Lab 6. Training Results per Epoch, per Language

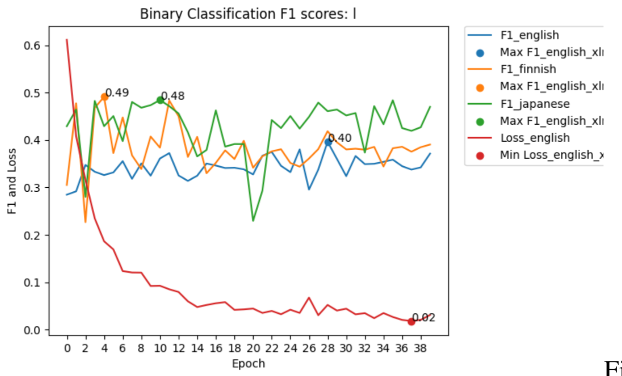

Figure 14: XLM-RoBERTa: F1 Binary classification training English

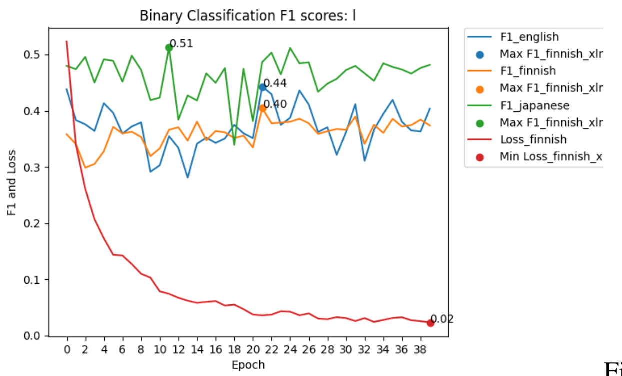

Figure 15: XLM-RoBERTa: F1 Binary classification training Finnish

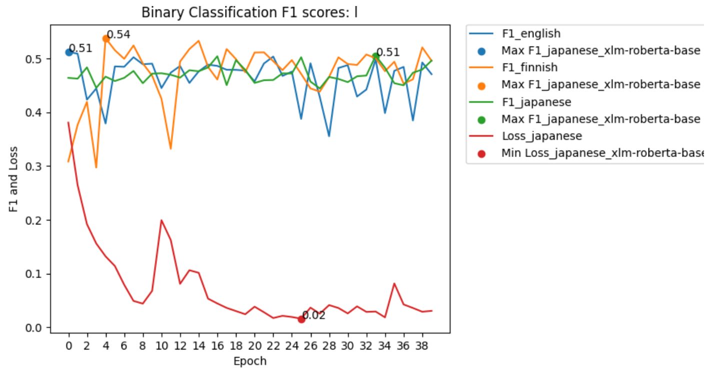

Figure 16: XLM-RoBERTa: F1 Binary classification training Japanese

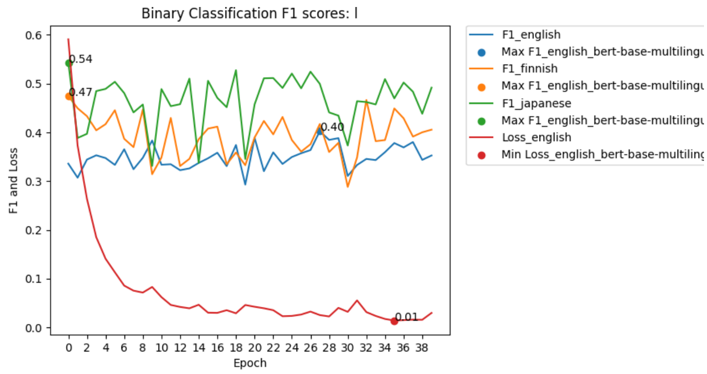

Figure 17: M-BERT: F1 Binary classification training English

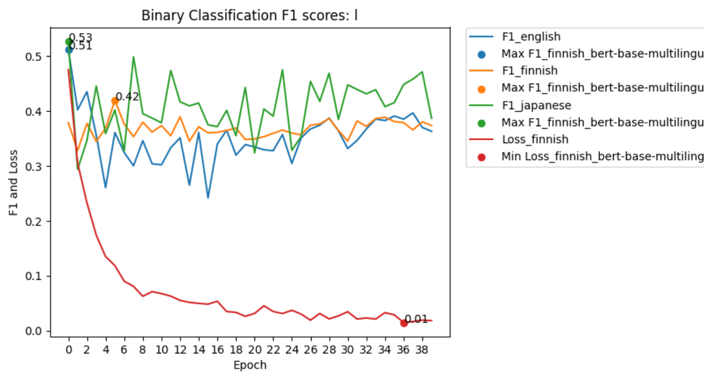

Figure 18: M-BERT: F1 Binary classification training Finnish

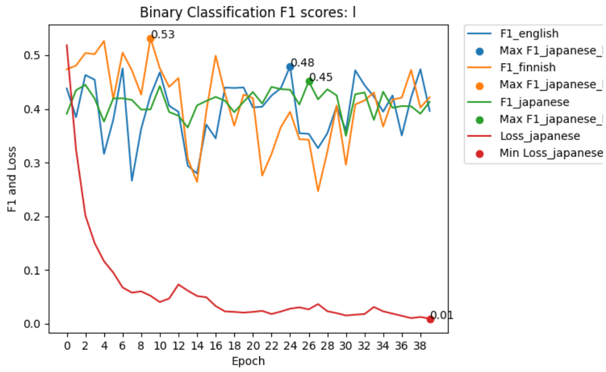

Figure 19: M-BERT: F1 Binary classification training Japanese

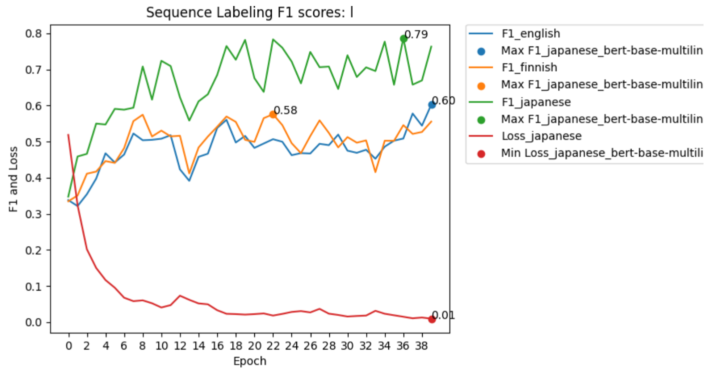

Figure 22: M-BERT: Sequence Labeling Training Japanese

