Beyond Perplexity: Examining Temporal Generalization of Large Language Models via Definition Generation

MSc Thesis *(Afstudeerscriptie)*

written by

Iris Luden

under the supervision of **Prof Dr Raquel Fernández**, and submitted to the Examinations Board in partial fulfillment of the requirements for the degree of

MSc in Logic

at the *Universiteit van Amsterdam.*

Date of the public defense:	**Members of the Thesis Committee:**
July 10, 2023	Dr Malvin Gattinger (chair)
	Prof Dr Raquel Fernández (supervisor)
	Dr Jelke Bloem
	Dr Pia Sommerauer

Institute for Logic, Language and Computation

Abstract

The emergence of large language models (LLMs) has significantly improved performance across various Natural Language Processing (NLP) tasks. However, the field of NLP predominantly follows a static language modeling paradigm, resulting in performance deterioration of LLMs over time. This indicates a lack of temporal generalization, i.e., the ability to adjust their capabilities to data beyond their training period. In real-life NLP applications, models are often pre-trained on data from one time period and then deployed for tasks which inherently involve temporally shifted data. So far, performance deterioration of LLMs is primarily attributed to the factual changes over time, leading to attempts of updating a LLMs factual knowledge to avoid performance deterioration. However, not only the facts of the world, but also the language we use to describe it constantly changes. Recent studies have indicated a relationship between performance deterioration and semantic change. Performance deterioration is typically measured using perplexity scores and relative performance on downstream tasks. But such dry comparisons of perplexity and accuracy do not explain the effects of temporally shifted data on LLMs in practice. Given the potential societal impact of NLP applications, it is crucial gain insight into how the performance deterioration, particularly caused by semantic change, is reflected in the output of LLMs. This thesis investigates how semantic change in temporally shifted data impacts the performance of a LLM on the downstream task of contextualized word definition generation. This approach offers a dual perspective: quantitative measurement of performance deterioration, as well as human-interpretable output through the generated definitions. First, I construct two diachronic corpora of Twitter and Reddit data, such that one overlaps in time with the pre-training period, and the other is temporally shifted. Next, I use a lexical semantic change system to collect a set of semantically changed target words, a set of stable words, and a set of emerging new words. Third, I evaluate the performance of the definition generation model in both time periods, and analyze whether semantic change impacts performance. Fourth, I compare the results with cross entropy and perplexity scores for the same inputs. The results indicate that (i) the model's performance deteriorates more for semantically changing words compared to semantically stable words, (ii) the model exhibits significantly lower performance and potential bias for emerging new words, and (iii) the performance does not correlate with loss or (pseudo)-perplexity scores.

Acknowledgements

I would like to thank my supervisor Raquel Fernández for her extensive feedback, uplifting work ethic and critical point of view. I always found our meetings inspiring, motivating, and fun. I would also like to thank Pauline Sander, Andrey Kutuzov, and Mario Giullianelli for the interesting discussion meetings we've had about computational linguistics and semantic change. It is special and exciting to meet people with like minded interests. A special thanks to Mario who, together with Raquel, planted the seeds of this thesis.

Another acknowledgement should be made for H., K., and L., who invested time and effort to carefully annotate the results.

I would like to thank the committee members, Pia Sommerauer, Jelke Bloem, and Malvin Gattinger for taking the time to read me thesis, and for their enthusiastic and interesting questions during the defense.

I am especially grateful to Tanja, for her kind and listening ears, whom I always enjoyed talking to. And all my friends and family: thank you for your many attempts to understand my research, for your unconditional support and kindness, thank you for being in my life.

Contents

Chapter 1

Introduction

Models of natural language processing (NLP) are ubiquitous in modern society, encompassing various language applications such as search engines and chatbots. With the rise of large neural language models, state-of-the-art performances have been achieved for a variety of downstream tasks (Min et al., 2021; Qiu et al., 2020; Sun et al., 2022). However, recent studies have indicated that such language models may suffer from performance deterioration over time (Lazaridou et al., 2021; Osborne et al., 2014; Dhingra et al., 2022; Luu et al., 2022). This is not surprising, considering the world is constantly changing and that language is continuously evolving. A language model's ability to generalize well to future data from beyond their training period is called *temporal generalization* (Lazaridou et al., 2021, pg. 2).

Lazaridou et al. (2021) showed that the performance of several Large Language Models (LLMs) deteriorates as the time of the test data lies further in time from that of the training period. They showed an increase in the LLM's perplexity of 16% within one year time, and that the LLMs performance deteriorates for the downstream task of question answering. At the same time, Agarwal and Nenkova (2022) show that the performance on downstream tasks like sentiment analysis and named entity recognition does not necessarily deteriorate over time. Therefore, they state that performance only deteriorates for tasks where the correctness labels of the task are time dependent.

Both Lazaridou et al. (2021) and Agarwal and Nenkova (2022) focus on how factual changes reflected in data sets impact a language models' performance. However, language *use* also changes over time, with new words and phrases emerging, and words obtaining new senses and connotations. Naturally, these changes are also reflected in the data we use to train language models.

Recent studies have pointed out that performance deterioration over time is related to semantic change, but these results are mainly based on a LLM's perplexity scores (Su et al., 2022; Ishihara et al., 2022). Röttger and Pierrehumbert (2021) point out that perplexity is not necessarily an indicator of performance deterioration on downstream tasks. Moreover, increase in perplexity does not give insight into how the performance of language models on downstream tasks

is impacted by semantic change. Besides factual changes, does semantic change in temporally shifted data also impact LLM performance on downstream tasks? It is relatively unambiguous to assess a LLMs ability to process factual information using a question answering task. However, semantic change can be subtle and implicit, and therefore difficult to expose. If performance indeed deteriorates due to semantic change, how does this performance deterioration manifest in a LLMs performance on downstream tasks? With the increasing use of large language models, and especially generative applications like ChatGPT and Bard, it is important to be aware of the possible mistakes that LLMs make when they are not up-to-date.

