UNIVERSITY OF COPENHAGEN
FACULTY OF SCIENCE

MSc thesis

Danish-language Retrieval Augmented Generation

Investigating the RAG capabilities of LLMs in Danish

Oscar Reves

Advisor: Serge Belongie

Submitted: June 2, 2025 (error(s) in tables 4.4 and 4.7 edited June 4, 2025)

This thesis has been submitted to The Faculty of Science, University of Copenhagen

Abstract

Retrieval augmented generation (RAG) has the capacity to dynamically equip generative large language models with real-world knowledge, without fine-tuning. Since its introduction in Lewis et al. (2020), RAG has seen widespread adoption and has been the subject of extensive research. The RAG capabilities of models in Danish however are mostly undocumented. In this thesis, I investigate the RAG-capabilities of Danoliterate[1] open-source language models. I introduce a novel dataset for question-answering, and an accompanying knowledge base, and combine them with existing datasets to assess the performance of five open-source generative language models. Additionally, I fine-tune an existing embedding model to use for retrieval. I find that RAG increase the performance of models in Danish question answering tasks, but that performance is limited by retrieval accuracy.

[1] The term 'Danoliterate' has been borrowed from (Holm et al., 2025), in the hopes that it will become standardized

Acronyms

CAISA National Centre for Artificial Intelligence in Society.

LLM Large Language Model.

NLP Natural Language Processing.

RAG Retrieval Augmented Generation.

SOTA State-of-the-art.

Contents

Chapter 1

Introduction

LLM-based chatbots have the potential to handle information requests autonomously, with obvious applications for the private and public sector. However, actors in Denmark (Such as the emerging Center for AI in Society) wishing to implement such chatbots face two potential hurdles:

1. The propensity for LLMs to provide incorrect information (Hallucinations)
2. The performance of LLMs in Danish may be inferior to the same models in English, since the training corpus is likely to include far fewer Danish examples

Problem (1) can be partially mitigated by providing LLMs access to external knowledge bases. One such approach is retrieval augmented generation (RAG) (Lewis et al., 2020). Since RAG relies on encoding/decoding models trained on large volumes of internet text, it is possible however that performance on Danish knowledge-based tasks may be deficient in a manner similar to problem (2).

Problem Statement

The purpose of this thesis is to investigate the Danish RAG-based question-answering capabilities of modern pre-trained open-source LLMs. To this end, I will:

1. Provide a background on the basics of language modeling, specifically:
   a) Encoder-only models such as BERT (Devlin et al., 2019)
   b) Causal language models
2. Survey and summarize the literature on RAG as a method for question answering and the metrics used to evaluate it.
3. Survey and summarize challenges relating to low-resource language modeling in general, and Danish in particular, including:
   a) A summary of current Danish NLP benchmarks and metrics
4. Design and implement a model capable of Danish language RAG, based on a pre-trained open-source language model
5. Evaluate the performance of the above model, and compare it to the performance of models in high-resource languages

Chapter 2

Background

This section will cover relevant background topics. The first sub-section is dedicated to neural networks, and will introduce basic terminology and provide relevant definitions. The second sub-section covers the basics of language-modeling, including definitions and technical details of specific models, and is written in a chronological manner. The third sub-section serves as a summary of relevant literature on retrieval augmented generation, while the fourth details methods for evaluating RAG models. The fifth and final sub-section is devoted to characterizing the state of research on Danish-language natural language processing and identifying existing challenges.

Where mathematical definitions are provided, readability has been prioritized over rigor. As such, indexes may occasionally be underspecfied, cardinality of sets may be undefined, and functions may be presented without specifying domains and co-domains. Unless otherwise specified, assume that matrices and vectors are finite and real-valued, and when multiplied have complementary dimensions.

2.1 Neural Networks

This section will briefly cover some rudimentary aspects of neural networks. For a more comprehensive introduction, readers are referred to (Fleuret, 2023).

Neural networks are a class of parameterized differentiable functions that map some arbitrary number of input values to some arbitrary number of output values. The parameterization of neural networks allows them to be represented as acyclic directed graphs whose nodes are referred to as "neurons". A neural network contains a set of input neurons (one for each input value) and a set of output neurons (one for each output value). The remaining neurons are grouped into *hidden layers* based on their distance from the input neurons.

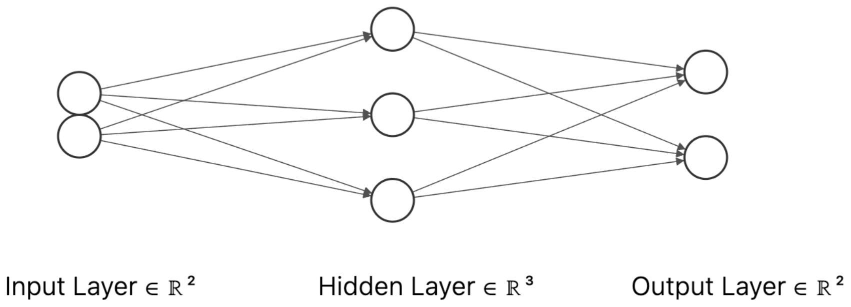

Input Layer ∈ $\mathbb{R}^2$
Hidden Layer ∈ $\mathbb{R}^3$
Output Layer ∈ $\mathbb{R}^2$

Figure 2.1.1: A simple neural network with 2 input neurons, a single hidden layer with three neurons, and 2 output neurons

Values flow from the input neurons to the output neurons along the edges. The edges in a neural network represent *weights*. When a value is passed from one neuron to the next, it is multiplied by the weight of the edge through which it flows. A neuron sums the incoming values and passes them on. Viewing each layer as a vector and collecting all weights between two layers in a matrix whose row and column indices represent the source and destination neuron respectively, the mapping from one layer to the next reduces to a single matrix multiplication. To capture complex relationships however, hidden layers are often equipped with an *activation function*. Neurons in the hidden layers apply the activation function to their value before passing it on. The values of a hidden layer are known as a latent representation.

Formally, for input values $x \in \mathbb{R}^n$, the latent representation of the successive layer $h \in \mathbb{R}^m$ is calculated as $h = f(b + Hx)$ where $b \in \mathbb{R}^m$ and $H \in \mathbb{R}^{m \times n}$ are learnable parameters and f is a non-parameterized function. The vector b is known as the *bias*. The values of successive layers is calculated in a similar manner, such that $h_l = f(b_l + H_l(h_{l-1}))$ and the process of performing this calculation is known as a forward pass. Finding the values of H and b for each hidden layer so that inputs produce a desired output is referred to as *learning*.

Training neural networks is done by equipping them with a *loss function*. A loss function is a measure of the quality of your outputs, with lower values being desirable. So long as a loss function is differentiable, the composite nature of a neural network allows for the gradient of the loss function to be propagated through the network in a process known as *back-propagation*, yielding a gradient with respect to the parameters. This means that a neural network can be optimized with regards to the loss function through gradient descent. Repeating this process for various inputs is known as *training* the network.

A common machine learning task is classification. In a classification, inputs (such as images or text) are mapped to a fixed set of labels. Regardless of the remaining architecture of your network, this can be achieved by assigning each node in the output layer to a label in your set. By applying a softmax function, the output of the network will always sum to one, and can thus be viewed as a predicted probability distribution over your labels. For an arbitrary vector $\mathbf{v} \in \mathbb{R}^n$

$$\text{softmax}(\mathbf{v})_i = \frac{e^{v_i}}{\sum_j^n e^{v_j}} \tag{2.1}$$

In classification settings, the loss chosen is often cross-entropy loss. Cross entropy loss is a measure of the difference between two probability distributions. For two discrete probability distributions $\hat{P}$ and P, represented as vectors such that $P_i = P(x = i)$ the cross entropy is defined as:

$$\mathcal{L}_{\text{cross-entropy}}(P, \hat{P}) = -\sum_j^{|P|} P_j \log \hat{P}_j \tag{2.2}$$

A common approach in classification is to view the correct label for a specific sample as a target probability distribution encoded as a one-hot vector whose only positive element is the index of the correct label. For a given sample input and label (x_i, y_i), where k is the index of label y_i such that $P(y_i)_k = 1$ the cross-entropy loss then simplifies considerably:

$$\mathcal{L}(P(y_i), \hat{P}(y_i)) = -\sum_j^{|P|} P(y_i)_k \log \hat{P}(y_i)_k = -P(y_i)_k \log \hat{P}(y_i)_k \tag{2.3}$$

$$= -\log \hat{P}(w_i)_k \tag{2.4}$$

Thus in classification cross-entropy loss for a single sample is simply the negative logarithm of the probability the model assigned to the correct label.

Another machine learning approach is known as contrastive learning. Contrastive learning views the output of a neural network as a representation of the input, and tries to learn representations such that similar inputs (positive pairs) have similar representations, and dissimilar inputs (negative pairs) have dissimilar representations. If this sounds trivial, consider that the latter similarity may be defined mathematically, while the former may be defined abstractly. A mathematical function that maps the pixel values of pictures of cats differently than the pixel values of pictures of dogs, for example, has no intuitive definition.

Loss functions for contrastive learning vary, but a common example is the InfoNCE loss (van den Oord et al., 2018). As presented in Wang et al. (2024a), for a single positive sample (q_i, p_i) and a set of negative samples $\{(q_i, p_{ij}^-)\}_j^n$, and with parameterized representations $Q_i = f_\theta(q_i)$ and $P_i = f_\theta(p_i)$ and a similarity metric $s(\cdot, \cdot)$, the InfoNCE loss is defined as

$$\mathcal{L}_{InfoNCE}(Q_i, P_i) = -\log \frac{e^{s(Q_i, P_i)}}{e^{s(Q_i, P_i)} + \sum_j e^{s(Q_i, P_{ij}^-)}} \tag{2.5}$$

It is clear that (2.5) is minimized by maximizing $s(Q_i, Pi)$ while minimizing $s(Q_i, P_{ij}^-) \forall j$. Thus minimzing InfoNCE loss produces parameters θ such that the embedding function

$f_\theta(\cdot)$ maps positive pairs in a similar way and negative pairs in dissimilar way. Less clear perhaps is the idea that, by prepending (q_i, p_i) to $\{(q_i, p_{ij}^-)\}_j^n$, and viewing the result as a vector, equation (2.5) is revealed to be the negative logarithm of the softmax of this vector. When viewing the softmax term as a prediction that (q_i, p_i) is a positive pair, the entire RHS of equation (2.5) is revealed to be a cross-entropy loss.

Although classification may seem like a restrictive, label-dependent task, many seemingly unrelated tasks can be formulated as classification tasks. As we will see in the coming chapter, language modeling is no exception.

2.2 Language Modeling

2.2.1 A Brief History (1913-2017)

The idea of computationally generating language predates the inception of modern computers. In perhaps the most famous example, Alan Turing introduced the idea of the imitation game (Turing, 1950), variations of which have become known in common parlance as “Turing tests”. Turing posited that the ability of machines to convincingly imitate a human in written language was a qualified measure of their ability to think.

While Turing did not specify a mechanism for generating language, an early approach has its roots in work by the Russian mathematician Andrey Markov as early as 1913 (Li, 2022). In studying successive occurrences of Cyrillic vowels and consonants in Alexander Pushkin’s novel *Eugene Onegin*, Markov demonstrated by laborious hand computation that the probability of a vowel occurring was dependent on whether or not it was preceded by a vowel or consonant (Markov, 2006), and that this probability could be estimated from samples of the text. Though simplistic, Markov’s idea exemplifies two key insights: namely that natural language is probabilistic in nature, and that the underlying probabilities can be derived from samples of language. Inferring these probabilities is known as statistical language modeling, and models produced in this way will be the subject of this section.

n-gram Models While Markov applied his ideas to vowels and consonants, they can be intuitively applied to words. The probability of any given word occurring after another word can be estimated by the relative frequency of these two words (known as a bi-gram) occurring successively in a sample of text. This insight leads naturally to the idea of generative sampling: Assuming we can sample an initial word, the following word can simply be sampled from this list of relative frequencies. In the language of Markov chains, for each word there exists a *transition probability* to every other word. Assuming a finite vocabulary of size V, the probability of a sequence of words $w_1, w_2, ..., w_n$ can then be modeled as:

$$P(w_1, w_2, ..., w_n) = P(w_1) \prod_{i=2}^{n} P(w_i | w_{i-1}) \tag{2.6}$$

However, this model betrays the underlying assumption that the probability of a word is determined exclusively by the immediately preceding word. Such an assumption is intuitively absurd - sequences sampled this way would be haphazard and semantically empty strings of popular words. Fortunately, the bi-gram model can be extended to an n-gram model, where:

$$P(w_1, w_2, ..., w_n) = P(w_1) \prod_{i=2}^{n} P(w_i | w_{i-n+1:i-1}) \tag{2.7}$$

Unfortunately however, while the bi-gram model can be uniquely described by the uni-gram and bi-gram transition probabilities ($\mathcal{O}(V^2)$, quadratic in vocabulary size and linear in sequence length) an n-gram model requires $\mathcal{O}(V^n)$ transition probabilities (Li, 2022), and is therefore untractable.

This intractability, known as the "curse of dimensionality" (Bengio et al., 2003), motivated the development of neural language models and the associated vectorized representations of words.

Word embeddings Intuitively, some words are more similar to each other than to other words. The words "cat" and "dog" are similar in the sense that they are both animals, but also in the sense that they are both nouns. Thus we can imagine multiple different measures of similarity. If we imagine such a measure as an orthogonal direction in a vector space, the representation of words translates naturally to vectors whose elements are the degrees to which they satisfy this measure. "cactus" being a noun but not an animal, would thus be less similar to "cat" than "dog" is by the two measures proposed, and thus we can imagine that the likelihood of encountering the sentence "I pet the cactus" versus "I pet the cat" may be informed by encountering the sentence "I pet the dog".

Vectorized representations of objects are common in machine learning, and the individual elements of a vector representation are referred to as *features.* Unlike the hypothetical features above, features in neural models are abstract and mostly not interpretable to humans, having been learned through back-propagation to minimize loss on a training task. [1] In (Bengio et al., 2003), this task is maximizing the log-likelihood of the training data, a task which is equivalent to minimizing cross entropy loss. Thus we have a new language model (Li, 2022):

$$P(w_1, w_2, ..., w_n) = f_\theta(\boldsymbol{w_1}, \boldsymbol{w_2}, ..., \boldsymbol{w_n}) \tag{2.8}$$

Wherein f_θ is a neural network with parameters θ and in which $\boldsymbol{w_i}$ is the feature vector for word w_i and where both the feature vectors and the parameters are learned jointly. Thus the task of neural language modeling can be thought of as simply learning parameters $\theta^* \in \mathcal{R}^N$ and embeddings $W^* \in \mathcal{R}^{D \times V}$ where N is total parameter count of the neural network, D is embedding dimensionality, and V is vocabulary size, such that a loss function $\mathcal{L}$ is minimized:

$$\theta^*, W^* \approx \arg\min_{\theta, W} \mathcal{L}(f_\theta(W(w_{1:n})) \tag{2.9}$$

While embedding strategies, the architecture of the neural network, and the loss function may vary, the RHS of equation (2.9) is an accurate descriptor of most neural language models even today.

Word2Vec Building upon the basic conception of a neural language model proposed in (Bengio et al., 2003), researchers at Google pioneered the Word2Vec embedding model.

Two different learning objectives were used to learn Word2Vec embeddings. In the continuous bag-of-words-model, a word is predicted from its surrounding context, both future and past, with no regard to word order:

$$P(w_i) = f_\theta(\{\boldsymbol{w_{i-4:i+4}}\} \setminus \ \{\boldsymbol{w_i}\}) \tag{2.10}$$

[1] The idea of embedding words in abstract feature spaces predates modern neural networks, dating back as far as 1973. See (Rumelhart and Abrahamson, 1973)

Conversely, in the continuous skip-gram model, a single word is used to predict words in the surrounding context:

$$P(\{w_{i-4:i+4}\} \setminus \{w_i\}) = f_\theta(\boldsymbol{w_i}) \tag{2.11}$$

Predicting a word from it's surrounding context as in equation (2.10) would later become known as a *cloze* task, while the inverse task in equation (2.11) would become known as an *inverse* cloze task.

Besides outlining different learning objectives and efficient training methods, Word2Vec was notable because performing arithmetic operations on embedded words produced results that align with human intuition. For example, the nearest neighbor in the embedding space of the sum of the embeddings of "Czech" and "Currency" is "koruna", while a similar method for "Vietnam" + "capital" produces "Hanoi" (Mikolov et al., 2013a). Results like these reinforced the idea that the structure of the embedding space seemed to encode the semantic content of words, and that some semantic content could be translation-invariant. Another similar projection, where "King" - "Man" + "Woman" = "Queen" is presented in the abstract of (Mikolov et al., 2013b), and has served as a canonical example in the field. [2]

Results from Word2Vec and other embedding research such as (Pennington et al., 2014) made it clear that learning word embeddings may have value as an independent task, and that the embeddings may serve as stand-alone feature representations that can be combined with neural networks divorced from the ones with which they were originally jointly trained.

Contextual word embeddings Though word embeddings opened a new frontier in NLP research, they failed to capture an important element of language: context. Words have different meanings, and absent of their context it is impossible to determine which of several meanings a homonym is supposed to represent [3]. Nowhere is this more obvious than in translation tasks. Setting aside syntax and grammar, if words were non-ambiguous, translation could be achieved by simple bijective mapping. Since this is not the case, it is clear that the surrounding words play a role in defining the meaning of a single word. This begs the question: if the semantics of words are context-dependent, and embeddings capture semantics, shouldn't embeddings be context-dependent?

Deep Learning In the years following Word2Vec, the concept of pre-training on a common dataset before fine-tuning for a specific task became widespread in computer vision, where using models pre-trained on the ImageNet dataset was standard practice. This practice was largely driven by the advent of "deep learning", in which neural networks with many layers were trained. Deep learning itself was largely enabled by the pioneering use graphics processing units (GPUs) for training neural networks. The dominant paradigm in computer vision classification tasks at the time was centred around convolutional neural networks (Cun et al., 1990), whose fundamental operations consist largely of highly parallelizable matrix multiplications, for which GPUs are specialized (Krizhevsky et al., 2012).

[2] Interestingly, the analogy task of (Mikolov et al., 2013b), where problems of the form "*a* is to *b* as *c* is to _____" were used as evaluation tools, is conceptually analogous to that of (Rumelhart and Abrahamson, 1973), in which it was used for learning.

[3] Consider how the word "pet", which could refer both to a verb and to a noun, is informed by its context in the sentence "I pet my dog"

Unfortunately, the dominant paradigm at the time in natural language processing was centered around non-parallelizable recurrent architectures (Li, 2022), which maintain a hidden state vector while processing words sequentially, largely insulating them from the benefits of GPU-based training regimens (Vaswani et al., 2017). Besides their non-parallelizable nature, recurrent models suffered from the issue of vanishing gradients. As sequence lengths grew, the contribution of early tokens to the hidden state would diminish. Mitigating this was a central challenge in research.

The advent of the transformer neural network architecture in 2017 would pave the way for overcoming both issues.