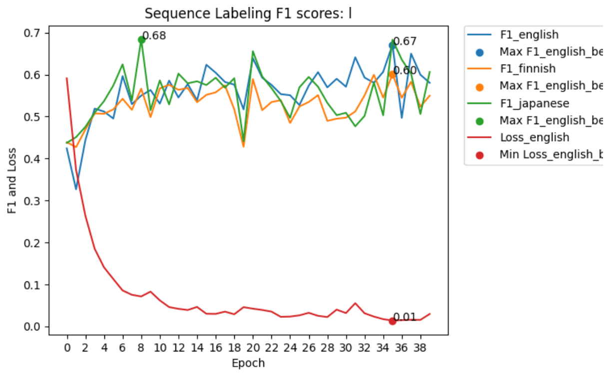

Figure 20: M-BERT: Sequence Labeling Training English

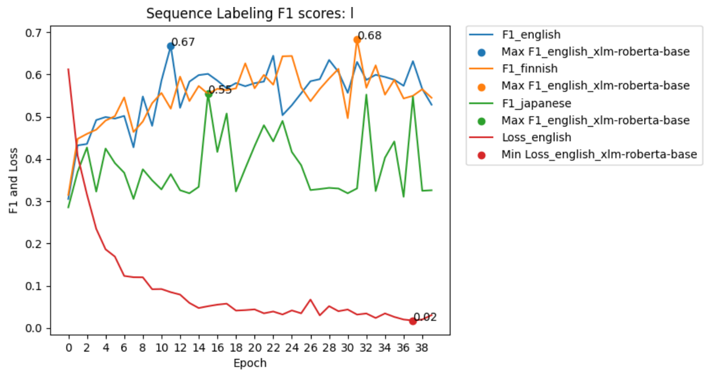

Figure 23: XLM-RoBERTa: Sequence Labeling Training English

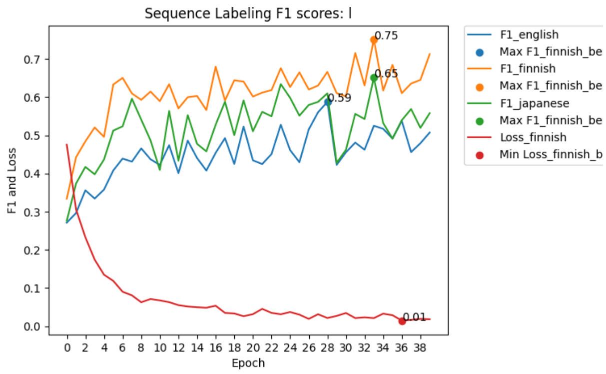

Figure 21: M-BERT: Sequence Labeling Training Finnish

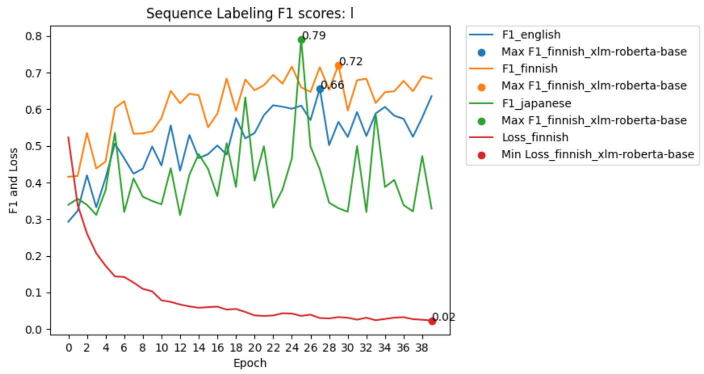

Figure 24: XLM-RoBERTa: Sequence Labeling Training Finnish

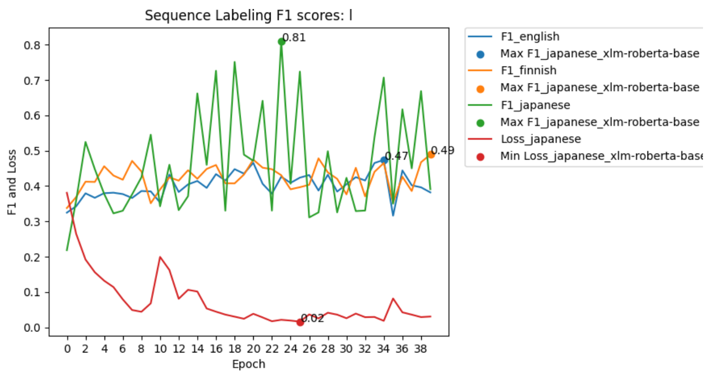

Figure 25: XLM-RoBERTa: Sequence Labeling Training Japanese

L Lab6: Training hyperparameters

Table 10: Training hyperparameters

Dropout Probability	0.25
Batch Size	32
Learning Rate	0.00005
Epochs	40

M Lab5: Overview of parsed-question context pairs

Table 11: Overview of parsed question-context pairs from training (Tr) and validation (Vl) splits post-tokenization

	Ans	Unans	Total	Fails	>512
Eng Tr	2892	3693	6550	804	35
Eng Vl	414	495	905	81	4
Fin Tr	5652	6846	12474	1203	24
Fin Vl	663	843	1502	180	4
Jap Tr	669	4389	4967	3720	91
Jap Vl	65	518	577	453	6

N Lab6: Overview of parsed question-context pairs per model

Table 12: Overview of parsed question-context pairs from training (Tr) and validation (Vl) splits with M-BERT and XML-RoBERTa tokenizers

	Ans	Unans	Total	Fails	>512
M-BERT					
Eng Tr	3551	3670	7221	108	60
Eng Vl	454	491	945	39	6
Fin Tr	6365	6799	13164	430	107
Fin Vl	783	836	1619	52	15
Jap Tr	2953	4250	7203	1200	375
Jap Vl	353	507	860	135	41
XML-R					
Eng Tr	3299	3663	6962	356	71
Eng Vl	454	491	945	39	6
Fin Tr	6256	6813	13069	554	78
Fin Vl	755	841	1596	80	10
Jap Tr	671	4284	4955	3539	284
Jap Vl	79	511	590	419	27