The goal of this thesis is to examine the impact of semantic change on a LLM's ability of temporal generalization on the task of definition generation. Examining temporal generalization via definition generation offers a dual perspective: on the one hand, it allows for a quantitative measurement of performance, and on the other hand, the generated definitions themselves provide human interpretable output that give insight into how process LLMs semantic information of the target words under scrutiny. Chapter 2 provides a background overview of language models, semantic change, and LSCD systems. Chapter 3 addresses the limitations of the current static language modeling paradigm and the potential obstacles involved in incorporating temporal dynamics into language models. Chapter 4 outlines the methodology used to investigate the impact of temporal shifts and semantic change on the LLM `T5-base` (Raffel et al., 2022). This includes (i) constructing diachronic corpora C_1 and C_2 from Twitter and Reddit, (ii) selecting sets of stable, changing, and emerging target words with the help of an LSCD system, (iii) fine-tuning `T5-base` for definition generation, (iv) generating definitions for a set of contexts containing the target words, (v) conducting human evaluation on the generated definitions, and (vi) calculating cross-entropy loss and perplexity scores for the context sentences.

Chapter 5 presents the results, which indicate that (i) the model's performance is adversely affected when processing temporally shifted input data compared to input overlapping with `T5-base`'s pre-training period, (ii) the performance deteriorates more for semantically changing and emerging words as opposed to semantically stable words, and (iii) cross-entropy loss and (pseudo)-perplexity scores do not necessarily indicate poor performance on the task of definition generation. The results demonstrate that definition generation is a promising and intuitive approach to evaluate an LLM's ability of temporal generalization, particularly with respect to its capacity to process semantic information rather than syntactic structures or factual information. My findings also underline the importance of assessing the capacity of temporal generalization of fine-tuned LLMs more explicitly that through perplexity scores, as perplexity is necessarily representative of how well LLMs performs on downstream tasks.

Chapter 2

Background

The field of Natural Language Processing (NLP) studies the design and analysis of computational algorithms and representations for processing natural human language. The goal of NLP is to develop new computational tools for applications that involve processing human natural language. Examples of such practical applications are translating between languages, extracting information from texts, and holding conversation with humans (Eisenstein, 2019). Language is a complex and dynamic system, subject to continuous evolution. One particular aspect of this evolution is the phenomenon of semantic change, whereby the meanings of words within a language undergo transformations over time. Factors such as cultural shifts, technological advancements, and societal changes can contribute to these shifts in word meanings.

This background chapter provides an overview of how NLP models are conventionally trained and evaluated. Furthermore, I discuss different forms of semantic change, and approaches in NLP to detect semantic change between corpora. Section 2.1 explains the use of word embeddings in NLP, section 2.2 explains different pre-training architectures used to train and fine-tune large language models, and section 2.3 explains how models of NLP can be evaluated. Section 2.5 sets out the phenomenon of semantic change, the different forms in which it can occur, and its possible driving forces. Section 2.6 and provides an overview of the state-of-the-art approaches to LSCD, as the experiment utilizes LSCD systems.

2.1 Word embeddings

Models of NLP represent the meaning of words of natural language with *word embeddings*, which are multi-dimensional vectors that reflect how words co-occur with other words (Almeida and Xexéo, 2019; Eisenstein, 2019). Word embeddings can be regarded as points in a multi-dimensional vector space, often called a *semantic vector space*. The main idea is that the word embeddings represent the meaning of the words such that words with similar meanings lie closer to-

gether in this semantic vector space. The similarity between word embeddings in a semantic vector space can be estimated using different geometric measures, depending on the model at hand (Boleda, 2020). Different geometric relations may also reflect different types of semantic relations, such as synonymy and analogy between words (Almeida and Xexéo, 2019; Ranjan et al., 2016). With this, models of NLP rely on the *distributional hypothesis* (Boleda, 2020; Almeida and Xexéo, 2019; Eisenstein, 2019). The distributional hypothesis presumes that words that are using the same context, tend to have the same meaning. Following this presumption, the meaning of a word could be represented by the context in which it occurs.

A distinction is commonly made between *type-based* and *token-based embeddings* (Tahmasebi et al., 2021; Kutuzov et al., 2022; Kurtyigit et al., 2021). In the early stages of NLP, models were trained to learn type-based embeddings, which represent each word type in the vocabulary by a single embedding. These are also referred to as *vector space models*. In essence, *type-based word embedding* reflects how the word or n-gram it represents co-occurs with other words within a specified distance.[1] This way, the meaning of each word type is represented by their distributional properties in the corpus as a single word embedding. Type-based embeddings can be learnt from co-occurrence statistics of the training corpus, but can also be learnt using machine learning techniques like dimensionality reduction (e.g. SVD) and neural networks (e.g. CBOW or Skip-Gram) (Qiu et al., 2020). Popular examples of models that learn type-based embeddings are GloVe (Pennington et al., 2014) and Word2Vec (Mikolov et al., 2013).

Type-based embeddings are *context-free* or *context-independent*: the representation of a word is the same regardless of the context it appears in (Qiu et al., 2020). Using a single representation of distributional properties based solely on word form, it is impossible to differentiate between word meanings across different contexts (Tahmasebi et al., 2021). Consequently, it is impossible to differentiate between multiple senses of polysemous words when using type-based embeddings. For instance, the occurrences of the phrases MONEY BANK and RIVER BANK in the corpus will contribute to the same distributional representation of the word type BANK. Thus, BANK is represented by the same embedding in both sentences, disregarding how context can influence word meaning.