Transformers Originally invented as a translation model, the transformer (Vaswani et al., 2017) addresses the issue of sequential processing by simply disregarding it. Instead of processing a sequence of length n sequentially, a transformer processes the entire input at once, by way of something called an attention head. By maintaining two matrices K and Q each of order $n \times d_k$, and a value matrix V of order $n \times d_v$, with d_k and d_v simply being embedding dimensions, an attention head transforms a sequence of n token embeddings into a sequence of n *contextual* embeddings by a simple mostly linear transformation:

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V \tag{2.12}$$

Although (Vaswani et al., 2017) refers to the entire RHS of equation (2.12) as "attention", common practice in the field is to refer the softmax output as the attention, since it produces a square matrix of order $n \times n$, whose elements determine the degree to which each token "attends" to each other token when calculating its contextual embedding. This is clear when realizing that the product of the softmax term with V produces a matrix in which each row is a linear combination of the value vectors.[4] The values of Q, K and V are projections of the input sequence X, produced by the matrix multiplications XW^Q, XW^K and XW^V respectively. The weight matrices W^Q, W^k and W^V are the actual learned parameters of an attention head.

A transformer consists of two parts, an encoder and a decoder. The encoder uses stacked attention heads and linear layers to produce contextual embeddings of the input, after which the decoder uses stacked attention heads and linear layers to map the contextual embeddings of the input to a probability distribution over the vocabulary, from which new tokens are sampled one at a time. After each new token the decoder attends to a concatenation of the input and output tokens to generate a new token.

When coupled with backpropagation, a transformer thus allows sequence-to-sequence mapping through iterative forward passes, in which the embedding of each token influences the embedding of every other token, and learns the degree to which this should happen, while processing the entire input sequence simultaneously, in a highly parallelizable manner, with the slight caveat that compute cost is quadratic in context length.

Though (Vaswani et al., 2017) developed the transformer as a tool for translation, the applicability of the architecture as a universal approach to scalable sequence-to-sequence modeling was readily apparent, and could well be characterized as a paradigm shift in machine learning. As of writing (Vaswani et al., 2017) has over 170,000 citations, and "X is all you need", an homage to the title with "attention" substituted, has become a

[4] For intuition, imagine a scenario in which $d_k = 1$ and in which V is a matrix whose columns are an arbitrary embedding of each token, like Word2Vec.

memetic naming format for published machine learning research, exemplified by (Li et al., 2024), (Picard, 2021) and (Gheini et al., 2021).

2.2.2 Large Language Models (2017-2025)

Following its release, the constituent parts of the transformer (encoder and decoder) would inspire two disparate language modeling paradigms. Models derived the encoder, known as encoder-only models, are best characterized by the model BERT, while models derived from the decoder are known as causal models, best characterized by the GPT series of models. Although the latter predates the former, to best represent their influence on the field, these two paradigms will be presented in reverse chronological order.

BERT Since producing general purpose contextual word embeddings is a sequence-to-sequence task, doing so was a natural application of the transformer architecture. One approach to the problem can be found in (Devlin et al., 2019). Riffing on the sesame-street style naming theme of ELMo (Peters et al., 2018), BERT is a transformer-based contextual word embedding model built around the paradigm of masked language modeling, where training consists chiefly of predicting words in sequences based on their surrounding context, similar to the cloze task (equation (2.10)).

Besides the masked language modeling objective, BERT was pre-trained with a next sentence prediction task, in which it was tasked with determining whether or not two sentences were successive. When combined with task-specific fine-tuning, BERT achieved SOTA in eleven different NLP tasks, including a multi-percentage increase in the GLUE benchmark (Wang et al., 2018).

Variants of pre-trained BERT quickly became the canonical embedding models of the field, and much of published researched now revolved around fine-tuning BERT for task-specific purposes. Fine-tuning could either be done end-to-end, propagating error signals through every layer, or by simply adding a single linear layer.

Tokenization Besides the widespread adoption of BERT, Devlin et al. (2019) also standardized several language modeling concepts, the first of which is sub-word tokenization (as introduced in (Wu et al., 2016)). In language modeling, a string is split into disjoint sub-strings known as tokens. The simplest approach is to tokenize on the word-level, with each word being one token. Recognizing however that semantic meaning is conveyed by word fragments such as prefixes and suffixes, sub-word tokenization schemes split strings into tokens that are sometimes smaller than words to capture these effects. Sub-word tokenization also allows (in theory) for embedding out-of-vocabulary words, provided they are composed of known subwords.

In addition to sub-word tokenization, Devlin et al. (2019) also helped standardize the use of special tokens. Special tokens are tokens inserted into a text before embedding. The masked language objective during pre-training was achieved by randomly replacing a random subset of tokens with the token "`[MASK]`", and then attempting to predict the masked tokens based on contextual word embeddings produced by the surrounding tokens. Coupled with the original token and cross-entropy loss, back-propagation then trains the entire model to maximize the likelihood of correct prediction.

Another noteworthy special token used by BERT is the classification token `[CLS]`. When encoding a sequence, BERT prepends the entire sequence with "`[CLS]`". The embedding of this token is then viewed as representation of the entire sequence, allowing down-stream tasks such as classification to be performed solely by mapping a single token-level embedding onto a set of labels. This radically simplifies downstream tasks by circumventing discussion about pooling strategies, allowing for classification heads to have a fixed input size even for variable length sequences, and simplifies downstream training by only generating a single error signal.

Despite its many benefits, BERT suffered from a single crucial deficit. As a consequence of its masked language modeling objective, BERT is bi-directional (the "B" in BERT), since token-level embeddings are calculated using both future and past tokens. While (Devlin et al., 2019) theorize that this bi-directionality is at the core of it's supremacy as an embedding model, this makes it unsuitable for generative tasks, since sampling new tokens trivially only can be conditioned on past tokens. For this reason, BERT and similar models are referred to as encoder-only.

Causal language models At the time of BERT's release, the chief competing paradigm for neural language modeling was next token prediction, in which a token is predicted based exclusively on the preceding tokens. With a context window of size k:

$$P(t_i) = f_\theta(t_{i-k}, t_{i-k+1}, ..., t_{i-1}) \tag{2.13}$$

Such models can be seen as neural analogues to the non-parametric n-gram models presented in equation (2.6) Since the probability of a token is determined only by previously occurring words, such models are known as *causal* models.

One approach to this task was presented in (Liu et al., 2018), which constructed an artificial summarization task by viewing Wikipedia articles as ground truth summaries of their references. The approach then concatenated input and output sequences, and then used a single transformer decoder stack to predict tokens, with each token attending only to past tokens. In practice this was achieved by *attention masking*, a process by which attention scores for future tokens are manually set to 0. This process is trivial, since for the self-attention matrix of an arbitrary token sequence, the future attention weights simply correspond to the upper triangular.

Realizing its potential for unsupervised natural language modeling, the transformer-decoder was embraced by OpenAI in (Radford and Narasimhan, 2018), in which they refer to the task as *generative pre-training.* The approach birthed a series of models referred to as generative pre-trained transformers.[5] GPT-1 was pre-trained with a language-modeling objective on the Book-corpus, a corpus containing several thousand published books, after which it was fine-tuned with a set of auxiliary supervised NLP tasks. Supervised fine-tuning was performed in a multi-task learning set-up, in which error signals were derived simultaneously from the unsupervised language modeling and the supervised auxiliary tasks.

Conceptual echoes At this point a small retrospective is informative. (Bengio et al., 2003) defined the task of neural language modeling as follows; to predict the probability of a single word, under the assumption that language is causal and local, and thus the probability of a word w_i is determined only by a limited number of its predecessors:

1. Embed the $n-1$ preceding words as $x = (C(w_{i-n}), C(w_{i-n+1}), ...C(w_{i-1}))$[6]

[5] Or GPT for short, an acronym with which the reader is likely already familiar

[6] The notation is somewhat under-specified here. Embedding is performed by simply indexing into an embedding matrix

2. Calculate $y = b + Wx + U\tanh(d + Hx)$ where H, W and U are learnable weight matrices and b and d are learnable bias vectors

3. Calculate $\hat{P}(w_i) = \text{softmax(y)}$

By choosing the appropriate dimensions for H and W, $|y| = |V|$, and thus $\hat{P}(w_i)$ can be viewed as a probability distribution over the vocabulary, with each element consisting of a predicted probability for a specific word. The correct prediction can then simply be viewed as a one-hot vector $P(w_i)$ of length $|V|$ whose positive element is at index k corresponding to the right word. Thus the task of language modeling is reduced to a classification task, explaining the reappearance of the cross-entropy loss from section 2.1.

Conversely, the approach to language modeling applied in (Radford and Narasimhan, 2018) by predicting a token u_i can be summarized as follows:

1. Embed the $n-1$ preceding tokens as $h_0 = (W_e(u_{i-n}), W_e(u_{i-n+1}), ..., W_e(u_{i-1})) + W_p$

2. For l in $[1, n]$ calculate $h_l = \texttt{transformer\_block}(h_{l-1})$

3. Calculate $\hat{P}(u) = \text{softmax}(h_n W_e^T)$ [7]

Where a transformer block is a stack of attention heads and linear layers. The summaries should illustrate that, despite 15 years of difference, the approach in (Radford and Narasimhan, 2018) is a direct conceptual successor to the neural language model of (Bengio et al., 2003), whose only true differences are the architecture(s) of the neural networks, and the fact that the former uses sub-word tokenization and fine-tunes the parameters with downstream tasks. When considering the fact that a transformer includes skip connections, and that Wx is a skip connection, the differences in architecture reduce to amount of layers, choice of activation function (ReLU vs tanh), and the inclusion of attention heads. In a coincidence somewhat serendipitous to this thesis, (Bengio et al., 2003) cites Danish research (Jensen and Riis, 2000) as the inspiration for using neural networks to construct vectorized representations of symbols.

Although the embeddings produced by GPT-1 were later eclipsed by BERT for downstream performance, generating text with GPT style models is trivial, since the output for a sequence is a probability distribution over the vocabulary, from which the next token can then be sampled. Since this sampling can be applied iteratively, where the input to the next step is the new sequence of tokens obtained by appending the sampled token to the input sequence, GPT style models are known as autoregressive decoders.

Large language models Though they differ in approaches, causal and encoder-only models are both transformer based language models whose pre-training consists chiefly of unsupervised tasks. Coupled with the existence of the internet as a large repository of unlabeled data, this meant that language models were not bottle-necked by limited amounts of expensive annotated data. Leveraging this fact, the 1.5 billion parameter GPT-2 (Radford et al., 2019) was trained on a bespoke dataset of over 8 million documents, compiled by scraping the content of outbound links from the social media platform Reddit. The result was a model capable of zero-shotting SOTA on 7 language modeling tasks.

[7] Once again, regardless of presentation embedding and embedding simply consists of indexing into a fixed embedding matrix, with the exception of $\mathrm{W_p}$, which is a positional encoding matrix necessary in transformers

The trend of scaling model and pre-training corpus size continued. An empirical evaluation performed in (Kaplan et al., 2020) found that language modeling performance on the *test* set scaled with each of model size, corpus size and compute, according to a predictable power law distribution, with no observed upper bound. Besides introducing GPT-3, (Brown et al., 2020) further explored this relationship, concluding that performance continued to increase up to and including models with 175 billion parameters. In addition to zero and few-shot performance on a number of NLP tasks, GPT-3 was capable of generating news articles that humans had difficulty distinguishing from human-written articles. For an informative example, see figure 3.13 in (Brown et al., 2020), in which the ability of humans to distinguish between real and synthetic news articles converges to random chance as parameter count increases.

As a result, though (Radford and Narasimhan, 2018) presents generative pre-training as a means to an end for contextual embeddings for downstream language tasks that are largely non-generative, it became clear that language generation was a reachable target. Four short years later, by coupling their frontier model with a public api reachable in a browser, OpenAI released Chat-GPT onto a largely unsuspecting public. Seven decades after its inception, in the eyes of many who could now chat with a language model in real time, Alan Turing's theoretical concept of an imitation game ceased to be a hypothetical.

The modern era By publicly demonstrating the power and scalability of generative pre-training on unlabeled data, the release of ChatGPT sparked an arms race of sorts, and the years since 2022 have seen a rapid proliferation of large language models. The price of stock in the technology company Nvidia, which produces a large share of the world's GPUs, has more than quadrupled since January 2022(Nasdaq, Inc., 2025), and OpenAI now faces competition from a wealth of other companies offering publicly accessible large language models, including Meta, Google, Anthropic and xAI. In January 2025, newly elected US President Donald Trump announced project Stargate, a joint venture with private actors to build AI-focused compute infrastructure at a projected cost of $500 billion (Reuters Staff, 2025). For comparison, (Stine, 2008) estimates the combined inflation-adjusted price of the Manhattan and Apollo projects at roughly $180 Billion.[8]

Due to the quantities of compute required to train modern generative LLMs, associated costs are often prohibitive to hobbyists and even small-scale institutional actors. Nevertheless, many open-source models have been trained and are readily available on the Hugging Face hub, an online platform for hosting models and datasets that has become the de facto figurative town square for the open-source language modeling community.

Open source models hosted on Hugging Face include models by large corporations, such as the Gemma family of models by Google (Team et al., 2024a) or the Llama "herd" from Meta (Grattafiori et al., 2024), but also models developed by smaller organizations (Jiang et al., 2023) and even hobbyists, among whom fine-tuning models is popular. Besides models and datasets, the Huggingface hub also hosts a number leader-boards, where models compete against each other publicly on various benchmarks. Thus the Hub fosters both collaboration and competition in tandem, and it's growth is a testament to resources and interest dedicated to language modeling. An analysis performed in 2023 found that the Hub contained over 300 thousand models (Osborne et al., 2024).

Training strategies for modern LLMs are diverse. Several are worthy of mention. The first is knowledge distillation (Hinton et al., 2015). Knowledge distillation is a form of

[8] The figure given in the report (∼$120 Billion 2008), was inflation-adjusted to 2025 using the consumer price index inflation calculator hosted by the US Bureau of Labor Statistics

teacher-training wherein one model learns from the output of another model. In the context of LLMs, this means that the student model receives as input the softmax output of the teacher model. Consider for a second why this makes sense. In the training described earlier in this subsection, the cross-entropy loss when predicting a token was calculated with respect to the correct token, and only the correct token. This means indirectly that "close" predictions are penalized to the same extent as incorrect predictions (see table 2.1). This also means that only one error signal is produced per token. Conversely in knowledge distillation, for each token in the corpus an error signal for each token in the vocabulary is produced. Knowledge distillation allows a small model to learn to effectively mimic a larger model, and is key to proliferation of capable language models with parameters in the single digit billions.

Context tokens	**Predicted token**	**Loss**
I like to smoke	cigarettes	$-\log(\hat{P}[w_i = \text{“cigarettes”}])$
I like to smoke	cigars	$-\log(\hat{P}[w_i = \text{“cigarettes”}])$
I like to smoke	aubergine	$-\log(\hat{P}[w_i = \text{“cigarettes”}])$

Table 2.1: Cross entropy-loss for causal language modeling with $w_i =$ "cigarettes". The loss is invariant to the probabilities assigned to tokens that are semantically close to the ground truth token

Limitations of LLMs Besides generating language, studies have shown that language models can memorize and reproduce factual knowledge (Petroni et al., 2019). Knowledge stored in this manner is referred to as parametric knowledge, since it is contained in the parameters of the model. Parametric knowledge has a number of shortcomings, however. Namely, it is:

1. Non-verifiable. Since language models often operate as black-boxes, the validity of knowledge is not immediately verifiable.
2. Static. Models are often trained on text corpora generated before a specific date, and thus have no knowledge of facts beyond this date.
3. Non-private. Implementing actors may want models to have access to specific knowledge only in certain contexts due to reasons of confidentiality.

While these limitations can all be partially mitigated by re-training models to incorporate new knowledge in a process known as fine-tuning, this process can be computationally expensive. While the generative BART-large model used in (Lewis et al., 2020) contains 400M trainable parameters, modern LLMs such as LLaMA (Touvron et al., 2023) number in the billions, making fine-tuning potentially prohibitively expensive.

The above issues thus motivate the need for language models that can incorporate external knowledge bases without further training (so called non-parametric memory). One such approach is retrieval augmented generation.

2.3 Retrieval Augmented Generation

Retrieval augmented generation (RAG), originally formulated in (Lewis et al., 2020), is a framework for extending the capabilities of generative language models by giving them access to external knowledge bases. A RAG model consists of three parts, namely:

1. A generator (a generative LLM)

2. A retriever (retrieves relevant parts of the knowledge base)

3. A knowledge base (a text corpus)

Unlike traditional seq2seq models, when responding to a text prompt (commonly called a *query* in RAG literature) RAG models first retrieve relevant passages from the knowledge base, after which the generator uses both the query and the passages from the knowledge base as context for generating a response. Since the retriever and knowledge base are separate from the generator, they can be substituted or amended without retraining the generator.

The individual components of a RAG model can vary, but common approaches consist of an auto-regressive LLM generator and a neural model for retrieval. In (Lewis et al., 2020) for example, the auto-regressive BART (Lewis et al., 2019) was tested as a generator both with a neural DPR retriever (Karpukhin et al., 2020), and with a term-based retrieval method, using a clean plain-text version of a Wikipedia dump as the knowledge base.

The following subsection will cover various retrieval methods.

2.3.1 Retrieval

Retrieval, also known as information retrieval, is a method of extracting relevant parts of a text corpus given a text query, and is a subject of its own, with its own dedicated conference since 1992 (Harman, 1993). (Karpukhin et al., 2020) defines a retriever in the following way: Given a corpus $\mathcal{C}$ and a query q, a retriever $R : (q, \mathcal{C}) \rightarrow \mathcal{C}_{\mathcal{F}}$ returns a subset of documents $\mathcal{C}_{\mathcal{F}} \subset \mathcal{C}$. Given a ranking function, $S : (q, \mathcal{C}) \rightarrow s$ where s is some notion of relevance, retrieval can then be phrased as returning the top-k documents with the highest relevance score.

Early retrieval methods focused on term-matching the query with documents. Representing both documents and the query as a bag-of-words, a naive implementation could thus simply return the k documents containing the most words also found in the query. Such an approach however fails to account for the obvious limitation that words of low specificity are likely to co-occur in many documents and queries. Consider how unlikely the word "the" is to indicate relevance between two texts. Methods to correct for this, such as by weighting documents with the co-occurrence of rarer words higher, such as the ubiquitous TF-IDF, date back as far as the 1970's (Sparck Jones, 1972), and culminate with the introduction of the BM25 ranking function in 1994 (Robertson et al., 1994).

The advent of neural language modeling allowed for ranking based on learned embeddings. In DPR (Karpukhin et al., 2020), corresponding documents and queries are encoded separately via BERT, and a contrastive learning objective is used to train both encoders to encourage clustering of questions and documents. After training, each document can then be encoded with the learned document encoder, with the resulting embeddings serving as an index for the corpus. In the context of retrieval, these embeddings are known as keys. Viewing the dot product of an encoded query [CLS] token $E_Q(q)$ with an encoded document [CLS] token $E_D(d)$, retrieval then reduces to:

$$R(q, \mathcal{C}) = \underset{d \in C}{\arg\max}\, E_Q(q)^T E_D(d)$$

ColBERT (Khattab and Zaharia, 2020) uses a similar ranking method to Karpukhin et al. (2020), but instead employs a ranking function based on the maximum token-level cosine similarity. Where E_q and E_d represent normalized BERT embeddings of the of the query and document respectively, indexed at the token level. Unlike DPR, ColBERT captures token-level similarities between the documents and queries, and similar strategies that employ token-level similarities have become known as late interaction models.

While DPR and ColBERT both rely on supervised contrastive learning, unsupervised approaches also exist. Many of these methods rely on contrastive learning with synthetically generated query/key pairs, such as (Lee et al., 2019), in which the complementary spans of the inverse cloze task[9] serve as keys and queries, or (Izacard et al., 2021), in which keys and queries are randomly sampled substrings of a document.

Other methods include (Izacard and Grave, 2020), in which a retriever is trained as part of a full RAG stack, using the attention scores of the generator model as a training signal.

More innovation with regards to retrieval includes instruct-tuned embedding, in which passages and queries are accompanied by a natural language description of the task for which they are embedded before encoding, thereby making the embeddings task-specific (Su et al., 2023), and LLM2Vec, in which generative LLMs are re-trained as encoders by fine-tuning with bi-directional attention and unsupervised contrastive learning (BehnamGhader et al., 2024). The latter approach has birthed embedding models with parameter counts in the billions.

2.3.2 Generation

Generation in a RAG set-up is achieved by combining the query with the results from retrieval and feeding them to a suitable generative language model. Though (Lewis et al., 2020) and (Izacard and Grave, 2021) both dedicate sections to different strategies for conditioning on retrieved contexts during generation, modern LLMs are orders of magnitude more capable than in 2020, with context windows long enough to allow for the the all of the retrieved context to be passed to the model along with the query.

While prompting strategies vary, this aspect of RAG is essentially straight-forward.

2.3.3 Knowledge Bases

The final component of a RAG model is the set of documents on which retrieval is performed, known as the knowledge base. The prototypical knowledge base, as in (Lewis et al., 2020), is Wikipedia.

To prepare a knowledge base for retrieval, it is generally divided into smaller sections in a process known as chunking, after which the chunks are embedded via an encoder model. The resulting keys are then stored in a document or database known as an index, upon which similarity search is performed. The entire process is known as vector-indexing a corpus.

Strategies for chunking vary. While Lewis et al. (2020) naively splits Wikipedia into disjoint 100-word chunks, more sophisticated methods exist. Instead of splitting a text based on white-space, a tokenizer can be used. In this way, a corpus can be prepared in such a way that it matches the context-window of a specific embedding model with minimal padding. Additionally, chunking can be semantically aware, by trying to preserve sentences or paragraphs in single chunks. Finally, metadata can be included, such as by prepending the title of the source article to a chunk before encoding. All three methods are applied in (Lee et al., 2019).

[9] See (2.11)

Though matching a key to a query can be done by brute force similarity search, the size of knowledge bases (often containing millions of keys) has motivated alternative strategies that can minimize both search latency and storage costs. These strategies include non-exhaustive searches, or searches with compressed encodings. Since they do not guarantee optimal retrieval results, they are collectively referred to as approximate nearest neighbor searches. A collection of these methods is presented in the FAISS library (Douze et al., 2025).

2.4 Evaluating RAG

2.4.1 Evaluation With Question Answering

The purpose of RAG is to equip generative models with knowledge. For this reason, question-answering is uniquely positioned as a task for evaluating the capabilities of RAG models. Question answering (QA) tasks, as the name implies, are tasks within which a model is tasked with providing answers to a set of questions. Broadly speaking, question answering tasks can be categorized as open domain, extractive or multiple choice.

In open domain QA, models simply provide the answers to questions. Open Domain QA datasets therefore usually consist of simply a set of questions and a set of answers. Any dataset that includes questions and answers can be treated as open domain, whether or not it was intended to be approached in this manner. An example is Natural Questions (Kwiatkowski et al., 2019), which consists of 300,000+ questions paired with a short and long answer, along with the name of the Wikipedia article from which the answer is derived.

In extractive question answering, models are given a question, a passage, and then tasked with identifying the part of the passage that answers the question. Since the answer is often a sub-string of the passage, answers may be provided as simply character or word indices. Since the information required to answer the question is provided alongside the question, extractive QA is often referred to as reading comprehension. Extractive QA datasets include SQuAD (Rajpurkar et al., 2016) and TriviaQA (Joshi et al., 2017).

In multiple choice settings, models are given a question and a set of answers, and tasked with identifying the right answer. For modern generative models, this can be done by simply assigning an index to each candidate answer and prompting the model to output the index of its choice. Alternately the negative log-likelihood of each candidate answer can be calculated by language modeling, and the candidate with the lowest score is viewed as the chosen answer. The latter has the benefit of working for encoder-only models, despite their lack of generative abilities.

Evaluation in the multiple choice setting is trivial. For extractive question answering, evaluation has traditionally consisted of various measures of lexical overlap between the generated answer and the ground truth answer(s), the strictest of which is the exact match (EM) metric[10], in which any deviation from the ground truth is viewed as an incorrect answer. A more lenient metric consists of viewing both the generated answer and the ground truth as bags-of-words, and then calculating the F1-score.

Despite equivalence to the exsting task of similarity scoring, for which a large body of literature exists, evaluation in the open-domain setting is more complicated, since lexical overlap is extremely sensitive to paraphrasing (Two semantically equivalent sentences

[10] EM is sufficiently strict that human performance on SQuAD, measured by way of inter-annotator agreement, is only 0.77 (Rajpurkar et al., 2016)

could have zero lexical overlap). Nonetheless, standard approaches for many years consisted of increasingly convoluted measures of lexical overlap such as BLEU (Papineni et al., 2002), ROUGE (Lin, 2004) and Q-BLEU (Nema and Khapra, 2018).

Like many other areas of the field, the proliferation of contextual embeddings would revolutionize similarity scoring, which could now be approached via well-defined notions of similarity for vectors, such as cosine-similarity. Approaches include BERTScore (Zhang et al., 2020), BLEURT Sellam et al. (2020) and Sentence-BERT Reimers and Gurevych (2019), all three of which are BERT-based models.

Rapidly advancing generative capabilities would likewise open a new frontier for evaluation of question-answering. In a paradigm that would become known as LLM-as-a-judge, the task of comparing an answer to a reference answer was simply outsourced via prompting to a language model (Zheng et al., 2023). Such approaches are sometimes also referred to as “soft evaluation”, since they are only assumed to correlate with correctness, and are potentially subject to their own set of errors.

2.4.2 Evaluation of Sub-Components

Since the RAG framework is inherently modular, evaluation of the individual components is also possible, and can be informative with regards to diagnosing downstream performance.

For retrieval, evaluation usually consists of metric@k-type evaluations, in which a given metric (precision, recall or accuracy) is measured for the top-k retrieved results, averaged across a whole dataset. Since retrieval is a subject of research of its own, several datasets exist to specifically benchmark retrieval. (Thakur et al., 2021) collected several different retrieval benchmarks from multiple domains into a single multi-task retrieval benchmark known simply as BEIR.

(Es et al., 2024) presents an LLM-as-a-judge framework that evaluates RAG models without any associated ground truth answers, by asking an LLM to evaluate along several different dimensions, such the relevance of the retrieved context to the original query, and the faithfulness of the answer to the context.

Multilingual NLP The development of any scientific field is resource-dependent, and NLP is no exception. In NLP, besides the practitioners and funding, a central resource is datasets, since datasets provide not only the basis on which models are trained, but also the shared framework for evaluation. Thus, in NLP different languages can be taxonomized by the availability of corresponding datasets. One such taxonomy is provided in (Joshi et al., 2020), which divides languages into 6 categories based on the volume of available data, both labeled and unlabeled.

If available data drives development of models, and models can provide tangible benefits (such as the automatization of certain public services), countries and cultures speaking lower resource languages risk being left behind. Thus research in multilingual NLP is motivated not only academically, but societally.

Such research focuses not only on the development of datasets but also models. Both datasets and models can be monolingual or multilingual. Though dated, an overview of both can be found in (Doddapaneni et al., 2025), which primarily presents variations of BERT.

Though none of the theory described thus far has been language specific, English is the lingua franca of not only academia, but also the internet at large, and therefore much of NLP research is focused exclusively or primarily on the English language. Nonetheless, research exists and thrives in other languages, including Danish. This research has produced a number of different datasets and benchmarks.

2.5 Danish-language NLP

There exists a diverse body of NLP research focusing on the Danish language. This research has produced a number of datasets and models. As part of a national strategy for artificial intelligence, the Danish Ministry of Digitalization maintains the website sprogteknologi.dk, which hosts an overview of news and resources related to Danish NLP.

Much of research focuses on applying existing methods from NLP to Danish contexts. Examples include DaNE, a Danish named entity recognition dataset Hvingelby et al. (2020), and DanFEVER (Nørregaard and Derczynski, 2021), a Danish claim verification dataset inspired by the English FEVER.

Due to their shared origins and linguistic similarities, resources are often grouped across the Scandinavian languages. Examples include the ScandEval benchmark (Nielsen, 2023), and the more recent Scandinavian Embedding Benchmark (Enevoldsen et al., 2024), in which models are assessed on a range of embedding tasks. In his master's thesis Søren Vejlgaard Holm from the Technical University of Denmark introduced the Danoliteracy benchmark, which combined novel and existing datasets into a single benchmark, intended to measure the Danish language proficiencies of generative LLMs (Holm et al., 2025), (Holm, 2024).

Besides datasets, the Danish NLP community has produced many variants of English models, such as MeDA-BERT (Pedersen et al., 2023), a Danish BERT model trained on a large corpus of Danish medical data. Researchers from Aalborg University, the IT University of Copenhagen and the Pioneer Center for Artificial Intelligence collaborated to to train the generative SnakModel, the lessons from which they summarized in (Zhang et al., 2025). SnakModel was trained by continuous pre-training of Llama2-7B$_{\text{base}}$ on a Danish corpus of 13.6 billion words.

Despite this, the literature survey performed for this thesis was unable to find any published research detailing the RAG-capabilities of Danoliterate models. This may be due in part to the lack of appropriate benchmarks against which to test such models. Large-scale standardized question answering datasets are similarly lacking in the body of research, and no Danish equivalent to canonical QA datasets such as SQuAD, Natural Questions, or HotpotQA exists.

Where QA datasets do exists, they are often the product of human translation from an English dataset, such as the danish subset of ScandiQA, a Scandinavian QA dataset derived from MKQA (Longpre et al., 2021), itself derived from Natural Questions. MKQA consists of a subset of Natural Questions, translated into 26 languages. ScandiQA paired a subset of these translations with machine translation of accompanying English passages from the original Natural Questions, thus no part of ScandiQA consists of naturally occurring Danish text.

Exceptions to translated datasets include the Citizenship test introduced in the Danoliterate benchmark, which consists of questions from tests administered to applicants for Danish citizenship or permanent residency, but it only includes 605 non-unique questions. A notable exception can be found in WebFAQ (Dinzinger et al., 2025), a massive multilingual QA dataset consisting of FAQ-style questions scraped from Common Crawl, the Danish subset of which contains 769 thousand question/answer pairs, coupled with the source URL from which they were extracted.

Chapter 3

Methodology

The purpose of this thesis is to investigate the question-answering capabilities of open-source language models in Danish. This section details the experiments performed to investigate these capabilities, and the associated design choices, including the chosen datasets and models, the inference modes, and the knowledge bases.

3.1 Datasets & Models

3.1.1 Datasets

DR News Quiz The DR News Quiz is a dataset containing 105 multiple-choice questions in Danish. It was compiled for the purpose of this thesis. Questions were scraped from the weekly news quiz hosted online by DR (*Danmarks Radio*), a taxpayer-funded Danish public broadcaster.

Question:	I sin første nytårstale fokuserede kong Frederik især på én specifik del af befolkningen. Hvem er det, han taler om her? *In his first new years soeach king Frederik focused on one part of the population in particular. Who is he referring to here?*
Options:	Sygeplejersker (*Nurses*) De unge (*The youth*) Tidligere minkavlere (*Former mink farmers*)

Table 3.1: A sample question from the DR News Quiz

The dataset was designed for testing the Danish question answering capabilities of RAG-models, and thus is intended to be accompanied by the DR News Corpus as a knowledge base. It contains 124 multiple choice questions, corresponding the questions from the first 15 weeks of 2025. Each quiz consists of 6 or 7 questions related to news that week. The questions are in Danish and often, but not always, relate to Danish domestic news. Each question has three possible answers. Unfortunately, many questions are accompanied by a small video, played in an embedded media player, and therefore do not make sense as standalone questions.

DR News Corpus The DR News Corpus is a collection of 4105 news articles published online by DR, and compiled for the purpose of this thesis. It consists of articles posted in the first 15 weeks of 2025, compiled by web-scraping google-indexed news articles posted online by DR.

Each sample in the dataset contains the title and the body of a single article in plain-text, along with the URL from which the article was scraped. Due to the imperfect nature by which the articles were scraped, the corpus is not an exhaustive body of articles posted by DR in this period. The samples contain no meta-data besides the URL, since the corpus is intended to be searched semantically.

Citizenship Test The Citizenship test (Holm et al., 2025) is a Danish multiple choice question answering dataset. The dataset consists of questions from tests issued by the Danish government to prospective citizens and permanent residents, and was compiled by Søren Vejlgaard Holm as part of his MSc thesis.

The dataset comprises 605 Danish questions, focusing on issues of Danish history, civics, culture and geography. The dataset is available on Huggingface. (Søren Vejlgaard Holm, 2024)

Title:	Øresundsbroen slår rekord
	The Øresund bridge sets a record
Body:	Øresundsbroen slår rekord. Aldrig har så mange af os kørt over Øresundsbroen som i år. Over 7,5 millioner ture over broen er det blevet til, og det er det højeste nogensinde. Øresundsbroen vurderer (...)
	The Øresund bridge sets a record. Never before have so many of us driven across the Øresund bridge as the present year. Over 7.5 million trips over the bridge have take place, and this is the highest ever. The Øresund bridge assesses (...)

Table 3.2: A sample snippet from an article in the DR News Quiz

Synthetic Retrieval Data A collection of 100k synthetic query/document pairs generated using Gemma-2-27b-it by Kapser Groes Albin Ludvigsen, using a method inspired by (Wang et al., 2024b). Available on huggingface.

Danish SQuAD SQuAD (Rajpurkar et al., 2016) is an extractive question-answering dataset. The dataset consists of passages from Wikipedia articles, and accompanying questions, along with accompanying answers. Each answer is a verbatim span from the passage, and questions often contain multiple possible answers. The dataset contains over 100,000 questions, and was compiled by mechanical Turks. The dataset is publicly available on Huggingface. A Danish machine translated version of the SQuAD and SQuAD 2.0 datasets is available online

Documentation for the dataset is sparse, but the online location suggests that it was translated using the Translate Align Retrieve method presented in (Carrino et al., 2019). Though this dataset was a primary target of an early version of this thesis, it was abandoned due to poor quality of the translations. A description has been left here for completeness.

3.1.2 Models

Gemma-2-9b-it Gemma-2-9b-it is a 9 billion parameter generative languge model released by Google (Team et al., 2024b). It was pre-trained on a corpus of 8 trillion tokens, with soft targets for knowledge distillation provided by a larger unspecified model before being fine-tuned with supervised fine-tuning, including instruct-tuning.

Snakmodel-7b-instruct Snakmodel-7b-instruct is a 7 billion parameter generative language model trained in collaboration by researchers from Aalborg University, the IT University of Copenhagen and the Pioneer Center for Artificial Intelligence (Zhang et al., 2025). It is derived by continuous pre-training from Llama2-7b on a Danish corpora of 13.6 billion words before undergoing instruct-tuning on machine-translated instruction-following datasets.

Nous-Hermes-2-Mistral-7B-DPO Nous-Hermes-2-Mistral-7B-DPO is a 7 billion parameter generative language model. Characteristic of the intricate ecosystem on Huggingface, its lineage is complex. Released by Nous Research (Huggingface, 2024), the model is derived from Mistral-7B by first instruct-tuning, then training with direct preference optimization (Rafailov et al., 2023), a supervised alternative to the standard protocol of reinforcement learning with human feedback Ouyang et al. (2022).

Yi-34B Yi-34B is a 34 billion parameter generative language model by the Chinese company 01.Ai, on a dataset of roughly 3 trillion English and Chinese tokens. From the technical specification provided in (AI et al., 2025), it is unclear where the multilingual non-English non-Chinese capabilities come from, or whether the model is intended to be used outside of the context of these two languages.

Suzume-llama-3-8b-multilingual Suzume-llama-3-8b-multilingual is an 8 billion parameter generative language model derived by fine-tuning on a dataset of multilingual conversations. (Devine, 2024)

E5 The E5 family is a series of embedding models trained contrastively with weak supervision (Wang et al., 2024a). A multilingual suite was introduced in (Wang et al., 2024c). Unless otherwise specified, references to E5 in this report refer to mE5$_{\text{large-instruct}}$, which is an xlm-roberta-base model trained on automatically generated multilingual query/passage pairs from subsets of Common Crawl, before undergoing supervised fine-tuning with a mixture of real and synthetic datasets.

3.2 Data Collection

DR News Quiz Since the online quiz consists of dynamically loaded Javascript content, it is not an appropriate target for traditional web-scraping. Therefore, the quiz was scraped using a custom built web-scraper that uses the Selenium software package for Python[1] to simulate user interactions (button-clicking) in a headless browser, iteratively brute-forcing each question and saving both the question, the answer options, and the index of the correct answer. The scraper was written for this thesis and released publicly[2].

Since Javascript content can be slow to load, and since fresh URL GET-requests are only made when a question in answered incorrectly, the request frequency is relatively low, and complies with good etiquette on scraping.

DR News Corpus Though every news article published by DR is freely available online, the main interface(s) of DRs news site consist of dynamically updated section-specific front pages meant to serve as a daily feed of relevant articles, and DR do not expose an index over their past articles via their sitemap. Thus a lift of URLs was compiled, after which each URL was scraped using the Beautiful Soup python package (Richardson, 2023).

Danish Wikipedia Dump The Danish Wikipedia dump from March 1, 2025, containing every single Danish article on Wikipedia, was downloaded from the official repository maintained by the Wikimedia foundation. The Wikipedia dump was processed from xml to plain text using the WikiExtractor tool (Attardi, 2015).

[1] https://selenium-python.readthedocs.io/

[2] available at: https://github.com/OscarReves/dr-news-scraper

3.3 RAG Pipeline

Since experiments largely consisted of question-answering with retrieval, the bulk of the experimental work consisted of building a software base that was sufficiently modular to allow for experiments to be run with different generative models, different retrieval strategies, different knowledge bases, and with different question-answering approaches. To this end, a general-purpose RAG pipeline was developed in python.

Both the DR News Quiz and Citizenship Test were targeted with a variety of RAG-based approaches, varying the approaches used for retrieval, generation and evaluation. Both datasets were approached both as multiple choice datasets, and as open-domain.

3.3.1 Knowledge Base Construction

Two knowledge bases were constructed for the purposes of this assignment, one from the articles in the DR News Corpus, and one from the articles in the Danish Wikipedia dump.

In both cases the articles were chunked using the langchain RecursiveCharacterTextSplitter, set to split by new-line characters with a chunk size of 256 and chunk overlap of 128. In order to improve their value as standalone-texts, each chunk was prepended with the title of its source article.

To match the expected format of the E5 embedding model primarily used for retrieval, each chunk was prepended with the string "passage: " before encoding in batches of 2048, after which an index was constructed using the FAISS package. Context embeddings were L2-normalized before storing in the index, reducing cosine-similarity search to dot-product similarity search.

3.3.2 Prompting

Prompts were kept mostly consistent across models. The system prompts for multiple choice and open-domain settings are almost identical, and can be seen in figures A.1.2 and A.1.1. The prompting for open-domain and multiple choice question-answering can be seen in figures A.1.4 and A.1.5 respectively. For compatible models, prompts were formatted using Huggingface's `apply_chat_template()`, which separates system and user prompts with special tokens. Every prompt was passed with `add_generation_prompt` set to True, and all responses were generated deterministically without beam-search and with temperature scaling set to 0 for reproducibility, and to reduce compute costs.