Nevertheless, NLP applications relying on type-based embeddings have reached high performances, such as for sentiment classification, named entity recognition, part-of-speech tagging, information retrieval, and lexical semantic change detection (LSCD) (Kutuzov et al., 2018; Schlechtweg et al., 2019; Min et al., 2021).

Token-based embeddings, on the other hand, represent words of natural language context-dependently, because every word-context pair yields a unique representation (Qiu et al., 2020; Min et al., 2021). Examples of such language models are BERT (Devlin et al., 2018), RoBERTa (Liu et al., 2019), `T5`(Raffel et al., 2022), BART (Lewis et al., 2020), and the GPT family (Brown et al.,

[1] Often called *window*, or else specified by the borders of a sentence.

2020; Radford et al., 2019, 2018). These are large neural models which contain an encoder layer, hidden layers, and a decoding layer. Their neural architectures make it possible to generate a unique context-dependent representations of the word of interest: First, the word-context pair is encoded and fed to the model, which is processed by the neural network in a series of layers. Each of these hidden layers performs a transformation on the input of the previous layer. Once the input sequence has been processed by the model, each layer of the neural network can be considered a certain representation of the input sequence. Next, the contextualized word embedding can be retrieved from these hidden layers. For instance, the last hidden layer can be used as a contextualized word embedding, but also be constructed by performing transformations on a selection of hidden layers, depending on the specific model architecture (Wang et al., 2020). Further explanation on the pre-training architectures of such models is provided in section 2.2.1.

Since token-based embeddings have a unique representation for every word-context pair, token-based embeddings are capable of representing multiple senses of polysemous words. Language models that learn token-based embeddings, particularly deep neural language models, have outperformed type-based approaches on a wide variety of tasks, which is why token-based approaches using deep neural networks have come to dominate the field of NLP (Min et al., 2021). The state-of-the-art models that learn token-based embeddings are called Large Language Models, and will be further discussed in the next section.

2.2 Large Language Models

A language model (LM) is a computational model that represents human natural language as probability distributions over (sequences of) words in a language. Language models are trained to learn statistical patterns and structures of language by analyzing large amounts of textual data. The main idea is that a language model can predict the probability of a given word or sequence based on the context words in a sentence or document (Eisenstein, 2019; Almeida and Xexéo, 2019; Ranjan et al., 2016). Large Language Models (LLMs) are deep neural language models that contain substantially more parameters than early neural language models. Since LLMs contain many hidden layers and parameters, they also require much more training data to fully train the model parameters and prevent overfitting (Qiu et al., 2020).[2]

LLMs are first *pre-trained* by specific training objectives, whereafter they can be *fine-tuned* for new tasks. Qiu et al. (2020) describe LLMs as *second generation language models* because they inherently differ in how they can be deployed for downstream tasks. Whereas type-based (*first-generation*) models require additional architectures to be used for downstream tasks, which still

[2]Overfitting occurs when a machine learning model becomes overly specialized to the training data and fails to generalize well to new, unseen data, resulting in poor performance. It happens when the model captures noise or random variations in the training data rather than the underlying patterns, leading to decreased accuracy and predictive power.

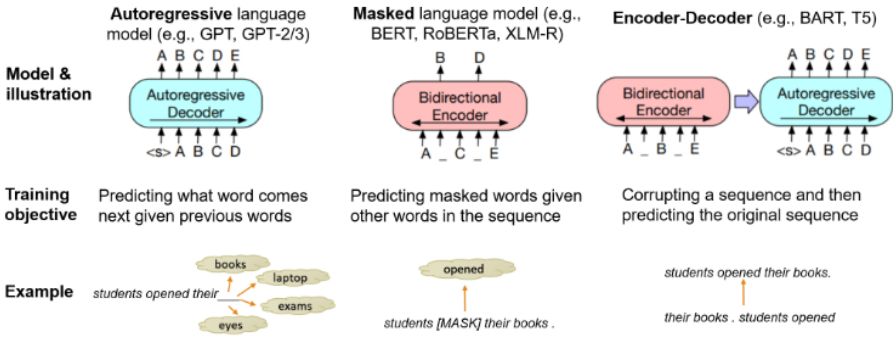

Figure 2.1: Three classes of pre-training architectures (Min et al., 2021)

must be trained from scratch, LLMs can be further tweaked to transfer the information learnt for one task to be used in new tasks by means of fine-tuning and prompting (Min et al., 2021; Sun et al., 2022; Qiu et al., 2020).

2.2.1 Pre-training LLM architectures

The introduction of the Transformer model architecture by Vaswani et al. (2017) has facilitated the ability to increase pre-trained LLM sizes. A Transformer architecture is an encoder-decoder architecture, consisting of an encoder for input sequence encoding and a decoder for output sequence generation. Transformers have a self-attention mechanism which allows them to capture dependencies between different positions in the input sequence. This mechanism enables the model to attend to relevant tokens and weigh their contributions effectively. Unlike recurrent models that process inputs sequentially, transformers can capture long-range dependencies in parallel, making it easier to model larger contexts. This parallelization leads to efficient computation, making it easier to scale up the model to handle larger inputs and larger parameter sizes. As a result, nearly all popular language models are now based on the Transformer architecture (Min et al., 2021).