Unfortunately, the instruction "Context may not be relevant" was accidentally included in prompts given to Gemma-2-9b-it in the open domain setting of all tasks, and must be considered as a potential source of error. Very preliminary testing indicates that responses are mostly identical whether or not this sentence is included, but re-runnning the whole test suite is not possible.

The exact prompts used can be found in the appendix.

3.3.3 Evaluation

In multiple choice settings, accuracy was calculated by extracting the first character of the output and comparing it to the answer label. Evaluation for open-domain question-answering was done with LLM-as-a-judge, using Gemma-2-9b-it as the judge. The prompt for evaluation can be found in figure A.1.6. The phrasing of the prompt was partially inspired by suggestions from ChatGPT (Running a GPT-4o backend), and similar to the system prompt "You are a helpful assistant", prompts of the kind "You are an expert ..."

are generic and ubiquitous for LLM-as-a-judge tasks and can thus similar prompts can be found in published research such as (Liu et al., 2025) and (Shang et al., 2025).

Inter-annotator agreement was calculated as the average pairwise agreement across all annotators. Evaluations produced by each model were compared against each other, and to human evaluation.

3.3.4 Compute

Inference was performed primarily on the P1 Cluster hosted at the Technical University of Denmark, on a node equipped with two 80GB NVIDIA H100s, access to which was granted by the Pioneer Centre for Artificial Intelligence. Every model used was downloaded from Huggingface and run completely locally.

Though not designed for public consumption, the full codebase can be found on GitHub (Reves, 2025).

3.3.5 Retrieval

Retrieval was performed on demand during question answering with E5. After receiving a batch, each query was L2-normalized, and prepended with the string "query: " before searching. All retrieval was based on brute-force similarity search, which consists of encoding the processed queries and searching the index based on cosine-similarity.

To establish a baseline against which neural search could be measured, BM25 retrieval was occasionally used. For the DR News Quiz setting, BM25-retrieval was achieved using the rank_bm25 python package (Brown, 2019), while in the Citizenship test setting it was implemented as a bespoke hybrid solution using sparse matrices to allow for efficient scaling across more than a million chunks without leaving a Python environment.

Preliminary testing of retrieval methods was done by comparing the retrieval accuracy of various approaches on the Danish SQuAD 1.0 development set, simply treating the passages in the development set as a knowledge base. These preliminary tests showed $\text{E5}_{\text{large-instruct}}$ outperforming a slew of competitors, thus it was chosen as the primary embedding model to use.

Additionally, to develop a quantitative measure of retrieval accuracy, retrieval was tested on a corpus of 100,000 synthetic query/passage pairs. Further evaluation of retrieval quality was done by comparing downstream accuracy scores when provided with different quantities of retrieved contexts.

3.4 Experiments

3.4.1 Citizenship Test

The Citizenship test was approached both as a multiple-choice QA task (its original setting), and as an open-domain QA task. In the open-domain setting, models were simply not given the multiple choice options, and were prompted to answer the question via generative sampling using the prompt in figure A.1.4. Answers were then evaluated using LLM-as-a-judge.

In the multiple-choice setting, the prompt from figure A.1.5 was given to models. This prompt is almost identical to the one used by (Holm et al., 2025), save for the retrieved context, and the sentence asking models to answer based on it. Importantly, the retriever was **not** given the multiple choice options, and therefore retrieved results are identical in both the multiple-choice and open-domain settings. This was done under the assumption that many answer options are named entities that would make retrieval too easy.

To quantify the impact of retrieval, question answering was performed with and without top-5 retrieval for each of the five generative LLMs in the catalogue, in both the open-domain and multiple-choice settings, using mE5$_{\text{large-instruct}}$ as the retriever, and the wiki-dump knowledge base described in the previous section. Additional tests were performed using Gemma-2-9b-it as the generator but with different quantities of retrieved passages, both in the open-domain and multiple-choice settings. To establish a baseline for retrieval, a single round of QA was performed with BM25 top-5 retrieval.

3.4.2 DR News

The exact same suite of experiments was performed for the DR News Quiz, but using the DR News Corpus as a knowledge base.

3.4.3 LLM-as-a-judge

To test the quality of evaluations produced by the LLM-as-a-judge framework, 7 human annotators were asked to evaluate a set of 100 generated/reference answer pairs, with the majority vote considered a gold standard evaluation. The generated answers were produced by SnakModel-7b-instruct in the open-domain setting of the Citizenship test. To create a balanced set, 50 each of answers marked incorrect and correct by Gemma-2-9b-it were sampled randomly. The questions were presented to annotators in a Google form, with instructions identical to those given to models during evaluation. The form allows questions to be unanswered. Blank answers were assigned a value equal to the reciprocal of the annotator counts, so as to be marked as different when comparing evaluations between annotators, but to be neutral with respect to the arithmetic mean.

To determine which model's evaluations were most closely aligned with human judgement, a gold standard evaluation of each answer in the 100-question subset was constructed by taking the majority vote human evaluation.

3.4.4 E5 Fine-tuning

To test whether Danish RAG-frameworks benefit from domain-specific fine-tuning of the embedding model used for retrieval, a pre-trained mE5$_{\text{large}}$ model was trained with InfoNCE loss using in-batch negatives on a 79/1/20 random train/val/test split of the Danish subset of the WebFAQ question-answering dataset. Training was performed with AdamW optmization with an initial learning rate of $2 \cdot 10^{-5}$. The model was trained with an early stopping objective consisting of three consecutive epochs of non-improvement in the validation loss, on a cluster of 4 NVIDIA 48 GB l40s GPUs on the Hendrix cluster hosted by the University of Copenhagen.

Chapter 4

Results

4.1 Citizenship Test

Gemma-2-9b-it performed best in both the open-domain and multiple-choice settings, with and without retrieved contexts. In the open-domain setting without retrieval, Snakmodel-7b-instruct came in second, but was the worst performing model when provided with context, and the only model whose performance decreased with retrieval.

The performance of Gemma-2-9b-it was much higher in the multiple choice setting than in the open-domain setting. This disparity was not consistent across models however. Snakmodel-7b-instruct was unable to follow the prompt instructions, consistently prepending its answers with “Svaret er mulighed” (English: “*The answer is option*”)

A selection of the easiest and most difficult questions in the open-domain setting, based on average score achieved by all models with retrieval, can be found in table 4.4, while the questions most impacted by retrieval can be found in table 4.5. Interestingly, retrieval consistently made several unanswerable questions answerable, but also made several questions answerable from parametric memory unanswerable across the board. Also notable is the fact that the majority of questions whose accuracy deteriorated with retrieval are questions beginning with the phrase “Hvilken af følgende” (English: *Which of the following*).

Model	**No retrieval**	**top-5 retrieval**
Gemma-2-9b-it	**81.0**	**92.1**
Snakmodel-7b-instruct	0.00 (?)	0.02 (?)
Nous-Hermes-2-Mistral-7B-DPO	61.8	74.5
Suzume-llama-3-8B-multilingual	58.5	73.2
Yi-34B-Chat	55.9	74.7

Table 4.1: Accuracy scores with and without retrieval for various models in the multiple-choice setting for the Citizenship test

Model	No retrieval	top-5 retrieval
Gemma-2-9b-it	**70.1**	**76.4**
Snakmodel-7b-instruct	68.6	63.3
Nous-Hermes-2-Mistral-7B-DPO	59.2	73.7
Suzume-llama-3-8B-multilingual	63.0	72.6
Yi-34B-Chat	56.7	73.7

Table 4.2: Accuracy scores with and without retrieval for various models in open-domain setting for the Citizenship test, evaluated with LLM-as-a-judge using Gemma-9b-it

Model	No retrieval	top-1	top-5	top-10	top-25
Gemma-2-9b-it	70.1	66.6	76.4	77.2	**78.5**
(with BM25 retrieval)	-	-	62.0	-	-
(multiple-choice)	81.0	85.5	92.1	94.4	-

Table 4.3: Accuracy scores for Gemma-2-9b-it on the Citizenship quiz in the open domain setting without retrieval and with top-k retrieval for different values of k. Retrieving a single results decreases accuracy, after which increasing k improves accuracy but with diminishing returns

Easiest questions	**Average score**
Hvilken helligdag bliver afskaffet fra 2024? *Which holiday was removed in 2024?*	1
På hvilken ø ligger Danmarks sydligste punkt? *On which island is Denmark's southernmost point?*	1
Der er 50 ugers barsels- og forældre orlov i Danmark, hvoraf 32 uger frit kan fordeles mellem moren og faren. Hvor mange ugers orlov har faren herudover ret til? *There are 50 weeks parental leave in Denmark, of which 32 weeks can be freely distributed between the mother and the father. How many weeks of paternal leave does the father have a right to in addition to these?*	1
Hardest questions	
Hvad hedder farvandet mellem København og Sverige? *What is the name of the body of water between Copenhagen and Sweden?*	0
Hvilken dansk komponist er kendt for blandt andet sine operaer? *Which Danish composer is know among other things for his operas?*	0
Hvilken dansk film vandt en Oscar i 2021? *Which Danish movie won an Oscar in 2021?*	0

Table 4.4: Easiest and hardest questions in the open-domain setting of the Citizenship test, defined by the average accuracy assigned by LLM-as-a-judge with Gemma-9b-it across all models with top-5 retrieval

Easiest questions	**Average Δ**
Hvad er Karen Blixen især kendt for?	1
What is Karen Blixen known for in particular?	
Hvilken domstol dømte tidligere udlændinge- og integrationsminister Inger Støjberg for ulovligt at have adskilt asylsøgende par?	1
Which court convicted former minister of integration Inger Støjberg for illegally having separated ayslum seeking couples?	
Er Danmark medlem af NATO?	1
Is Denmark a member of Nato?	
Hardest questions	
Hvilken rettighed er beskyttet af den danske grundlov?	-1
Which right is protected by the Danish constitution?	
Hvilket af følgende lande er i rigsfællesskab med Danmark?	-1
Which of the following countries is in the Danish commonwealth?	
Hvor mange ugers ferie har lønmodtagere på det danske arbejdsmarked normalt ret til?	-1
How many weeks of vacation are employees in the Danish labor sector usually entitled to?	

Table 4.5: Questions most impacted by retrieval in the open-domain setting of the Citizenship test, defined by average change in evaluation score assigned by LLM-as-a-judge with Gemma-2-9b-it across all models before and after top-5 retrieval. Note that $\Delta = 1$ means that all models answered incorrectly without retrieved context, and correctly with, while $\Delta = -1$ means the opposite

4.2 DR News Quiz

Consistent with the results for the Citizenship test, Gemma-2-9b-it was the highest performing model in both the open-domain and multiple choice settings, as can be seen from tables 4.7 and 4.8 respectively. Unless otherwise specified, all retrieval was performed using $\text{E5}_{\text{large-instruct}}$.

Open-domain performance of Gemma-9b-it increased consistently with retrieval, with a notable boost in accuracy (15.2%, see table 4.6) achieved by simply including the single best-scoring context. Performance further improves with the number of results retrieved, but with diminishing returns. Doubling the quantity of retrieved contexts (from 5 to 10) yields an accuracy increase of only 2.9%. When 25 contexts are retrieved, performance deteriorates. Multiple choice performance increases with k up to and including $k = 25$, but with strongly diminishing returns.

Open-domain question answering capabilities with no context were conistently low, as can be expected for questions relating to events occurring after the knowledge cut-off date, though Gemma-2-9b-it and Snak-Model-7b-instruct are notable standouts. The inclusion of retrieved contexts boosts performance universally, with all models achieving above 40% accuracy when provided with top-5 results. This convergence to a similar accuracy range is a noteworthy result.

Accuracy scores in the multiple choice setting (table 4.8) were higher than open-domain for every model, both with and without retrieval, with the exception of Snakmodel-7b-instruct, which as the sole model consistently repeated part of the prompt, breaking the character extraction on which the accuracy is based. To account for this, custom accuracy scores were calculated and included in parentheses in for completeness. Without retrieval, performance for Nouse-Hermes-Mistral-7B-DPO almost exactly matches the expected accuracy of random guessing, while Yi-34B-Chat falls short of even this standard.

Model	No retrieval	top-1	top-5	top-10	top-25
Gemma-2-9b-it	18.1	33.3	45.7	**48.6**	44.8
(with BM25 retrieval)	-	-	33.3	-	-
(multiple-choice)	56.2	73.3	79.0	81.0	**81.9**

Table 4.6: Accuracy scores for Gemma-2-9b-it on the DR News Quiz in the open domain and multiple settings without retrieval and with top-k retrieval for different values of k

Model	No retrieval	top-5 retrieval	Δ
Gemma-2-9b-it	**18.1**	**45.7**	27.6
Snakmodel-7b-instruct	10.5	44.8	34.3
Nous-Hermes-2-Mistral-7B-DPO	4.76	41.0	**36.2**
Suzume-llama-3-8B-multilingual	9.52	42.9	33.4
Yi-34B-Chat	3.81	40.0	**36.2**
Mean	9.39	42.9	33.51

Table 4.7: Accuracy scores with and without retrieval for various models in the open-domain setting for the DR News Quiz, evaluated with LLM-as-a-judge using Gemma-2-9b-it

Model	No retrieval	top-5 retrieval
Gemma-2-9b-it	**56.2**	**79.0**
Snakmodel-7b-instruct	0.00 (?)	0.00 (?)
Nous-Hermes-2-Mistral-7B-DPO	36.2	56.2
Suzume-llama-3-8B-multilingual	35.2	61.0
Yi-34B-Chat	28.6	48.6

Table 4.8: Accuracy scores with and without retrieval for various models in the multiple-choice setting for the DR News Quiz, evaluated with LLM-as-a-judge using Gemma-2-9b-it

4.3 LLM-as-a-judge

The pairwise evaluation agreement between models and the gold standard consisting of human majority vote is plotted in figure 4.3.1.

Overall, evaluations produced by Gemma-2-9b-it were closest to the gold standard, and thus most aligned with human evaluation. Human evaluations showed significant variation. Inter-annotator agreement was on average 88.5, while the average agreement

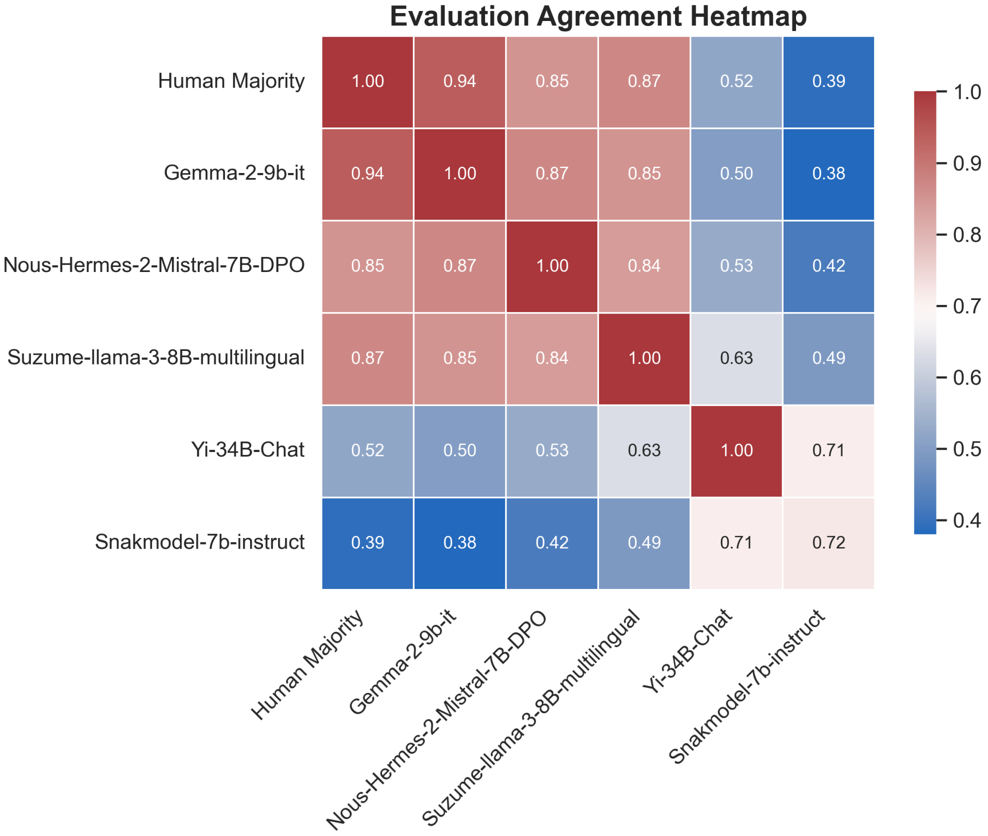


Figure 4.3.1: Pairwise evaluation agreement between different models and majority vote human annotation on the 100-question subset

Model	Accuracy	Recall	Precision	F1
Gemma-2-9b-it	94.0	96.0	92.3	94.1

Table 4.9: Accuracy, precision, recall and F1 scores of evaluations produced by LLM-as-a-judge evaluation by various models for a 100-answer subset, assessed against the gold standard

between individual human annotator and the gold standard was 92.3. Thus Gemma-9b-it is actually marginally *more* aligned with the human average than the average human is.

Since Gemma was the best scoring model, its evaluations were further assessed against the gold standard and evaluated on accuracy, recall, precision and F1, the results of which can be found in table 4.9.

On the selected subset, human majority voting evaluated 52% of generated answers as correct, whereas Gemma-2-9b-it evaluated only 50, demonstrating that human evaluation is slightly more lenient. All else being equal, a stricter model should result in lower recall than precision, suggesting that difference in leniency is not the main contributor to the disparity in accuracy.