Even though LLMs are all deep neural models, LLM pre-training architectures may vary in terms of how they encode the input and the objectives used for pre-training. Min et al. (2021) describe three classes of pre-trained language models: autoregressive language models, masked language models, and encoder-decoder models (see 2.1. These models are typically trained using self-supervised learning approaches. Self-supervised learning is a type of training where a model learns from unlabeled data by defining a pretext task. The goal is to learn useful representations of the input data that can then be fine-tuned or utilized for downstream tasks (Min et al., 2021).

Autoregressive language models are trained to predict the next word based on all previous input words with the objective of maximizing the log-likelihood.

Such models are called *unidirectional*, since their predictions are only dependent on the words preceding the target word, and not on the following tokens of the input sequences. Popular examples are GPT, GPT-2 and GPT-3 (Radford et al., 2018, 2019; Brown et al., 2020).

Masked language models (MLMs) are trained to predict the masked word conditioned on the entire sequence, taking into account the context words on both sides. When training an MLM, random words are chosen to be masked, using a special [MASK] token. The training objective is to recover the original tokens at the masked positions. This forces the model to collect *bidirectional* information in making predictions (Min et al., 2021). Popular examples are BERT (Devlin et al., 2018) and RoBERTa (Liu et al., 2019).

Encoder-decoder models, also referred to as sequence-to-sequence models, are trained to generate a sequence $y_1, y_2, \ldots, y_n$ given an input sequence $x_1, x_2 \ldots x_n$. So instead of predicting a single word based on an input sequence, their training objective is to predict an output sequence, by maximizing the log-likelihood of the output sequence $y_1, y_2, \ldots y_n$ conditioned on the input and a number of other parameters. In order to do so, the input token sequence can be modified in various ways, for example by shuffling the input sequence or by masking words in the sequence (similar to MLM). The output sequence should be the reconstructed, original sequence. Popular examples are T5 (Raffel et al., 2022) and BART (Lewis et al., 2020).

The training objectives of transformer LLMs allow them to consider the token sequence order, either unidirectional or bidirectional. Therefore they perform better at tasks where word order, grammar, and long-term dependencies between words are relevant (Qiu et al., 2020). Since LLMs take sequences as input, this enables the extraction of context-dependent word embeddings: given a particular sequence, the hidden layers encoding the target word can be used to construct context-specific word embeddings. There are different strategies to produce these embeddings, such as summing or averaging over a set of n hidden layers, concatenating them, or using *Max-pooling* and *Min-pooling* (Miaschi and Dell'Orletta, 2020).

The reader may notice that none of the pre-training architectures include any training objective that relates to the temporal feature of the language. In fact, the training input is often fed to the models randomly, and the models architectures do not consist of any properties that allow the LLMs to relate any of the semantic information that they learn from the training data to the time at which it was uttered or written. In chapter 3, I further elaborate on the consequences this has on a LLMs performance.

2.2.2 Prompting and fine-tuning

The deep neural architecture of LLMs allows them to be fine-tuned for new downstream tasks. Rather than having to design additional models for each specific task, the intended tasks can be reformulated so that they are similar to the tasks that were used for training the language models (Sun et al., 2022; Min et al., 2021; Liu et al., 2023). The initial pre-trained models can be considered

as trained for a particular task already: In the case of MLM class language models, the initial model was trained to perform the task of 'masked word prediction', the autoregressive models were trained for 'next word prediction', and the encoder-decoder models were trained to predict sequences.

Prompting is the practice of adding natural language text to the input or output, often in the form of instructions, demonstrations or templates, to encourage the pre-trained models to perform a specific task (Min et al., 2021). The idea is that prompting can be used such that the NLP task of interest is simplified or reduced to a similar task that the pre-trained language model learnt during training. To illustrate, one can reformulate the "next word prediction" objective to a word analogy task. Simply prompt the model with the input: *Complete this sentence: A man is to a woman as a king is to a ....* Since the training objective was already next word prediction, the model will return whatever word is most likely to complete the sentence. In this case, the completion of the analogy that was prompted. By designing tasks this way, one can utilise the capacities that the model has obtained during pre-training.

Fine-tuning is a process of tweaking a pre-trained model further for a different but related task. Fine-tuning is often done by adding, tweaking or replacing one or more hidden layers of the initial neural model. It is common to *freeze* the remaining layers from the original model, such that the weights of these layers remain the same when training the model for the new task (Min et al., 2021). To illustrate, a LLM can be fine-tuned for the task of sentiment classification by adding an additional classification layer to the pre-trained LLM, and optionally modifying some of the final layers of the pre-trained LLM. Next, the model can be fine-tuned on a data set of text samples along with their corresponding ground truth sentiment labels. For instance, each sample may be a sentence or a short paragraph, and the labels can be positive, negative, or neutral. During training, the model iteratively adjusts its weights to better capture sentiment-related features and patterns, and learns to associate specific language patterns with sentiment labels (Devlin et al., 2018; Raffel et al., 2022; Min et al., 2021; Qiu et al., 2020).

This way, the information processed during pre-training for one task can be leveraged for a new task, without having to train a model from scratch. Building and evaluating a model for a specific downstream task is computationally demanding in itself, if a model performs well at one similar task, we can fine-tune it for the requested task without having to train it from scratch.

2.3 Evaluation of language models

When evaluating how well NLP models have learnt to represent and process natural language, a distinction is often made between *intrinsic* and *extrinsic* evaluation (e.g. Jang et al. (2022); Lazaridou et al. (2021)).