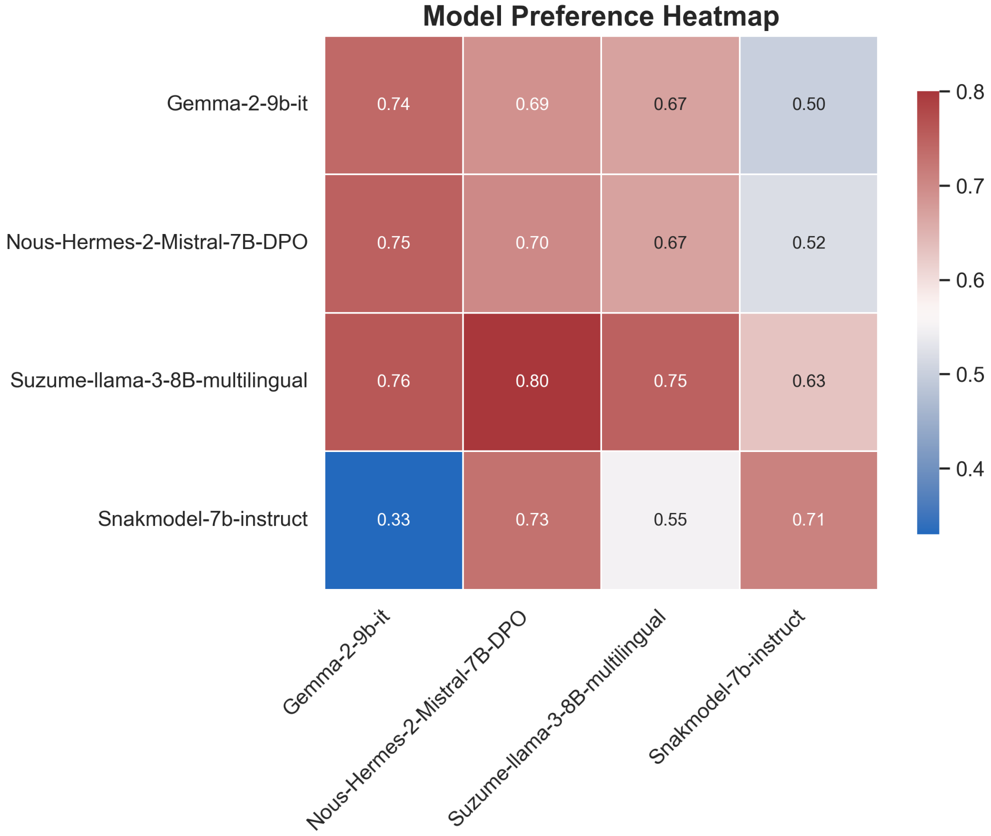


Figure 4.3.2: Pairwise evaluation preference between models. Each row represents LLM-as-a-judge scores given by a single model to different 100-question answer sets generated by every other model. The lack of symmetry otherwise common in heatmaps is a positive finding: it indicates that models do not prefer their own outputs when evaluating. The score given by Gemma to Snakmodel's answers should be considered invalid, since this was the selection criteria for the question subset

4.4 Fine-tuning E5

While fine-tuning E5$_{\text{large}}$ on WebFAQ, validation loss was calculated at regular intervals, and weights were saved at the end of every epoch. Based on the observed divergence in validation loss and training loss, a candidate embedding model was loaded from the weights at checkpoint 2 and dubbed E5$_{\text{large-FT}}$. The retrieval accuracy of E5$_{\text{large-FT}}$, along with E5$_{\text{large}}$ and E5$_{\text{large-instruct}}$ was measured for different values of k on both the synthetic retrieval data, and the test split of Danish WebFAQ, the results of which are displayed in tables 4.10 and 4.11 respectively. Due to parsing failures the synthetic dataset was reduced to a size of roughly 75k.

Unfortunately, though E5$_{\text{large-FT}}$ saw improvements in retrieval accuracy on the test set of WebFAQ, not only does it fail to generalize to the synthetic dataset, performance actually deteriorates compared to the base model E5$_{\text{large}}$. For this reason, constructing new indexes for the other tasks was decided against.

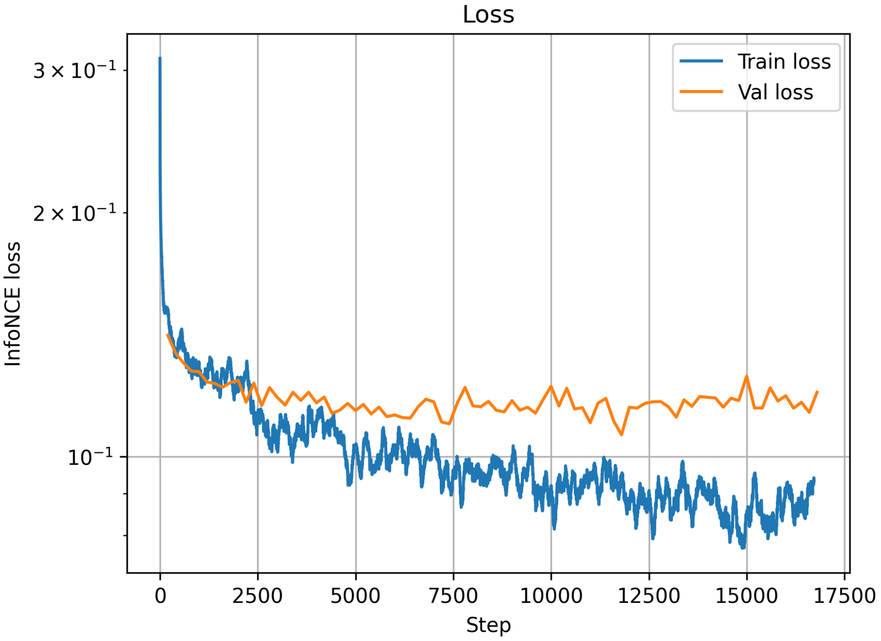


Figure 4.4.1: Validation and training loss for fine-tuning E5$_{\text{large-instruct}}$ on the Danish subset of WebFAQ. With a batch size of 256, an epoch correspons to approximately three thousand steps

Model	**top-1**	**top-5**	**top-10**	**top-25**	**top-50**	**top-100**	**top-1000**
E5$_{\text{large-instruct}}$	46.2	64.7	71.6	79.8	85.3	89.9	97.8
E5$_{\text{large}}$	46.6	65.6	71.4	79.3	84.9	89.4	-
E5$_{\text{large-FT}}$	39.2	57.0	64.1	73.1	79.4	85.2	-

Table 4.10: Retrieval accuracy@k on synthetic retrieval data for different values of k

Model	**top-1**	**top-5**	**top-10**	**top-25**	**top-50**	**top-100**	**top-1000**
E5$_{\text{large-instruct}}$	61.2	74.4	77.7	80.5	82.8	85.1	-
E5$_{\text{large}}$	68.2	78.1	80.2	82.7	84.7	86.7	-
E5$_{\text{large-FT}}$	72.6	81.2	84.6	87.8	90.1	92.2	-

Table 4.11: Retrieval accuracy@k on the Danish subset of WebFAQ for different values of k. Top-1000 was left out to conserve compute resources. To avoid contamination, values on the bottom row are only calculated for the test set

Chapter 5

Discussion & Analysis

The most immediate result from the previous section is an affirmative answer to the central question of this thesis: RAG frameworks built upon open source models perform capably on Danish question-answering tasks.

This is most clear from the results presented in 4.1 and 4.2. Since performance generally improves with retrieval, it can be assumed that the models are incorporarating non-parametric knowledge into the generation process. However, the fact that accuracy does not increase to 100% raises a simple question - why not? This section will try to answer that question.

5.1 Diagnosing Performance

Diagnosing performance in RAG systems can be difficult, for several reasons. Chief among them is the modular nature of the RAG framework. Each of the three components (retriever, generator and knowledge base) is a potential source of error. Therefore, any reasonable attempt to diagnose a RAG failure should first attempt to attribute the failure to a specific component. To this end, each negative result in separate answer sets produced by Gemma-2-9b-it was inspected manually and classified according to the following taxonomy:

1. **False negative** (FN): Correct answers that were incorrectly evaluated as incorrect.

2. **Answerability failure** (ANS): Incorrect answers that were caused by unanswerable questions.

3. **Generation failure** (GEN): Incorrect answers that were incorrect despite the presence of the answer in the retrieved context

4. **Retrieval failure** (RET): Incorrect answers for which the retrieved context did not contain the answer

5. **Knowledge base failure** (KB): Incorrect answers derived from incorrect information in the knowledge base

Scenario	**ANS**	**FN**	**GEN**	**RET**	**KB**
DR News quiz (open-domain)	17	9	3	**28**	-
DR News quiz (multiple-choice)	5	-	-	**17**	-
Citizenship test (open-domain)	**53**	31	14	39	5
Citizenship test (multiple-choice)	1	-	8	**36**	3

Table 5.1: Caption

The results of the taxonomy are presented in 5.1. The vast majority of incorrect answers are due to retrieval failure. The second largest source of error came from unanswerable questions, particularly those that had become unanswerable in the absence of the multiple-choice options. The third largest was false negatives produced by the LLM-as-a-judge evaluation method. Generation produced relatively few errors, and errors in the knowledge base contributed a negligible amount, several of which were duplicates.

The following subsection will present an analysis of each failure source, based on select examples.

5.1.1 Answerability failures

Question: Hvilken af følgende byer har flest indbyggere?

Which of the following towns has the most citizens?

Answer: Esbjerg

As could be expected, many questions in the multiple-choice setting became unanswerable when viewed as open-domain questions. This is particularly true for questions of the form “Hvilken af følgende...” (English: *Which of the following*), variants of which were ubiquitous in the Citizenship test. As an example, variants of the question, “Hvilken rettighed er sikret i grundloven” (English: *Which right is guaranteed by the constitution?*) appear 9 times, with correct answers spanning five different rights.

Question: Ved hvilken bygning blev der begået et terrorangreb i 2015?

Which building was the site of a terrorist attack in 2015?

Generated Answer: Bataclan

Reference Answer: København s [sic] Synagoge

Occasionally the error for a question could arguably be attributed to multiple sources. The question above illustrates this. Is it reasonable to assume this question refers to a terrorist attack in Denmark? And if so, should this assumption be influence the retrieval, or just the generator?

While questions that had become ambiguous in this way were abundant in both datasets, the DR News Quiz also contained numerous questions that were unanswerable even when paired with answer options. These include questions that had originally been presented along with an embedded video player when displayed in the browser. These usually refer to the contents of the video implicitly with the word “her” (English: *here*), and rarely make sense as stand-alone questions.

Question: En kajakroer bliver her fanget i en hvals gab. Hvilken slags hval er der tale om?

[In this video] *a kayaker is caught in mouth of a whale. What kind of whale?*

Answer: En pukkelhval

A hump-back whale

Besides the ambiguities caused by absent answer options or missing video content, some questions are plague by an implicit temporal element. Questions such as “Hvad hedder Radikale Venstres politiske leder?” (English: *What is the name of Radikale Venstre's* [political party] *political leader*?) are listed with the answer that was true at the time the test was administered, but is no longer true. This is, somewhat surprisingly, the only error of this kind in the Citizenship test. In the DR News Quiz the question “Donald Trump lufter her muligheden for, at USA overtager..?” (English: *Here* [referring to a video] *Donald Trump is airing the possibility for the USA to take over…?*) is from week 6, 2025. The correct answer is “Gaza-striben” (English: *the Gaza strip*). Many models however focus on similar comments from the president regarding Greenland, and thus answer incorrectly.

5.1.2 False Negatives

Question: Hvad er det USA's nytiltrådte præsident, Donald Trump, her finder i skrivebordet i Det Hvide Hus?

[In this video] *what does the USA's newly inaugarated president Donald Trump find in the White House desk?*

Generated Answer: Et brev adresseret til “47”

A letter adressed to “47”

Reference Answer: Et brev fra Joe Biden

A letter from Joe Biden

Evaluation: 0. The generated answer is not a match for the reference answer. both answers mention a letter, the content of the letter is different. The reference answer specifies the letter is from Joe Biden, which is a crucial detail missing from the generated answer.

The prompts used for evaluation were the only set of prompts that went through a multi-stage development process. Early versions were much too critical, and seemed to be unable to distinguish between which of the answers was the ground truth. More verbose and seemingly more detailed generated answers were often evaluated as correct, despite not matching the reference answer. Prompts that overemphasized the importance of similarity between the answers would produce false negatives when generated answers were more specific, but still correct. The final iteration thus emphasizes that answers should match, but explicitly states that generated answers more specific than the reference answer should be considered true. The full prompt is presented in section ??

Arriving at this prompt was largely a heuristic process, based on manual inspection of the evaluations. In an exercise in machine psychology, models were instructed to explain their answer, and these explanations were used to guide the process.[1]

Despite the careful prompting, and despite being most closely aligned with human annotators, evaluations produced by Gemma-2-9b-it were occasionally comically unpredictable. The evaluations, and the validity of the whole framework, are therefore the subject of a dedicated section presented later in this chapter.

Question:	Donald Trumps søn Donald Trump Jr. inviterede grønlændere til frokost under sit besøg i Nuuk. Hvem er med på telefonen her? *Donald Trump's son Donal Trump Jr. invited Greenlanders to lunch during his visit in Nuuk. Who is joining by phone here* [in the video]?
Generated Answer:	Donald Trump
Reference Answer:	Sin far, Donald Trump *His father, Donald Trump*
Evaluation:	0. While both answers identify Donald Trump as being involved, the generated answer is too general. The reference answer is more specific by stating that Donald Trump Jr. invited Greenlanders to lunch with his **father**, Donald Trump.

5.1.3 Generation Failures

Generation failures for Gemma-2-9b-it were relatively rare, contributing only 25 errors across all four experiments. Though this may be surprising, it is worth keeping in mind that Gemma-2-9b-it was chosen specifically based on its performance. Supremacy notwithstanding, Gemma still occasionally displays surprising failure modes. In some instances, such as in 5.3 generation failures were obvious and easy to classify. In these cases they were however difficult to explain. Table 5.4 presents an especially egregious example.

Others were harder to classify. In a common occurrence, both the generator and the retriever would interpret questions in an overly general manner. See table 5.5 for an example in which a questions about "grundloven" (English: *the Constitution*) is interpreted by both models as being about the Romanian constitution. Deciding how to classify this error demands a certain level of discretion. On the one hand, models have not been informed that the questions are part of a test administered by the Danish government to quiz prospective citizens on their knowledge of Danish history and culture. On the other hand, the language in which the question is asked is a strong context clue. Interestingly, when asked the same question in English, the model gives an answer based on retrieved context relating to the most recent constitutional change in Thailand. This is interesting for two reasons. The first is that the result is retrieved from Danish Wikipedia, meaning that cosine similarity search works with bilingual query-passage pairs. The second is that

[1] Although Alan Turing is no longer with us, his thoughts on this exercise would doubtlessly be illuminating

Question:	Flere og flere formår - som 111-årige Kirsten Schwalbe her - at leve 99 år eller derover. Hvor mange danskere er i dag så gamle? *Like 111-year-old Kirsten Schwalbe here [in the video], more and more people manage to live to the age of 99 or above. How many Danes are currently this old?*
Emphasized context:	Title: Danmarks ældste fylder 111 år Text: Ifølge Ritzau bor der 1.265 personer i Danmark, som er over 100 år gamle. Men Kirsten Schwalbe er den eneste over 110 år. *Title: Denmark's oldest turns 111 years old* *Text: According to Ritzau over 1265 people living in Denmark are over 100 years old. But only Kirsten Schwalbe is above 110*
Generated Answer:	1256
Reference Answer:	2000 danskere (*Danes*)

Table 5.2: A generation failure that was originally misclassified as a knowledge-base failure. The question asks about people aged 99 or above, while the statistic refers to people aged over 100. A forgiveable mistake, all things considered.

(in this one particular instance) the retriever doesn't seem biased towards results relating to the language of the question.

Besides overlap with retrieval failures, some generation failures could arguably be classified as false negatives. When asked about women's suffrage in Denmark, Gemma provides the year of the first election Women could participate in, as opposed to the year that suffrage was adopted into law. Both years are present in the retrieved context, a mistake that feels almost forgivable. Conversely, the only two generation failures in the DR News Quiz are both clear cases of low-level language deficiency. In the first, Gemma performs an incorrect co-reference resolution resolution, incorrectly interpreting the Danish commonwealth as the target of speech by the Danish minister of Defense (the words were actually targeted at Trump, in *defense* of the commonwealth). In another, when asked to list the most common names among senders of hateful messages on social media, it instead lists ages.

Question:	Er det normalt gratis for voksne borgere at gå til læge? *Are doctor's visits usually free for adult citizens?*
Emphasized context:	Title: Tandlæge Text: I Danmark er tandpleje (tandlægebesøg) gratis for børn og unge under 18 år. Til voksne ydes der tilskud via Den Offentlige Sygesikring, men i modsætning til fx lægebesøg og hospitalsindlæggelser er det ikke gratis. Alle danske tandlæger arbejder under overenskomst mellem Tandlægeforeningen og Regionernes Lønnings- og Takstnævn. *Title: Dentist* *Text: In Denmark dentistry is free for children and youth under the age of 18. For adults a subsidy is provided through* [danish public health insurance,] *but as opposed to e.g. doctor's visits or hospitalizations, it is not free. All Danish dentist's work under the labour agreement between* [relevant parties]
Generated Answer:	Nej. (*No.*)
Reference Answer:	Ja. (*Yes.*)

Table 5.3: A sample question from the Citizenship test, with an answer generated by Gemma-2-9b-it in the open-domain setting, accompanied with context from retrieval. The context displayed is one of 5 passages retrieved, and emphasized because of its relevance.

Question:	I hvilke årtier udspiller det meste af tv-serien Matador sig? *During which decades does the majority of the tv-show Matador take place?*
Emphasized context:	Title: Matador (tv-serie) Text: Serien følger livet i en dansk provinsby fra 1929 til 1947. (...) *Title: Matador (tv-show)* *Text: The show follows life in a small provincial Danish village from 1929 to 1947 (...)*
Generated Answer:	1920'erne og 1930'erne. (*the 1920s and 1930s*)
Reference Answer:	1930'erne og 1940'erne. (*the 1930s and 1940s*)

Table 5.4: Another sample question from the Citizenship test, with an answer generated by Gemma-2-9b-it in the open-domain setting, accompanied by relevant context from retrieval. Some failure cases are difficult to diagnose

Question:	Hvornår blev grundloven sidst ændret? *When was the constiution last changed?*
Emphasized context:	Title: Rumænienes grundlov Text: Den nuværende grundlov stammer fra 1991 og blev sidst ændret efter en national folkeafstemning den 18.-19. oktober 2003 (...) *Title: The Romanian Constitution* *Text: The current constitution originates in 1991 and was last changed after a national referendum the 18.-19. October 2003 (...)*
Generated Answer:	Den 29. oktober 2003.
Reference Answer:	1953

Table 5.5: Another sample question from the Citizenship test, with an answer generated by Gemma-2-9b-it in the open-domain setting, accompanied by relevant context from retrieval. Some failure cases are difficult to diagnose

5.1.4 Retrieval Failures

The most significant result of the taxonomy is identifying retrieval errors as the largest source of failure. In all but one scenario, retrieval failures were the biggest source of errors by a considerable margin. In both multiple-choice settings, retrieval errors constituted not just a plurality but a majority of errors. As such, the performance of retrieval deserves some analysis.

Diagnosing retrieval failures presents its own set of challenges. In artificial environments such as SQuAD, each question is paired with a specific context from the knowledge base. In both the DR news quiz and the Citizenship test however, answers to the questions are only presumed to be contained in the knowledge bases. In cases where they aren't, this can be difficult to state definitively. Nevertheless, manual search has revealed some insights. At least two articles that contain answers for the DR News Quiz have been found online, but are missing from the DR News Corpus. In one case, there was a scraping failure noted in the logs. In another, an article was findable via Google search, but wasn't scraped. It is reasonable to believe that at least some other retrieval failures are attributable to missing parts of the knowledge base. A similar assumption can be made about Wikipedia-based retrieval, qualified by a number of observations:

Firstly the Danish Wikipedia is far less exhaustive than might be assumed. The wiki-dump used in this thesis contains 487k articles. As a comparison, (Singh et al., 2021) identified at least 6 million articles in the English Wikipedia in May of 2020. Thus the Danish Wikipedia is likely to be much less all-encompassing than its English counterpart.

Secondly, manual inspection of select results reveals deficiencies. Responding to a question in the Citizenship test, Gemma consistently answered that labour union membership is usually free. While answering without context, this is a notable parametric knowledge failure, but the answer does not improve with retrieval. Upon inspection, the Danish Wikipedia article for "Fagforening" makes no mention of whether labour unions charge dues (Wikipedia, 2025a). The corresponding English article mentions dues in the second paragraph of the lead (Wikipedia, 2025d), and a comparative glance at both articles is informative.

Since RAG improves when more results are retrieved, deficiencies in the knowledge bases are unlikely to fully account for the many retrieval errors however, since this indicates that the answer to some questions exists in the knowledge bases, but isn't found among the 5 contexts retrieved in the experiments upon which this taxonomy is performed. Therefore, a manual inspection of some of the retrieval results is warranted.