Intrinsic evaluation concerns evaluating the model parameters directly, either by examining the word embeddings, or by examining the model likelihoods. The first way to do this is by examining whether and how the geometric relations

between the word embeddings correspond to human judgements. For instance, the human judgements on semantic similarity between word pairs from ground truth data sets like WordSim-353 (Finkelstein et al., 2001) or SemEval-2012 task-2 (Jurgens et al., 2012) can be compared to the cosine similarities between the word embeddings of a model. The better the similarity scores between a model's word embeddings align with the human similarity judgements, the better the model is considered to represent a natural language.

LLMs can also be evaluated intrinsically by examining the model likelihoods using *cross entropy loss* and *perplexity scores*. Since LLM can be considered a probability distribution over all possible text sequences of a language, the cross entropy loss (H) and perplexity (PPL) and can be used to estimate how well the language model predicts a sample S (Ranjan et al., 2016). The cross entropy measures the degree of 'uncertainty' when encountering a text sequence, while the perplexity measures the degree of 'surprisal' a model has in predicting a text. The two measures are closely related, as $\texttt{PPL}(S) = e^{\texttt{H}(S)}$.

Let $\mathbb{P}(S)$ denote a language model's probability of the sequence $S = w_1, w_2, \ldots, w_n$, where each w_i is a word in the vocabulary, and $n \in \mathbb{N}$ is the number of words in the sequence. The higher the cross entropy loss, the more surprised the model is to encounter the sequence S. The cross-entropy score is calculated by:

$$\texttt{H}(S) = -\frac{1}{n} \log \mathbb{P}(S) \tag{2.1}$$

Perplexity is defined as the language model's inverse probability of the sequence S, normalized by n. The PPL score of a text $S = w_1, w_2, \ldots, w_n$ is calculated as follows:

$$\texttt{PPL}(S) := \sqrt[n]{\frac{1}{\mathbb{P}(S)}} \equiv \frac{1}{n} \log \mathbb{P}(S) \tag{2.2}$$

The higher the perplexity score for a given sequence is, the more 'surprised' the model is to encounter this sequence.

Notice that $H(x) < \texttt{PPL}(x)$ for all $x \in \mathbb{R}$; the difference is that the perplexity metric PPL penalizes low sequence-probabilities $\mathbb{P}(S)$ more than the cross-entropy loss does. In essence, cross entropy and perplexity provide similar information about the performance of a language model, but perplexity is a more intuitive and easier-to-interpret version of cross entropy. Lower perplexity scores indicate better performance, suggesting that the language model has a better understanding of the given data and can make more accurate predictions. Cross entropy, on the other hand, provides a more direct measure of dissimilarity between predicted and actual distributions, without the intuitive interpretation of perplexity as a measure of average uncertainty.

Since the cross-entropy and perplexity measure the degree of uncertainty or surprisal based on the likelihood of a complete sequence $(w_1, \ldots w_n)$, these metrics disregard how the presence of each word in the sequence contributes to the sequence likelihoods. For instance, it could be the case that only one

word in the sequence is particularly unlikely, while the rest of the sequence is relatively likely. Therefore another metric is *pseudo-perplexity* (PPPL), which subsequently compares the complexity of each term in the entire sequence. A model's pseudo-perplexity for a text T containing N tokens is:

$$\texttt{PPPL}(T) := \exp -\frac{1}{N} \sum_{S \in T} \texttt{PLL(S)} \tag{2.3}$$

where PLL is the *pseudo-log-likelihood* of the sequence $S = (w_1, \ldots w_n)$, computed by:

$$\texttt{PLL}(S) = \sum_{t=1}^{n} \log \mathbb{P}(w_t | S_{\setminus w_t}) \tag{2.4}$$

Extrinsic evaluation assesses how well a language model performs at a downstream task, often measured by accuracy and F1 scores. Common examples of such downstream tasks are sentiment analysis, named entity recognition, part-of-speech tagging, and question answering (Min et al., 2021; Sun et al., 2022). Popular evaluation benchmarks are GLUE (Wang et al., 2018) and SuperGLUE (Wang et al., 2019), which evaluate a model's sentence understanding through natural language inference problems, from resolving syntactic ambiguity to high-level reasoning.

Most evaluation data sets and benchmarks do not include any annotations or tasks that relate to the capacity for temporal generalization. That is, none of the mentioned benchmarks evaluate specifically how well a model performs on data from time periods that it was not trained on, nor do they evaluate the performance of models on specific periods of time. This is yet another way in which the field of NLP follows a static language modeling paradigm, which will be further elaborated on in chapter 3. In the next sections, I will discuss how and why language use changes over time (2.5), and how lexical semantic change detection systems are used to detect changes in language use (2.6).

2.4 Definition generation task

This thesis proposes and experiments with the evaluation of LLMs on the task of contextualized definition generation. This section covers some of the related works and challenges in (contextualized) definition generation.

Noraset et al. (2017) first introduced the task of generating a definition for a given word and its embedding. They explain that generated definitions of words can be a more direct and transparent representation of the embeddings' semantics. One constraint of their approach was that it relied on type-based embeddings. Since type-based embeddings collapse all contexts into one representation, this makes it difficult to generate different definitions for polysemous

words for which the definition is context-dependent. Gadetsky et al. (2018) addressed this issue by conditioning the model output on a sense-specific example sentence while still using type-based embeddings.

Noraset et al. (2017) pointed out some challenges and mistakes that definition generation models generally struggle with. The first is that of self-reference, which means that definitions make use of the target word itself to define the target word. A second mistake that is often made is part-of-speech mismatching: the generated definition corresponds to a different part-of-speech of the target word. This happens, for instance, when the provided context contains a noun form, but the model generates a definition for the verb form of the target word instead. A third issue is that the generated definitions can be incoherent or lack fluency. Fourthly, it is a challenge to ensure that the generated definitions are not too specific to the provided context nor too general.

Token-based methods enabled the possibility to generate context-specific definitions directly from the contextualized word embeddings. Mickus et al. (2019) were the first to approach definition generation as a sequence-to-sequence problem rather than token-to-sequence and train a definition generation model on the dataset constructed by Gadetsky et al. (2018). They used the Transformer model of Vaswani et al. (2017) and evaluated the generated definitions by comparing them to the definitions from the Oxford dictionary. Mickus et al. (2019) observe that a common mistake of definition generation models is the generation of *hallucinations*, which they describe as output where the factual information is wrong, although a non-careful reader may be deceived into thinking it's a proper definition.

Huang et al. (2021) optimized the definition modeling task to generate definitions with appropriate specificity. The approach of Huang et al. (2021) relies on a re-ranking mechanism on a pre-trained Text-to-Text Transfer Transformer (T5) model. This method of Huang et al. (2021) will be used in this thesis; further details on their pipeline are provided in Section 4.4.

A recent approach by Giulianelli et al. (2023) uses the `Flan-T5` language model to generate definitions. `Flan-T5` is a version of `T5` that is already fine-tuned on 1.8K tasks phrased as instructions and collected from almost 500 NLP datasets. Thanks to its massive multi-task instruction fine-tuning, the model excels at generalizing for unseen tasks. To obtain definitions from `Flan-T5` , they use prompts consisting of an example usage followed by an instruction. They use greedy search with target word filtering to ensure that the generated definitions do not contain the target word itself, which is a simple parameter-free decoding strategy.

2.5 Semantic change

Language is a cultural, dynamic and constantly evolving phenomenon. The reader likely recognizes how movies and books of only a few years ago may appear old fashioned. This may be because of the difference in pronunciation (phonological change), word choice (lexical change), and even due to the way the

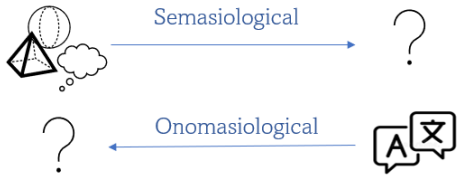

Figure 2.2: Two approaches to the study of lexicons

sentences are built (syntactic change) (Traugott, 2017). But the meaning that speakers convey with certain words can also change over time. For instance, the word VIRAL was primarily used to describe the spread of a virus or disease, but has come to describe content that spreads rapidly online in the age of social media (OED, 2023d). Another recent example is the word WOKE. Originally, this word was simply the past tense of the word 'wake', but in the recent years it is used to describe someone who is socially and politically aware, particularly with regard to issues of social and racial inequality (OED, 2023e). When the meaning that words convey changes over time, this is called semantic change. In this section, I first elaborate on what semantic change is. Next, I discuss different forms of semantic change, and the potential influences driving semantic change.

2.5.1 What is semantic change?

When the meaning that speakers convey with certain words changes over time, this is called semantic change. To determine whether a conveyed meaning has changed, one must first establish what word meaning is in the first place, which is a debate in itself.

When asking why humans choose to use the words they do and what they mean by them, one can view this from a onomasiological perspective or a semasiological perspective (Tahmasebi et al., 2021; Traugott, 2017). The onomasiological perspective focuses on the identification and naming of objects and concepts. The semasiological perspective, on the other hand, asks what meaning a word conveys within a given context. Change in language use from the onomasiological perspective is referred to as *lexical change*, while *semantic change* refers to change in language use from a semasiological perspective (Tahmasebi et al., 2021).[3]

So, how do we determine what the conveyed meaning is of a word within a certain context? In lexicography and linguistics, the meaning of a word is commonly decomposed into word *senses*. As was the case for BANK, words of the same form can convey different meanings. The meaning of a word is therefore commonly described by a set of senses $||w|| = \{s_1, \dots s_n\}$, each of

[3]Semantic change of words in particular (opposed to n-grams or phrases) is also referred to as *lexical semantic change*.

which is designated its own definition in a dictionary (Tahmasebi et al., 2021; Kilgarriff, 1997). Semantic change can then be seen as the change in this set of senses, either when a word obtains or loses a sense, or when the meaning of one of the senses changes.

Nevertheless, decomposing word meaning into senses or usages merely shifts the question: what meaning does a specific word (sense) convey, then? Tahmasebi et al. (2021) define linguistic meaning in terms of *denotation* and *connotation*. Denotation covers the "neutral" information content that a word refers to, while connotation refers to attitude or the sentiment that a word conveys. Using this notion of meaning, they define a *word sense* as referring to the "combination of the lexical item and the particular recognized meaning of that lexical item". Tahmasebi et al. (2021) illustrate the distinction between connotation and denotation with STINGY and THRIFTY, as these have roughly the same denotation, but a different connotation. However, the notion of *connotation* is subjective, because different speakers can have different connotations and/or associations with words. Consider the words SUNNY and RAINY, whose polarity is dependent on the subject; in some geographical areas in the world, the sun evokes associations of joy, while in others it represents drought and death. Likewise, rain may evoke associations of fertility and prosperity in some areas, while in others it evokes sadness. Still, intuitively, the sense RAINY in the sentence 'It is going to be rainy!' is identical, regardless of the sentiment it evokes in the speaker. This raises the question to what extent meaning is subjective to the interlocutors.