A pathological example can be found in table 5.6. The question is a straight-forward question about a simple well-defined quantity: How many municipalities are there in Denmark?. Despite this, E5$_{\text{large-instruct}}$ fails to retrieve the answer within the top-5. Why? The retrieved contexts mention the amount of municipalities in France, the Faroe Islands, Norway. The correct number for the Faroe Islands even appears twice.[2]. Context 3 (abridged in the table) lists an exhaustive history of the quantity of municipalities in Denmark, from 1965 to 2007. Even more frustratingly, the correct answer appears in the first sentence of the article from which context 3 was taken (Wikipedia, 2025b). Why was this sentence down-prioritized compared to the others?

The retrieval failures are especially surprising considering the prominence of the E5 models in various text embedding benchmarks, such as the Massive Text Embedding Benchmark (Muennighoff et al., 2023),

Unfortunately, as it stands, this thesis leaves this question unanswered. It is possible that the performance of E5$_{\text{large-instruct}}$ is deficient in Danish, though the performance in tables 4.10, 4.11, along with performance on the rest of the Citizenship test and DR News quiz dispute this. E5$_{\text{large-instruct}}$ also performs well against BM25 as a baseline, with RAG with E5 top-1 retrieval performing better with top-1 retrieval than RAG with BM25 top-5 retrieval does, for both the Citizenship test and the DR News Quiz.

One option is that the chosen chunking strategy is sup-optimal for use with E5. Measuring the impact of various chunking strategies is logistically difficult however, since any change in chunking strategy requires encoding the entirety of Wikipedia (more than a million chunks) to reconstruct the vector-index, requiring a non-trivial amount of both compute resources and time. Thus the sometimes pathological behaviour of E5 is left as an observed but unexplained result.

[2] This is not a bug, but a feature of the chunk overlap. The second mention appears in context 5, which has been left out of the table

Question:	Hvor mange kommuner er der i Danmark? *How many municipalities are there in Denmark?*
Context 1:	Title: Kommune Text: Pr. 1. januar 2009 var der 36.680 kommuner i Frankrig – heraf 112 i de oversøiske territorier.Færøerne.På Færøerne er der 29 kommuner pr. 1. januar 2017. Hver kommune har en valgt kommunalbestyrelse (eller byråd). Skotland. *Title: Municipality* *Text: As of January 1 2009 there were 36680 municipalities in France, of which 112 were in overseas territories.The Faroe Islands.On the Faroe Islands there are 29 municipalities as og January 1 2017. Each municipality has an elected municipal board (or council). Scotland*
Context 2:	Title: Kommune Text: Norge. I Norge er der 356 kommuner, pr. 1. januar 2020. *Title: Municipality* *Text: Norway. In norway there are 365 municipalities, as of January 1 2020.*
Context 3:	Title: Strukturreformen Text: Kommunernes antal og indbyggertal har varieret en del gennem tiderne. I 1965 var der flest, nemlig 1345. Der var 277 kommuner i Danmark fra 1. april 1970 (...) *Title: The Structural Reform* *Text: The quanitity and population of muncipalities has fluctuated signigificantly through the years. In 1965 there were the most, at 1345. As of April 1 1970 there were 277 municipalities in Denmark* (...)
Context 4:	Title: Kommuner i Danmark Text: Dertil kommer tre kommuner, der også fungerede som amter: Strukturer baseret på kommunerne. *Title: Municipalities in Denmark* *Text: This additonally includes three municipalities that also worked as counties: Structures based upon the municipalities*

Table 5.6: A pathological example of retrieved context for a question in the Citizenship Test. One context has been excluded, and context 3 has been abbreviated for readability.

5.1.5 Knowledge Base Failures

Though the previous sub-section detailed cases in which knowledge was presumed to be absent from knowledge bases, these cases are not considered knowledge base failures. No knowledge base can be expected to be completely exhaustive. Absences excused, this sub-section details surprising incidents in which incorrect information was found in the knowledge bases.

In the first case, presented in table 5.7, wrong answers are provided based on an article (Wikipedia, 2025c) which incorrectly states that the regent in Denmark is appointed by parliament. Though the exact nature of the Danish constitutional monarchy is complicated[3] the line of succession is not (Lars Løkke Rasmussen, 2009). This particular piece of incorrect information is the product of a Wikipedia editor misrepresenting the contents of a news article, replacing the word “udråbes” (English: *is proclaimed by*) with the word “udpeges” (English: *is appointed by*). The fact that a single rogue editor can impact the performance in this way is a consequence of the semantic search strategy of the retriever, which will retrieve relevant sounding information from an obscure article. This could potentially make RAG systems which derive information from public forums vulnerable to targeted injections of information, and is worth noting.

[3] The Danish state functions on a complicated constitutional basis that involves an *extremely* non-literal reading of the Danish constitution. I encourage readers to give the constitution in its current form a cursory reading, with an emphasis on §3

Question:	Hvordan bliver det afgjort, hver der skal være konge eller dronning i Danmark? *How is it decided who will be the new king or queen in Denmark?*
Emphasized context:	Title: Kroning Text: Repræsentativt monarki. I Danmark. I Danmark udpeges en konge (regent) af Folketinget. I praksis udråbes (proklameres) den nye konge eller dronning af Statsministeren. Kronprins Christian er jf. tronfølgeloven den næste i rækken. *Title: Coronation* *Text: Representative monarchy. In Denmark. In Denmark a king (regent) is appointed by parliament. In practice the new king or queen is shouted (proclaimed) by the prime minister. According to the* [law of royal succesion] crown prince Christian is the next in line.
Generated Answer:	Folketinget udpeger en konge (regent). *Parliament appoint a new king (regent).*
Reference Answer:	Den nye konge eller dronning arver tronen *The new king or queen inherits the throne*

Table 5.7: A failure case caused by incorrect information on Wikipedia

5.2 Interpretation of Results

This section will provide a qualitative analysis of select results, and what can be inferred from them.

Besides the general though limited viability of RAG, the other most notable holistic result from the experiments is the supremacy of Gemma-2-9b-it in the chosen suite of models. This is consistent with the Danoliterate leaderboard, which lists Gemma-2-9b-it as the highest performing open-source model on the Citizenship test.[4]

The DR News quiz was intentionally designed to be unanswerable with only parametric knowledge, since the scraped quizzes refer only to events in 2025,[5] and every model selected was released in 2024. This should make the DR News quiz the most appropriate scenario for testing RAG. Since the open-domain setting imbues some questions with unresolvable ambiguity, and the evaluation of answers is subject to errors of the LLM-as-a-judge method, the multiple-choice setting of the DR News Quiz should be the most informative with regards to RAG.

The results in table 4.8 seem to confirm this hypothesis.

[4] Sort of. Three variants of Llama 3 are listed above it, but each is listed with the suffix “@groq”, presumably indicating that the prompting was done through an API, though it’s unclear

[5] Almost. Some questions in the first quiz refer to events from very late 2024, such as the King’s speech New Year’s Eve 2024

Without access to multiple choice options, and with no retrieval, models are essentially guessing. Thus the large disparity in accuracy with and without retrieval ($\bar{\Delta} = 33.51$) aligns with expectations. The performance of Gemma-2-9b-it without retrieval is surprising though, and is subject to analysis in this subsection. The ability to infer likely answers to questions is itself a test of real-world knowledge.

In one question, Gemma is asked what the French politician Marine Le Pen has been sentenced to prison for. Le Pen was convicted of embezzling EU funds in March of 2025 (Pineau and Jabkhiro, 2025), but the trial has been ongoing since 2024. Thus the model was able to infer the right answer. Neither the question nor the answer were framed as hypotheticals however, and thus the answer, although correct, can only be viewed as a hallucination. Other answers correctly inferred by Gemma include references to annual events (Chinese New Years) or events that have happened before. Asked about the purpose of a rocket sent by NASA to the moon, Gemma correctly inferred that it was sent to look for water.

Perhaps most interestingly, when asked which group of people the Danish King gave special attention to in a speech on New Years Eve, it was able to correctly guess "De unge" (*English:* "the youth"). This example was repeated across models, with every model except Yi-34b-Chat guessing this answer correctly without retrieval. Astoundingly, an article describing the speech has not been included in the DR News Corpus due to a scraping failure, and therefore every single model answers incorrectly when provided with retrieved context, with many models incorrectly focusing on a speech delivered by the Dutch King Willem-Alexander, which focused on jews and muslims, despite the passage attributing the statements to him by name.

The example above illustrates a common pattern in which models have difficulty identifying and disregarding irrelevant context. This could perhaps serve as an explanation for the counterintuitive result in table 4.5, in which retrieving a single context decreases the performance of the model on the Citizenship test. It would be intuitive to think that the difference in accuracy is due to a couple of questions which were negatively impacted by retrieval. This is not the case however. Evaluations of the two answer sets only agree on roughly 75% of the answers. Providing a single retrieved passage causes Gemma to provide incorrect answers to 87 questions that it had answered correctly without context, and conversely to correctly answer 66 questions that it had previously answered incorrectly.

This behavior is exemplary of the unpredictability of language models. With that being said, the prompting strategies may be in part to blame. The prompt in the open-domain setting (see figure A.1.4) instructs models to answer "ud fra konteksten" (English: *based on the context*). Thus it is understandable that models would provide answers based on context, even if the answers contradict their parametric knowledge. Instruct-tuned models in particular could be susceptible to this behavior, since following instructions is what they have been fine-tuned to do.

It is possible that careful prompting could have allowed for models to evaluate the retrieved context against their parametric knowledge, and try to provide answers consistent with both. Models could have also been asked to evaluate whether context was relevant. In fact the existing task of claim verification, which asks models to determine whether a given text is supported by a given piece of evidence, is directly related. In this context, the DanFEVER claim verification dataset and associated experiments may provide evidence of the ability of Danish language models to do this. Likewise, the existing body of research on re-ranking may be relevant. Strategies for ignoring irrelevant context don't have to be implemented by the generator. Since retrieval is based on cosine similarity, a simple approach could have implemented a minimum threshold for the similarity score, and not provided the model with contexts below this threshold.

The prompts provided in the multiple choice setting state explicitly that context may or may not be relevant. This is not true in the open-domain setting, and is a clear oversight, but may provide interesting results. Despite this prompting, the answer patterns in the multiple choice setting vary significantly depending on whether or not context is provided. Providing a single context passage causes Gemma to change it's answer to 111 questions, only 69 of which are correct. Clearly, instructing the model that context may not be relevant is not sufficient.

(Röttger et al., 2024) showed that forcing models to pick from multiple choice answers affects their answers in an unpredictable way, and that models answer inconsistently depending on how they are prompted.

There are many other potential tweaks that could have been attempted, but prompting has been kept intentionally consistent in order to avoid a combinatorial explosion of test scenarios, and so analysis will focus on the results collected with the prompting and retrieval strategies outlined in the methodology section.

5.3 Validity of Metrics

The hard evaluation metric used for evaluating answers in the multiple-choice settings can be assumed to be a better metric than the soft evaluation. When viewing the parsing of answers (which simply extracts the first character) as part of the generative process, this hard evaluation is actually equivalent to the Exact Match metric, which has been standard in the literature since (Rajpurkar et al., 2016). Despite this, the metric produces what can be considered false negatives, most notably when evaluating SnakModel-7b-instruct. In almost every single answer in the multiple-choice setting, SnakModel repeated part of the prompt, causing its answer to fail the EM evaluation. This is what causes the low scores in tables 4.1 and 4.8. Whether or not this is fair to SnakModel is worth discussing.

On the one hand, the user prompt states "Svar kun med bogstavet for den rigtige mulighed" (English: *Answer only with the letter of the right option*), and SnakModel is an instruct-tuned model. The answers produced clearly do not adhere to this instruction. On the other hand, since SnakModel the only model designed specifically for Danish, it received a Danish system prompt. This system prompt was not a direct translation of the English system prompt given to the other models, and doesn't include equivalents of the several statements encouraging brevity. The fact that it was the only model to receive a different system prompt, and also the only model to consistently be unable to follow instructions is noteworthy.

More obviously, evaluations produced by the LLM-as-a-judge framework deserve skepticism. Addressing this skepticism was the purpose of the human annotation experiment outlined in section 4. The results of this experiment can only be described as extremely favorable to both the metric and to Gemma. A conspicuous note however is that Gemma-2-9b-it was chosen as the main evaluator, and also consistently awarded itself the highest score. Is it possible that Gemma is biased towards its own outputs? During evaluation it is unaware that the answers were produced by itself, bit is plausible that the latent representations during decoding affect the output. By definition, outputs produced by Gemma are sequences of tokens to which Gemma assigns a high probability. Since Gemma's evaluations were most closely aligned with human evaluation however, and since evaluation with the hard metric of the multiple-choice setting also favors Gemma, this is considered unlikely. Nonetheless, an experiment was performed in which answers produced by each model were evaluated by every other model in order to check for model-specific biases. The results of this experiment are presented in table 4.3.2, and dispute this idea.

5.4 Prompting Strategies

Though a design choice was made not to test various prompting strategies, a discussion of the prompts used is warranted.

The first point worth mentioning is that the prompts as presented in sub-section 3.3 do not accurately represent the prompts as presented to the models. The prompts are formatted using the Huggingface `apply_chat_template` function, which inserts special tokens between system and user prompts. When invoked with the argument `add_generation_prompt`, it additionally adds another special token to indicate the beginning of the response of the model. This format is intended to align input prompts with synthetic chats that a model may have seen during training, but is specifically designed for models trained to produce multi-turn dialogue. Whether or not it was appropriate to apply the template to instruct-tuned models is unclear.

In particular, since the multiple-choice prompt essentially asks models to continue the prompt, it could have been passed with the argument `continue_final_message`, which does not append the user prompt with a special token. But since this thesis is intended in part to serve as an advisory document to CAISA, it was deemed more appropriate to simulate a step of multi-turn dialogue, since this is the likely implementation form that public sector RAG systems would take.

5.5 Future Research

While the central question of the thesis was directly addressed and answered, there is much room for improvement, and the results are lacking in many ways. In particular, the revelation that retrieval failures were the largest source of error leads naturally to the conclusion that more attention should have been paid to the retrieval strategy. It is possible that the performance observed from mE5$_{\text{large-instruct}}$ is representative of the out-of-the-box retrieval capacity of frontier open-source multilingual embedding models in Danish, and that the limitations encountered are the product of hard technical limits. Conversely, the performance could be a result of sub-par implementation. There are several ways in which this could be true:

It is possible that performance was handicapped by a sub-optimal chunking strategy. Emphasis during chunking was placed on producing semantically coherent chunks, split at the sentence or paragraph level. For the wiki-dump, this produced chunks with an average length of 50.9 words (counting punctuation). It is possible that this may be too short to properly capture semantic content. When building the index, each chunk is encoded independently of its neighbors, and semantic search over the index has no notion of locality. Thus critical information distributed across several chunks will be retrieved only if each chunk is deemed relevant independently. This is true even for adjacent chunks. The chunk overlap is intended to counteract this fact, but without proper testing it is impossible to say whether it was sufficient. Further research should could prioritize determining the degree to which chunking impacts downstream accuracy.

It is also possible that a different choice of embedding model could have yielded higher performance. Language-specific retrieval benchmarks are mostly limited to high-resource languages, and to my knowledge no Danish retrieval benchmark exists, making educated guesses difficult. This lack of a canonical benchmark is critical, and should serve as a research priority. To this end several candidates exist. The Scandinavian Embedding Benchmark (Enevoldsen et al., 2024) mentioned in section 2.5 subdivides its 24 tasks into 4 categories, including retrieval, in which it lists the datasets twitterhjerne, TV2Nord Retrieval, and DanFEVER (Nørregaard and Derczynski, 2021) as examples. The first task originates in (Holm et al., 2025), and consists of only a few hundred examples. The second is originally a summarization dataset, and becomes a retrieval task only by viewing a news article and its summary as a positive pair. The final dataset is claim-verification dataset, intentionally designed to mimic the English FEVER dataset (Thorne et al., 2018).

Of these three, DanFEVER is the best candidate, for several reasons. Each claim in DanFEVER is listed along with evidence. Where the evidence is derived from Wikipedia, it is listed with a unique identifier to the source article. This makes automated retrieval scoring possible, while using the entire wikidump as the knowledge base. Perhaps more interestingly, claims tagged as “Refuted” or “NotEnoughInfo” could be used as hard negatives when fine-tuning an embedding model. This is a potentially fruitful avenue of research that is, to my knowledge, unexplored.

Indeed, determining the degree to which an embedding model used in retrieval benefits from Danish fine-tuning was the intended target of the E5 fine-tuning experiment that unfortunately yielded poor results. Trying to determin why was the target of extensive trouble-shooting, but nothing conclusive was learned.

Chapter 6

Conclusion

Though there is a rich body of literature focused on Danish NLP, and a larger body yet focused on Scandinavian or generalized multilingual NLP, research is related to Danish-language RAG is lacking. Besides a research gap with regards to the RAG-capabilities of generative models, this deficiency is expressed in a lack of standardized retrieval benchmarks as well as large-scale question answering datasets. While this thesis attempts to take a tentative first step, more research is warranted.

The results in this thesis indicate that retrieval augmented generation as an approach to question answering in Danish is viable, and can be achieved using only open-source models without any fine-tuning. Of several models tested for generation, Gemma-2-9b-it was the most capable, as demonstrated by its performance on the DR News Quiz and the Citizenship test. RAG is particularly beneficial in settings involving real world knowledge about specific events that have occurred after a model was trained. Increasing the amount of retrieved context generally improves the performance of a model, but the improvements show diminishing returns. The performance of models will sometimes deteriorate when provided with irrelevant context, causing them to answer incorrectly even though the answer was contained in their parametric knowledge, or could at least be inferred from it.

When paired with a reference answer, the answers generated by one model during open-domain question answering can be evaluated using another model. For a sufficiently capable model, the evaluations are well aligned with human evaluations, with Gemma 2-9b-it even surpassing the average performance of an individual annotator with respect to gold standard derived from majority vote. Despite this, performance on individual samples is sometimes unpredictable. When acting as judges, models do not give preference to their own answers.

The recent emergence of large natural language question answering datasets in Danish, such as the Danish subset of WebFAQ, or synthetic datasets, presents an interesting opportunity for fine-tuning embedding models for retrieval. Direct application of InfoNCE loss with in-batch negatives however yields disappointing results. Nonetheless, they remain as targets. NLP in other languages has evolved around a culture of standardized benchmarks, and hopefully the emergence of similar norms in Danish NLP will help foster inter-textuality and drive innovation in the field.

Appendix A

Appendix

A.1 Prompts

System Prompt (Open Domain)

```
You are a helpful assistant. You respond to questions in
Danish.
Respond briefly and accurately.
Do not generate any extra questions or superfluous text.
Be as concise as possible.
```

Figure A.1.1: The system prompt given to all models except Snakmodel-7b-it in the open domain setting. Models that don't accept a system prompt such as gemma-9b-it recieved it prepended to the user prompt for consistency.

System Prompt (Multiple Choice)

```
You are a helpful assistant. You respond to questions in
Danish.
Respond briefly and accurately.
Do not generate any extra questions or superfluous text.
Be as concise as possible.
The context may or may not be relevant.
```

Figure A.1.2: The system prompt given to all models except Snakmodel-7b-it in the open domain setting. Models that don't accept a system prompt such as gemma-9b-it recieved it prepended to the user prompt for consistency.