In theoretical linguistics and philosophy of language, meaning is also sometimes defined in terms of *extension* and *intension* (Chalmers, 2002). The extension refers to the possible referents that a word can have, either objects or concepts.[4] The intension refers to the semantic properties that specify the conditions for when a term could be used. For example, the intension of RED is the conditions for an object or concept to be red, while the extension denotes the set of all possible red entities that meet these conditions. As Chalmers put it, "the intension is evaluated at a possibility, and returns an extension in that possibility" (Chalmers, 2002, p.g. 145). In this framework, associations and connotations are not considered as descriptive aspects of meaning.

Of course, there are many more views on what meaning is, and on how or whether meaning is manifested in the real world and in the human mind. This has been a topic of interest in the philosophy of language for centuries. For example, Frege described meaning in terms of *Sinn* (sense) and *Bedeutung* (reference), where the reference is the object or concept an expression refers to, and the *Sinn* is the "thought" the word expresses (Chalmers, 2002). Other philosophers, like Wittgenstein, don't believe in such a thing as representations of meaning at all (Chalmers, 2002). As such, meaning cannot be decomposed into concepts like referents, intension, or connotation in the first place.

Kilgarriff (1997) points out that humans often have strong intuitions of the

[4]There is of course debate about whether words with non-material referents, like UNICORNS or SANTA CLAUS, are considered to have an extension.

different possible meanings of a word, however, there exists no proper taxonomy to classify word senses. Moreover, he illustrates that the way humans classify word senses is task-dependent; humans classify word senses differently depending on the level of abstraction by which they interpret word meaning.

Some studies therefore address word meaning not by *senses* but by empirically observed *usages* as a basis for categorising different meanings that a word can convey. For example, Schlechtweg et al. (2021) empirically determined clusters of "word usages", where each cluster consists of word-context pairs. These clusters were computed from over 100,000 human judgements on semantic proximity between pairs of sentences containing the same target word. With this, they approach word meaning in terms of semantic (dis)similarity between words throughout contexts.

All in all, meaning is context dependent, and culture dependent, and subjective. Therefore, an encompassing definition of semantic change is difficult to pinpoint, and lies in the philosophical domain. Different frameworks to define semantic change expose different aspects of semantic meaning, and often don't do justice to the muddled practice. Likewise, in this thesis, I consider lexical semantic change as the change in meaning of a word over time, reflected by its use. This could either be because the intension, extension, or association of a word changes, which is reflected in the various ways in which the word is used in practice.

2.5.2 Types of semantic change

The examples WOKE and VIRAL are both instances of semantic change where an existing word has obtained an extra sense, which is called *sense birth*. There are numerous other types of semantic change. Senses can also stop being used by a linguistic community, called *sense death* (Tahmasebi et al., 2021). Words can also obtain *broader* meanings, when a word sense becomes broader in meaning at a later time. This means that the *intension* changes, enabling more possible extensions. For instance, HACKER was formerly used to *only* refer to 'someone who use their computer skills to illegally access and sometimes tamper with information in a computer system', while the term is now used to refer to 'any person who is an expert at programming and solving problems with a computer' (OED, 2023b). The latter definition describes more possible objects than the former, making the extension larger. Likewise, word meanings can also *narrow*.[5]

Semantic change can also be the result of various forms of figurative speech. For instance, words can obtain new senses due to metaphorical language use, such as HEAD in 'head of state'. Metonymy is a type of semantic change that happens when the part of an object or concept is used to refer to the whole. For instance, THE CROWN can be used to refer to the UK royal house, and HOLLYWOOD to the US film industry. Another form of semantic change is *ellipsis*, where a part of an n-gram is used to refer to the extension of the complete n-gram, as is the case for MOBILE to refer to a mobile phone.

[5] Broadening and narrowing of word meanings is sometimes also referred to as generalization and specialization respectively.

Grammaticalization is a form of semantic change where words undergo a morphosyntactic change, i.e. words obtain a new grammatical function (Boleda, 2020; Tahmasebi et al., 2021). For example, PRETTY was first used as an adjective used similar to GOOD-LOOKING, and became the adverb synonymous with SOMEWHAT (Traugott, 2017).

Semantic change can also occur when speakers of one language adapt the meaning of words from another language. Tahmasebi et al. (2021) give the example of the Swedish verb SUGA which describes the use of the mouth to pull in liquid or air. This word has acquired a new sense describing 'to be unpleasant, inferior, etc.' borrowed from the English polysemous word SUCK.

When a new use of a word emerges, it consistently coexists with the previous usage (Traugott, 2017). Therefore word meaning rarely undergoes change at a single defined moment in time. Instead, words typically transition through polysemous stages before a dominant sense is established within a linguistic community (Traugott, 2017; Tahmasebi et al., 2021).

2.5.3 Neologisms

A distinct form of language change is the genesis of new words in a language's vocabulary. Such words are called *neologisms* (Lehrer, 2003). Neologisms are thus not instances of semantic change, but rather of lexical change, which is viewed from the onomasiological perspective.

Lehrer (2003) describes different forms of neologisms. Neologisms can arise through *blends* of existing words in the lexicon, where the underlying compounds of two words are combined (e.g. SMOKE + FOG $\mapsto$ SMOG). Neologisms can also be the result of concatenating two words (e.g. COCACOLONIZATION), or by the establishment of a words acronym (e.g. ID) (Lehrer, 2003).

2.5.4 Drivers of semantic change

As discussed, there are various ways in which the meaning of a word can change. Certain linguistic phenomena like figurative speech, foreign language use and metonymy can be triggers for semantic change. But what determines when new language use catches on, and manifests itself within a linguistic community? Several laws of semantic change have been proposed (Tahmasebi et al., 2021). For example, the *Law of Innovation* states that polysemous words tend to change more quickly than monosemous words. The *Law of conformity* states that frequent words (like stop words) change more slowly than infrequent words. The *Law of Parallel Change* states that semantically linked words such as synonyms or antonyms, also undergo similar change over time (Tahmasebi et al., 2021). *Zipf's Law* states that frequent forms tend to become shorter (Lehrer, 2003).