Snakmodel System Prompt

```
Du er Snakmodel, skabt af IT-Universitetet i København.
Du er en hjælpsom assistent.

(You are Snakmodel, created by the IT-University in
Copenhagen. You are a helpful assistant.)
```

Figure A.1.3: The system prompt given to SnakModel-7b-instruct. Translation in parantheses are not part of the prompt

Open-domain QA Prompt

```
Besvar følgende spørgsmål ud fra kontekst:
Kontekst: ...
Spørgsmål: ...

(Answer the following question based on the context:
 Context: ...
 Question: ...)
```

Figure A.1.4: The prompt given to models when answering open domain questions. Translation in parantheses are not part of the prompt

Multiple-choice QA Prompt

```
Givet konteksten, svar kun med bogstavet for den rigtige
mulighed.
#KONTEKST
...
#SPØRGSMÅL
...
#SVARMULIGHEDER
A: ...
B: ...
C: ...
#SVAR
Svaret er mulighed

(Given the context, answer only with the letter for the
right option.
#CONTEXT
...
#QUESTION
...
#OPTIONS
A: ...
B: ...
C: ...
#ANSWER
The answer is)
```

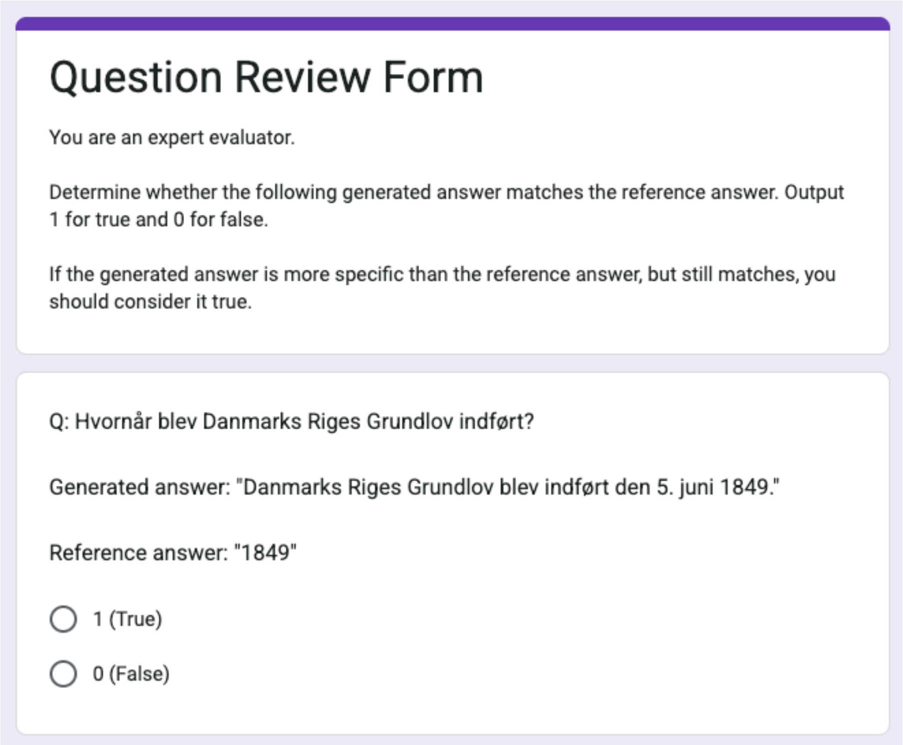
Figure A.1.5: The prompt used for answering questions in the multiple-choice setting. Inspired by the prompts used by (Holm et al., 2025), but expanded to include retrieved context. Translation in parantheses are not part of the prompt

Evaluation Prompt

```
You are an expert evaluator. Determine whether the
following generated answer matches the reference answer.
Output 1 for true and 0 for false.
If the generated answer is more specific than the
reference answer, but still matches, you should consider
it true.
Question: ...
Generated Answer: ...
Reference Answer: ...
Rate the quality (0 or 1) and briefly explain your
reasoning.
```

Figure A.1.6: The prompt used for LLM-as-a-judge evaluation of open-domain answers

A.2 Question Review Form



Question Review Form

You are an expert evaluator.

Determine whether the following generated answer matches the reference answer. Output 1 for true and 0 for false.

If the generated answer is more specific than the reference answer, but still matches, you should consider it true.

Q: Hvornår blev Danmarks Riges Grundlov indført?

Generated answer: "Danmarks Riges Grundlov blev indført den 5. juni 1849."

Reference answer: "1849"

○ 1 (True)

○ 0 (False)

Figure A.2.1: Questions as they appear to human annotators during evaluation

Bibliography

AI, ., :, Young, A., Chen, B., Li, C., Huang, C., Zhang, G., Zhang, G., Wang, G., Li, H., Zhu, J., Chen, J., Chang, J., Yu, K., Liu, P., Liu, Q., Yue, S., Yang, S., Yang, S., Xie, W., Huang, W., Hu, X., Ren, X., Niu, X., Nie, P., Li, Y., Xu, Y., Liu, Y., Wang, Y., Cai, Y., Gu, Z., Liu, Z., and Dai, Z. (2025). Yi: Open foundation models by 01.ai. *arXiv preprint arXiv:2403.04652*.

Attardi, G. (2015). Wikiextractor. `https://github.com/attardi/wikiextractor`.

BehnamGhader, P., Adlakha, V., Mosbach, M., Bahdanau, D., Chapados, N., and Reddy, S. (2024). Llm2vec: Large language models are secretly powerful text encoders.

Bengio, Y., Ducharme, R., Vincent, P., and Janvin, C. (2003). A neural probabilistic language model. *J. Mach. Learn. Res.*, 3(null):1137–1155.

Brown, D. (2019). rank_bm25: A fast implementation of the bm25 ranking function. `https://github.com/dorianbrown/rank_bm25`. Accessed: 2025-06-01.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. (2020). Language models are few-shot learners. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H., editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Carrino, C. P., Costa-jussà, M. R., and Fonollosa, J. A. R. (2019). Automatic spanish translation of the squad dataset for multilingual question answering. *CoRR*, abs/1912.05200.

Cun, Y. L., Boser, B., Denker, J. S., Howard, R. E., Habbard, W., Jackel, L. D., and Henderson, D. (1990). *Handwritten digit recognition with a back-propagation network*, page 396–404. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Devine, P. (2024). Tagengo: A multilingual chat dataset. *arXiv preprint arXiv:2405.12612*.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In Burstein, J., Doran, C., and Solorio, T., editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Dinzinger, M., Caspari, L., Dastidar, K. G., Mitrović, J., and Granitzer, M. (2025). Webfaq: A multilingual collection of natural qa datasets for dense retrieval.

Doddapaneni, S., Ramesh, G., Khapra, M., Kunchukuttan, A., and Kumar, P. (2025). A primer on pretrained multilingual language models. *ACM Comput. Surv.*, 57(9).

Douze, M., Guzhva, A., Deng, C., Johnson, J., Szilvasy, G., Mazaré, P.-E., Lomeli, M., Hosseini, L., and Jégou, H. (2025). The faiss library.

Enevoldsen, K., Kardos, M., Muennighoff, N., and Nielbo, K. L. (2024). The scandinavian embedding benchmarks: Comprehensive assessment of multilingual and monolingual text embedding. In Globerson, A., Mackey, L., Belgrave, D., Fan, A., Paquet, U., Tomczak, J., and Zhang, C., editors, *Advances in Neural Information Processing Systems*, volume 37, pages 40336–40358. Curran Associates, Inc.

Es, S., James, J., Espinosa Anke, L., and Schockaert, S. (2024). RAGAs: Automated evaluation of retrieval augmented generation. In Aletras, N. and De Clercq, O., editors, *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 150–158, St. Julians, Malta. Association for Computational Linguistics.

Fleuret, F. (2023). *The Little Book of Deep Learning*. lulu.com.

Gheini, M., Ren, X., and May, J. (2021). On the strengths of cross-attention in pretrained transformers for machine translation. *CoRR*, abs/2104.08771.

Grattafiori, A., Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Vaughan, A., Yang, A., Fan, A., Goyal, A., Hartshorn, A., Yang, A., Mitra, A., Sravankumar, A., Korenev, A., Hinsvark, A., Rao, A., Zhang, A., Rodriguez, A., Gregerson, A., Spataru, A., Roziere, B., Biron, B., Tang, B., Chern, B., Caucheteux, C., Nayak, C., Bi, C., Marra, C., McConnell, C., Keller, C., Touret, C., Wu, C., Wong, C., Ferrer, C. C., Nikolaidis, C., Allonsius, D., Song, D., Pintz, D., Livshits, D., Wyatt, D., Esiobu, D., Choudhary, D., Mahajan, D., Garcia-Olano, D., Perino, D., Hupkes, D., Lakomkin, E., AlBadawy, E., Lobanova, E., Dinan, E., Smith, E. M., Radenovic, F., Guzmán, F., Zhang, F., Synnaeve, G., Lee, G., Anderson, G. L., Thattai, G., Nail, G., Mialon, G., Pang, G., Cucurell, G., Nguyen, H., Korevaar, H., Xu, H., Touvron, H., Zarov, I., Ibarra, I. A., Kloumann, I., Misra, I., Evtimov, I., Zhang, J., Copet, J., Lee, J., Geffert, J., Vranes, J., Park, J., Mahadeokar, J., Shah, J., van der Linde, J., Billock, J., Hong, J., Lee, J., Fu, J., Chi, J., Huang, J., Liu, J., Wang, J., Yu, J., Bitton, J., Spisak, J., Park, J., Rocca, J., Johnstun, J., Saxe, J., Jia, J., Alwala, K. V., Prasad, K., Upasani, K., Plawiak, K., Li, K., Heafield, K., Stone, K., El-Arini, K., Iyer, K., Malik, K., Chiu, K., Bhalla, K., Lakhotia, K., Rantala-Yeary, L., van der Maaten, L., Chen, L., Tan, L., Jenkins, L., Martin, L., Madaan, L., Malo, L., Blecher, L., Landzaat, L., de Oliveira, L., Muzzi, M., Pasupuleti, M., Singh, M., Paluri, M., Kardas, M., Tsimpoukelli, M., Oldham, M., Rita, M., Pavlova, M., Kambadur, M., Lewis, M., Si, M., Singh, M. K., Hassan, M., Goyal, N., Torabi, N., Bashlykov, N., Bogoychev, N., Chatterji, N., Zhang, N., Duchenne, O., Çelebi, O., Alrassy, P., Zhang, P., Li, P., Vasic, P., Weng, P., Bhargava, P., Dubal, P., Krishnan, P., Koura, P. S., Xu, P., He, Q., Dong, Q., Srinivasan, R., Ganapathy, R., Calderer, R., Cabral, R. S., Stojnic, R., Raileanu, R., Maheswari, R., Girdhar, R., Patel, R., Sauvestre, R., Polidoro, R., Sumbaly, R., Taylor, R., Silva, R., Hou, R., Wang, R., Hosseini, S., Chennabasappa, S., Singh, S., Bell, S., Kim, S. S., Edunov, S., Nie, S., Narang, S., Raparthy, S., Shen, S., Wan, S., Bhosale, S., Zhang, S., Vandenhende, S., Batra, S., Whitman, S., Sootla, S., Collot, S., Gururangan, S., Borodinsky, S., Herman, T., Fowler, T., Sheasha, T., Georgiou, T., Scialom, T., Speckbacher, T., Mihaylov, T., Xiao, T., Karn, U., Goswami, V., Gupta, V., Ramanathan, V., Kerkez, V., Gonguet, V., Do, V., Vogeti, V., Albiero, V., Petrovic, V., Chu, W., Xiong, W., Fu, W., Meers, W., Martinet, X., Wang, X., Wang, X., Tan, X. E., Xia, X., Xie, X., Jia, X., Wang, X., Goldschlag, Y., Gaur, Y., Babaei, Y., Wen, Y., Song, Y., Zhang, Y., Li, Y., Mao, Y., Coudert, Z. D., Yan, Z., Chen, Z., Papakipos, Z., Singh, A., Srivastava, A., Jain, A., Kelsey, A., Shajnfeld, A., Gangidi, A., Victoria, A., Goldstand, A., Menon, A., Sharma, A., Boesenberg, A., Baevski, A., Feinstein, A., Kallet, A., Sangani, A., Teo, A., Yunus, A., Lupu, A., Alvarado, A., Caples, A., Gu, A., Ho, A., Poulton, A., Ryan, A., Ramchandani, A., Dong, A., Franco, A., Goyal, A., Saraf, A., Chowdhury, A., Gabriel, A., Bharambe, A., Eisenman, A., Yazdan, A., James, B., Maurer, B., Leonhardi, B., Huang, B., Loyd, B., Paola, B. D., Paranjape, B., Liu, B., Wu, B., Ni, B., Hancock, B., Wasti, B., Spence, B., Stojkovic, B., Gamido, B., Montalvo, B., Parker, C., Burton, C., Mejia, C., Liu, C., Wang, C., Kim, C., Zhou, C., Hu, C., Chu, C.-H., Cai, C., Tindal, C., Feichtenhofer, C., Gao, C., Civin, D., Beaty, D., Kreymer, D., Li, D., Adkins, D., Xu, D., Testuggine, D., David, D., Parikh, D., Liskovich, D., Foss, D., Wang, D., Le, D., Holland, D., Dowling, E., Jamil, E., Montgomery, E., Presani, E., Hahn, E., Wood, E., Le, E.-T., Brinkman, E., Arcaute, E., Dunbar, E., Smothers, E., Sun, F., Kreuk, F., Tian, F., Kokkinos, F., Ozgenel, F., Caggioni, F., Kanayet, F., Seide, F., Florez, G. M., Schwarz, G., Badeer, G., Swee, G., Halpern, G., Herman, G., Sizov, G., Guangyi, Zhang, Lakshminarayanan, G., Inan, H., Shojanazeri, H., Zou, H., Wang, H., Zha, H., Habeeb, H., Rudolph, H., Suk, H., Aspegren, H., Goldman, H., Zhan, H., Damlaj, I., Molybog, I., Tufanov, I., Leontiadis, I., Veliche, I.-E., Gat, I., Weissman, J., Geboski, J., Kohli, J., Lam, J., Asher, J., Gaya, J.-B., Marcus, J., Tang, J., Chan, J., Zhen, J., Reizenstein, J., Teboul, J., Zhong, J., Jin, J., Yang, J., Cummings, J., Carvill, J., Shepard, J., McPhie, J., Torres, J., Ginsburg, J., Wang, J., Wu, K., U, K. H., Saxena, K., Khandelwal, K., Zand, K., Matosich, K., Veeraraghavan, K., Michelena, K., Li, K., Jagadeesh, K., Huang, K., Chawla, K., Huang, K., Chen, L., Garg, L., A, L., Silva, L., Bell, L., Zhang, L., Guo, L., Yu, L., Moshkovich, L., Wehrstedt, L., Khabsa, M., Avalani, M., Bhatt, M., Mankus, M., Hasson, M., Lennie, M., Reso, M., Groshev, M., Naumov, M., Lathi, M., Keneally, M., Liu, M., Seltzer, M. L., Valko, M., Restrepo, M., Patel, M., Vyatskov, M., Samvelyan, M., Clark, M., Macey, M., Wang, M., Hermoso, M. J., Metanat, M., Rastegari, M., Bansal, M., Santhanam, N., Parks, N., White, N., Bawa, N., Singhal, N., Egebo, N., Usunier, N., Mehta, N., Laptev, N. P., Dong, N., Cheng, N., Chernoguz, O., Hart, O., Salpekar, O., Kalinli, O., Kent, P., Parekh, P., Saab, P., Balaji, P., Rittner, P., Bontrager, P., Roux, P., Dollar, P., Zvyagina, P., Ratanchandani, P., Yuvraj, P., Liang, Q., Alao, R., Rodriguez, R., Ayub, R., Murthy, R., Nayani, R., Mitra, R., Parthasarathy, R., Li, R., Hogan, R., Battey, R., Wang, R., Howes, R., Rinott, R., Mehta, S., Siby, S., Bondu, S. J., Datta, S., Chugh, S., Hunt, S., Dhillon, S., Sidorov, S., Pan, S., Mahajan, S., Verma, S., Yamamoto, S., Ramaswamy, S., Lindsay, S., Lindsay, S., Feng, S., Lin, S., Zha, S. C., Patil, S., Shankar, S., Zhang, S., Zhang, S., Wang, S., Agarwal, S., Sajuyigbe, S., Chintala, S., Max, S., Chen, S., Kehoe, S., Satterfield, S., Govindaprasad, S., Gupta, S., Deng, S., Cho, S., Virk, S., Subramanian, S., Choudhury, S., Goldman, S., Remez, T., Glaser, T., Best, T., Koehler, T., Robinson, T., Li, T., Zhang, T., Matthews, T., Chou, T., Shaked, T., Vontimitta, V., Ajayi, V., Montanez, V., Mohan, V., Kumar, V. S., Mangla, V., Ionescu, V., Poenaru, V., Mihailescu, V. T., Ivanov, V., Li, W., Wang, W., Jiang, W., Bouaziz, W., Constable, W., Tang, X., Wu, X., Wang, X., Wu, X., Gao, X., Kleinman, Y., Chen, Y., Hu, Y., Jia, Y., Qi, Y., Li, Y., Zhang, Y., Zhang, Y., Adi, Y., Nam, Y., Yu, Wang, Zhao, Y., Hao, Y., Qian, Y., Li, Y., He, Y., Rait, Z., DeVito, Z., Rosnbrick, Z., Wen, Z., Yang, Z., Zhao, Z., and Ma, Z. (2024). The llama 3 herd of models.

Harman, D. (1993). Overview of the first trec conference. In *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '93, page 36–47, New York, NY, USA. Association for Computing Machinery.

Hinton, G., Vinyals, O., and Dean, J. (2015). Distilling the knowledge in a neural network.

Holm, S. V. (2024). Are gllms danoliterate? benchmarking generative nlp in danish. Master's thesis, Technical University of Denmark. Available at https://sorenmulli.github.io/thesis/thesis.pdf.

Holm, S. V., Hansen, L. K., and Nielsen, M. C. (2025). Danoliteracy of generative large language models.

Huggingface (2024). Nous hermes 2 mistral 7b dpo - huggingface datacard. https://huggingface.co/NousResearch/Nous-Hermes-2-Mistral-7B-DPO).

Hvingelby, R., Pauli, A. B., Barrett, M., Rosted, C., Lidegaard, L. M., and Søgaard, A. (2020). DaNE: A named entity resource for Danish. In Calzolari, N., Béchet, F., Blache, P., Choukri, K., Cieri, C., Declerck, T., Goggi, S., Isahara, H., Maegaard, B., Mariani, J., Mazo, H., Moreno, A., Odijk, J., and Piperidis, S., editors, *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4597–4604, Marseille, France. European Language Resources Association.

Izacard, G., Caron, M., Hosseini, L., Riedel, S., Bojanowski, P., Joulin, A., and Grave, E. (2021). Towards unsupervised dense information retrieval with contrastive learning. *CoRR*, abs/2112.09118.

Izacard, G. and Grave, E. (2020). Distilling knowledge from reader to retriever for question answering. *CoRR*, abs/2012.04584.

Izacard, G. and Grave, E. (2021). Leveraging passage retrieval with generative models for open domain question answering. In Merlo, P., Tiedemann, J., and Tsarfaty, R., editors, *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 874–880, Online. Association for Computational Linguistics.

Jensen, K. J. and Riis, S. (2000). Self-organizing letter code-book for text-to-phoneme neural network model. In *6th International Conference on Spoken Language Processing (ICSLP 2000)*, pages vol. 3, 318–321.

Jiang, A. Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D. S., de las Casas, D., Bressand, F., Lengyel, G., Lample, G., Saulnier, L., Lavaud, L. R., Lachaux, M.-A., Stock, P., Scao, T. L., Lavril, T., Wang, T., Lacroix, T., and Sayed, W. E. (2023). Mistral 7b.

Joshi, M., Choi, E., Weld, D., and Zettlemoyer, L. (2017). TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In Barzilay, R. and

Kan, M.-Y., editors, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada. Association for Computational Linguistics.

Joshi, P., Santy, S., Budhiraja, A., Bali, K., and Choudhury, M. (2020). The state and fate of linguistic diversity and inclusion in the NLP world. In Jurafsky, D., Chai, J., Schluter, N., and Tetreault, J., editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293, Online. Association for Computational Linguistics.

Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., and Amodei, D. (2020). Scaling laws for neural language models. *CoRR*, abs/2001.08361.

Karpukhin, V., Oguz, B., Min, S., Wu, L., Edunov, S., Chen, D., and Yih, W. (2020). Dense passage retrieval for open-domain question answering. *CoRR*, abs/2004.04906.

Khattab, O. and Zaharia, M. (2020). Colbert: Efficient and effective passage search via contextualized late interaction over BERT. *CoRR*, abs/2004.12832.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, volume 25, pages 1097–1105. Curran Associates, Inc.

Kwiatkowski, T., Palomaki, J., Redfield, O., Collins, M., Parikh, A., Alberti, C., Epstein, D., Polosukhin, I., Devlin, J., Lee, K., Toutanova, K., Jones, L., Kelcey, M., Chang, M.-W., Dai, A. M., Uszkoreit, J., Le, Q., and Petrov, S. (2019). Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

Lars Løkke Rasmussen (2009). Bekendtgørelse af tronfølgeloven). `https://www.retsinformation.dk/eli/lta/2009/847`. LBK nr 847 af 04/09/2009.

Lee, K., Chang, M., and Toutanova, K. (2019). Latent retrieval for weakly supervised open domain question answering. *CoRR*, abs/1906.00300.

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., and Zettlemoyer, L. (2019). BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *CoRR*, abs/1910.13461.

Lewis, P. S. H., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., Rocktäschel, T., Riedel, S., and Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. *CoRR*, abs/2005.11401.

Li, H. (2022). Language models: past, present, and future. *Commun. ACM*, 65(7):56–63.

Li, J., Zhang, Q., Yu, Y., Fu, Q., and Ye, D. (2024). More agents is all you need. *arXiv preprint arXiv:2402.05120.*

Lin, C.-Y. (2004). ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Liu, P. J., Saleh, M., Pot, E., Goodrich, R., Sepassi, R., Kaiser, L., and Shazeer, N. (2018). Generating wikipedia by summarizing long sequences. In *International Conference on Learning Representations (ICLR)*.

Liu, Y., Shi, K., Fabbri, A., Zhao, Y., Wang, P., Wu, C.-S., Joty, S., and Cohan, A. (2025). ReIFE: Re-evaluating instruction-following evaluation. In Chiruzzo, L., Ritter, A., and Wang, L., editors, *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 12247–12287, Albuquerque, New Mexico. Association for Computational Linguistics.

Longpre, S., Lu, Y., and Daiber, J. (2021). MKQA: A linguistically diverse benchmark for multilingual open domain question answering. *Transactions of the Association for Computational Linguistics*, 9:1389–1406.

Markov, A. A. (2006). An example of statistical investigation of the text eugene onegin concerning the connection of samples in chains. *Science in Context*, 19(4):591–600. Translated by Gloria Custance and David Link.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., and Dean, J. (2013a). Distributed representations of words and phrases and their compositionality.

Mikolov, T., Yih, W.-t., and Zweig, G. (2013b). Linguistic regularities in continuous space word representations. In Vanderwende, L., Daumé III, H., and Kirchhoff, K., editors, *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 746–751, Atlanta, Georgia. Association for Computational Linguistics.

Muennighoff, N., Tazi, N., Magne, L., and Reimers, N. (2023). Mteb: Massive text embedding benchmark.

Nasdaq, Inc. (2025). NVIDIA Corporation Common Stock (NVDA) Advanced Charting. `https://www.nasdaq.com/market-activity/stocks/nvda/advanced-charting`. Accessed: 2025-06-02.

Nema, P. and Khapra, M. M. (2018). Towards a better metric for evaluating question generation systems. In Riloff, E., Chiang, D., Hockenmaier, J., and Tsujii, J., editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3950–3959, Brussels, Belgium. Association for Computational Linguistics.

Nielsen, D. (2023). ScandEval: A benchmark for Scandinavian natural language processing. In Alumäe, T. and Fishel, M., editors, *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 185–201, Tórshavn, Faroe Islands. University of Tartu Library.

Nørregaard, J. and Derczynski, L. (2021). DanFEVER: claim verification dataset for Danish. In Dobnik, S. and Øvrelid, L., editors, *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 422–428, Reykjavik, Iceland (Online). Linköping University Electronic Press, Sweden.

Osborne, C., Ding, J., and Kirk, H. (2024). The ai community building the future? a quantitative analysis of development activity on hugging face hub. *Journal of Computational Social Science*, 7:2067–2105.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., Schulman, J., Hilton, J., Kelton, F., Miller, L., Simens, M., Askell, A., Welinder, P., Christiano, P. F., Leike, J., and Lowe, R. (2022). Training language models to follow instructions with human feedback. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A., editors, *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744. Curran Associates, Inc.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In Isabelle, P., Charniak, E., and Lin, D., editors, *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Pedersen, J., Laursen, M., Vinholt, P., and Savarimuthu, T. R. (2023). MeDa-BERT: A medical Danish pretrained transformer model. In Alumäe, T. and Fishel, M., editors, *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 301–307, Tórshavn, Faroe Islands. University of Tartu Library.

Pennington, J., Socher, R., and Manning, C. (2014). GloVe: Global vectors for word representation. In Moschitti, A., Pang, B., and Daelemans, W., editors, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep contextualized word representations. In Walker, M., Ji, H., and Stent, A., editors, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Petroni, F., Rocktäschel, T., Lewis, P. S. H., Bakhtin, A., Wu, Y., Miller, A. H., and Riedel, S. (2019). Language models as knowledge bases? *CoRR*, abs/1909.01066.

Picard, D. (2021). Torch.manual_seed(3407) is all you need: On the influence of random seeds in deep learning architectures for computer vision. *CoRR*, abs/2109.08203.

Pineau, E. and Jabkhiro, J. (2025). France's le pen convicted of graft, barred from running for president in 2027. *Reuters*.

Radford, A. and Narasimhan, K. (2018). Improving language understanding by generative pre-training.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. (2019). Language models are unsupervised multitask learners.

Rafailov, R., Sharma, A., Mitchell, E., Ermon, S., Manning, C. D., and Finn, C. (2023). Direct preference optimization: your language model is secretly a reward model. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. (2016). SQuAD: 100,000+ questions for machine comprehension of text. In Su, J., Duh, K., and Carreras, X., editors, *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Reimers, N. and Gurevych, I. (2019). Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In Inui, K., Jiang, J., Ng, V., and Wan, X., editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Reuters Staff (2025). Trump to announce private sector ai infrastructure investment: Cbs reports.

Reves, O. (2025). Thesis. `https://github.com/OscarReves/thesis`. Accessed: 2025-06-02.

Richardson, L. (2023). Beautiful soup 4. `https://pypi.org/project/beautifulsoup4/`. Version 4.x, accessed 2025-06-01.

Robertson, S. E., Walker, S., Jones, S., Hancock-Beaulieu, M., and Gatford, M. (1994). Okapi at trec-3. In Harman, D. K., editor, *TREC*, volume 500-225 of *NIST Special Publication*, pages 109–126. National Institute of Standards and Technology (NIST).

Röttger, P., Hofmann, V., Pyatkin, V., Hinck, M., Kirk, H., Schuetze, H., and Hovy, D. (2024). Political compass or spinning arrow? towards more meaningful evaluations for values and opinions in large language models. In Ku, L.-W., Martins, A., and Srikumar, V., editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15295–15311, Bangkok, Thailand. Association for Computational Linguistics.

Rumelhart, D. E. and Abrahamson, A. A. (1973). A model for analogical reasoning. *Cognitive Psychology*, 5(1):1–28.

Sellam, T., Das, D., and Parikh, A. (2020). BLEURT: Learning robust metrics for text generation. In Jurafsky, D., Chai, J., Schluter, N., and Tetreault, J., editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.

Shang, G., Abdine, H., Khoubrane, Y., Mohamed, A., Abbahaddou, Y., Ennadir, S., Momayiz, I., Ren, X., Moulines, E., Nakov, P., Vazirgiannis, M., and Xing, E. (2025). Atlas-chat: Adapting large language models for low-resource Moroccan Arabic dialect. In Hettiarachchi, H., Ranasinghe, T., Rayson, P., Mitkov, R., Gaber, M., Premasiri, D., Tan, F. A., and Uyangodage, L., editors, *Proceedings of the First Workshop on Language Models for Low-Resource Languages*, pages 9–30, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Singh, H., West, R., and Colavizza, G. (2021). Wikipedia citations: A comprehensive data set of citations with identifiers extracted from english wikipedia. *Quantitative Science Studies*, 2(1):1–19.

Sparck Jones, K. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1):11–21.

Stine, D. D. (2008). The manhattan project, the apollo program, and federal energy technology r & d programs: A comparative analysis. Congressional Research Service, the Library of Congress.

Su, H., Shi, W., Kasai, J., Wang, Y., Hu, Y., Ostendorf, M., Yih, W.-t., Smith, N. A., Zettlemoyer, L., and Yu, T. (2023). One embedder, any task: Instruction-finetuned text embeddings. In Rogers, A., Boyd-Graber, J., and Okazaki, N., editors, *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1102–1121, Toronto, Canada. Association for Computational Linguistics.

Søren Vejlgaard Holm (2024). Citizenship test. https://huggingface.co/datasets/sorenmulli/citizenship-test-da.

Team, G., Mesnard, T., Hardin, C., Dadashi, R., Bhupatiraju, S., Pathak, S., Sifre, L., Rivière, M., Kale, M. S., Love, J., Tafti, P., Hussenot, L., Sessa, P. G., Chowdhery, A., Roberts, A., Barua, A., Botev, A., Castro-Ros, A., Slone, A., Héliou, A., Tacchetti, A., Bulanova, A., Paterson, A., Tsai, B., Shahriari, B., Lan, C. L., Choquette-Choo, C. A., Crepy, C., Cer, D., Ippolito, D., Reid, D., Buchatskaya, E., Ni, E., Noland, E., Yan, G., Tucker, G., Muraru, G.-C., Rozhdestvenskiy, G., Michalewski, H., Tenney, I., Grishchenko, I., Austin, J., Keeling, J., Labanowski, J., Lespiau, J.-B., Stanway, J., Brennan, J., Chen, J., Ferret, J., Chiu, J., Mao-Jones, J., Lee, K., Yu, K., Millican, K., Sjoesund, L. L., Lee, L., Dixon, L., Reid, M., Mikuła, M., Wirth, M., Sharman, M., Chinaev, N., Thain, N., Bachem, O., Chang, O., Wahltinez, O., Bailey, P., Michel, P., Yotov, P., Chaabouni, R., Comanescu, R., Jana, R., Anil, R., McIlroy, R., Liu, R., Mullins, R., Smith, S. L., Borgeaud, S., Girgin, S., Douglas, S., Pandya, S., Shakeri, S., De, S., Klimenko, T., Hennigan, T., Feinberg, V., Stokowiec, W., hui Chen, Y., Ahmed, Z., Gong, Z., Warkentin, T., Peran, L., Giang, M., Farabet, C., Vinyals, O., Dean, J., Kavukcuoglu, K., Hassabis, D., Ghahramani, Z., Eck, D., Barral, J., Pereira, F., Collins, E., Joulin, A., Fiedel, N., Senter, E., Andreev, A., and Kenealy, K. (2024a). Gemma: Open models based on gemini research and technology.

Team, G., Riviere, M., Pathak, S., Sessa, P. G., Hardin, C., Bhupatiraju, S., Hussenot, L., Mesnard, T., Shahriari, B., Ramé, A., Ferret, J., Liu, P., Tafti, P., Friesen, A., Casbon, M., Ramos, S., Kumar, R., Lan, C. L., Jerome, S., Tsitsulin, A., Vieillard, N., Stanczyk, P., Girgin, S., Momchev, N., Hoffman, M., Thakoor, S., Grill, J.-B., Neyshabur, B., Bachem, O., Walton, A., Severyn, A., Parrish, A., Ahmad, A., Hutchison, A., Abdagic, A., Carl, A., Shen, A., Brock, A., Coenen, A., Laforge, A., Paterson, A., Bastian, B., Piot, B., Wu, B., Royal, B., Chen, C., Kumar, C., Perry, C., Welty, C., Choquette-Choo, C. A., Sinopalnikov, D., Weinberger, D., Vijaykumar, D., Rogozińska, D., Herbison, D., Bandy, E., Wang, E., Noland, E., Moreira, E., Senter, E., Eltyshev, E., Visin, F., Rasskin, G., Wei, G., Cameron, G., Martins, G., Hashemi, H., Klimczak-Plucińska, H., Batra, H., Dhand, H., Nardini, I., Mein, J., Zhou, J., Svensson, J., Stanway, J., Chan, J., Zhou, J. P., Carrasqueira, J., Iljazi, J., Becker, J., Fernandez, J., van Amersfoort, J., Gordon, J., Lipschultz, J., Newlan, J., yeong Ji, J., Mohamed, K., Badola, K., Black, K., Millican, K., McDonell, K., Nguyen, K., Sodhia, K., Greene, K., Sjoesund, L. L., Usui, L., Sifre, L., Heuermann, L., Lago, L., McNealus, L., Soares, L. B., Kilpatrick, L., Dixon, L., Martins, L., Reid, M., Singh, M., Iverson, M., Görner, M., Velloso, M., Wirth, M., Davidow, M., Miller, M., Rahtz, M., Watson, M., Risdal, M., Kazemi, M., Moynihan, M., Zhang, M., Kahng, M., Park, M., Rahman, M., Khatwani, M., Dao, N., Bardoliwalla, N., Devanathan, N., Dumai, N., Chauhan, N., Wahltinez, O., Botarda, P., Barnes, P., Barham, P., Michel, P., Jin, P., Georgiev, P., Culliton, P., Kuppala, P., Comanescu, R., Merhej, R., Jana, R., Rokni, R. A., Agarwal, R., Mullins, R., Saadat, S., Carthy, S. M., Cogan, S., Perrin, S., Arnold, S. M. R., Krause, S., Dai, S., Garg, S., Sheth, S., Ronstrom, S., Chan, S., Jordan, T., Yu, T., Eccles, T., Hennigan, T., Kocisky, T., Doshi, T., Jain, V., Yadav, V., Meshram, V., Dharmadhikari, V., Barkley, W., Wei, W., Ye, W., Han, W., Kwon, W., Xu, X., Shen, Z., Gong, Z., Wei, Z., Cotruta, V., Kirk, P., Rao, A., Giang, M., Peran, L., Warkentin, T., Collins, E., Barral, J., Ghahramani, Z., Hadsell, R., Sculley, D., Banks, J., Dragan, A., Petrov, S., Vinyals, O., Dean, J., Hassabis, D., Kavukcuoglu, K., Farabet, C., Buchatskaya, E., Borgeaud, S., Fiedel, N., Joulin, A., Kenealy, K., Dadashi, R., and Andreev, A. (2024b). Gemma 2: Improving open language models at a practical size.

Thakur, N., Reimers, N., Rücklé, A., Srivastava, A., and Gurevych, I. (2021). Beir: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In Vanschoren, J. and Yeung, S., editors, *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1.

Thorne, J., Vlachos, A., Christodoulopoulos, C., and Mittal, A. (2018). FEVER: a large-scale dataset for fact extraction and VERification. In Walker, M., Ji, H., and Stent, A., editors, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 809–819, New Orleans, Louisiana. Association for Computational Linguistics.

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., Rodriguez, A., Joulin, A., Grave, E., and Lample, G. (2023). Llama: Open and efficient foundation language models.

Turing, A. M. (1950). Computing machinery and intelligence. *Mind*, LIX(236):433–460.

van den Oord, A., Li, Y., and Vinyals, O. (2018). Representation learning with contrastive predictive coding. *CoRR*, abs/1807.03748.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. *CoRR*, abs/1706.03762.

Wang, A., Singh, A., Michael, J., Hill, F., Levy, O., and Bowman, S. (2018). GLUE: A multi-task benchmark and analysis platform for natural language understanding. In Linzen, T., Chrupała, G., and Alishahi, A., editors, *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Wang, L., Yang, N., Huang, X., Jiao, B., Yang, L., Jiang, D., Majumder, R., and Wei, F. (2024a). Text embeddings by weakly-supervised contrastive pre-training.

Wang, L., Yang, N., Huang, X., Yang, L., Majumder, R., and Wei, F. (2024b). Improving text embeddings with large language models. In Ku, L.-W., Martins, A., and Srikumar, V., editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11897–11916, Bangkok, Thailand. Association for Computational Linguistics.

Wang, L., Yang, N., Huang, X., Yang, L., Majumder, R., and Wei, F. (2024c). Multilingual e5 text embeddings: A technical report.

Wikipedia (2025a). Fagforening — Wikipedia, the free encyclopedia. `http://da.wikipedia.org/w/index.php?title=Fagforening&oldid=11945929`. [Online; accessed 31-May-2025].

Wikipedia (2025b). Kommuner i Danmark — Wikipedia, the free encyclopedia. `http://da.wikipedia.org/w/index.php?title=Kommuner%20i%20Danmark&oldid=12013829`. [Online; accessed 31-May-2025].

Wikipedia (2025c). Kroning — Wikipedia, the free encyclopedia. `http://da.wikipedia.org/w/index.php?title=Kroning&oldid=11705698`. [Online; accessed 31-May-2025].

Wikipedia (2025d). Trade union — Wikipedia, the free encyclopedia. `http://en.wikipedia.org/w/index.php?title=Trade%20union&oldid=1292187805`. [Online; accessed 31-May-2025].

Wu, Y., Schuster, M., Chen, Z., Le, Q. V., Norouzi, M., Macherey, W., Krikun, M., Cao, Y., Gao, Q., Macherey, K., Klingner, J., Shah, A., Johnson, M., Liu, X., Łukasz Kaiser, Gouws, S., Kato, Y., Kudo, T., Kazawa, H., Stevens, K., Kurian, G., Patil, N., Wang, W., Young, C., Smith, J., Riesa, J., Rudnick, A., Vinyals, O., Corrado, G., Hughes, M., and Dean, J. (2016). Google's neural machine translation system: Bridging the gap between human and machine translation.

Zhang, M., Müller-Eberstein, M., Bassignana, E., and Goot, R. v. d. (2025). SnakModel: Lessons learned from training an open Danish large language model. In Johansson, R. and Stymne, S., editors, *Proceedings of the Joint 25th Nordic Conference on Computational Linguistics and 11th Baltic Conference on Human Language Technologies (NoDaLiDa/Baltic-HLT 2025)*, pages 812–825, Tallinn, Estonia. University of Tartu Library.

Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., and Artzi, Y. (2020). Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations (ICLR)*.

Zheng, L., Chiang, W.-L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E. P., Zhang, H., Gonzalez, J. E., and Stoica, I. (2023). Judging llm-as-a-judge with mt-bench and chatbot arena. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.