Cultural changes can also trigger semantic change (Tahmasebi et al., 2021; Hamilton et al., 2016a). Examples of cultural change are the emergence of new technologies, hot topics and (the cease of) taboos. For instance, after the invention of the mechanical car, the sense of CAR referring to non-motorized vehicles

has shifted to refer to motorized vehicles (Kutuzov et al., 2018). A recent example of a taboo triggering semantic change, is the word MASCARA, which has come to be an alias on the social media platform TikTok for domestic violence.[6] In fact, semantic change is likely accelerated by social media, because social media has enabled a great increase in the spread of communication (Tahmasebi et al., 2021; Del Tredici et al., 2019).

When words have undergone semantic change, this is naturally also reflected in the contexts in which they are used. In turn, this is reflected in textual data, and in the patterns in which words co-occur in sentences and texts. This data can be used to train systems that detect lexical semantic change between diachronic corpora. The next section discusses LSCD approaches, and how they can be used to detect different types of semantic change.

2.6 Lexical Semantic Change Detection

Change in language use is naturally mirrored in the contexts in which words are used, and reflected in the patterns in which words co-occur together in textual data. Therefore textual data from different time periods can be used to train models that help detect and analyse lexical semantic change. The area of NLP concerned with detecting and analysing semantic change at a lexical level between time periods is called Lexical Semantic Change Detection (LSCD). LSCD systems are designed to measure diachronic semantic shifts in a data-driven way (Kutuzov et al., 2018; Tahmasebi et al., 2021). The task of detecting or discovering semantic shifts from data can be formulated as follows. Given corpora $C_1, C_2, ...C_n$ containing texts created in time periods $1, 2, ...n$, the task is to locate words with different meaning in different time periods, or to locate the words which changed most (Kutuzov et al., 2018). Word embeddings are commonly used as input representation for this task. Approaches based on word embeddings assume that the changes in a word's collocational patterns and contexts also reflect or indicate changes in word meaning. This assumption implies that semantic shifts are reflected in large corpora through change in the context of the word which is undergoing a shift (Kutuzov et al., 2018).

Section 2.6.1 explains different approaches to type-based LSCD systems, and in particular the LSCD system that will be used in the experiments for this thesis. Section 2.6.2 provides an overview of approaches that make use of (contextualized) token embeddings.

2.6.1 Type-based LSCD systems

Various LSCD systems have been developed that make use of type-based word embeddings. The general idea is to train vector space models for each time period, and compare the word embeddings of time specific semantic space. This can be done by training models on each subcorpus separately (Gulordava and

[6] See https://www.euronews.com/culture/2023/02/03/what-is-the-mascaratrend-and-is-it-an-adequate-tool-for-free-speech-on-tiktok

Baroni, 2011; Jatowt and Duh, 2014; Eger and Mehler, 2017; Kulkarni et al., 2015), or by incrementally training models for each time period (Kim et al., 2014). The way the word embeddings from different time periods can be compared, differs by the methods used to train the models.

When embeddings from models of different time periods share the same dimensions, the word embeddings can be compared directly. This is the case, for example in Gulordava and Baroni (2011) and Jatowt and Duh (2014), who use word embeddings where every dimension corresponds to a term from the joined vocabulary of subcorpora $C_1, C_2, \ldots C_n$. The semantic change is quantified using cosine similarity between the word embeddings from different time periods, so-called *self-similarity* (Jatowt and Duh, 2014). Words with a high self-similarity are considered *stable* words, while words with a low self-similarity are considered to have undergone semantic change between the time-specific corpora. Jatowt and Duh (2014) further explain the behavior of the semantic change by analyzing the most common context words, and by comparing a target word with so-called "contrastive word pairs", pairs of words for which it is known that they should be highly similar at some point in time (e.g. mouse - rat).

Vector space models based on sole co-occurrence counts have sparse embeddings of thousands of dimensions, making their use computationally demanding. Therefore, it makes more sense to use more advanced and lower-dimensional models such as Word2Vec or SkipGram (Mikolov et al., 2013) and GloVe (Pennington et al., 2014). However, since these models have inherently opaque dimensions, and are randomly initialized, models of the same architecture will produce vector spaces of different dimensions (Kutuzov et al., 2018; Tahmasebi et al., 2021). This makes it meaningless to use cosine similarity between word embeddings of models form different time periods directly. Several methods have been proposed to surpass this.

Instead of training a separate model for each subcorpus $C_1, \ldots C_n$, Kim et al. (2014) incrementally train the models for each time period. This is done by initializing the vectors of the model of time period i by the vectors obtained from time the corpora $C_0, \ldots, C_{i-1}$.

Kulkarni et al. (2015) propose to *align* the vector spaces using linear transformations that preserve the general vector space structure. They assume that vector spaces trained on the same corpus are equivalent under linear transformations, and that most words do not change in meaning over time. They explain that when the alignment model fails to align words, this is a possible indicator of semantic shift. This idea of aligning vector spaces has been adapted in other LSCD systems (e.g. Hamilton et al. (2016a,b); Zamora-Reina et al. (2022)).

Alternatively, Dubossarsky et al. (2019) and Jatowt and Duh (2014) relabel target words during training with their timestamp, called *temporal referencing*. Furthermore, Eger and Mehler (2017) make use of *second-order embeddings*, which are constructed from the pairwise cosine similarity scores between all word embeddings in the semantic space. The second-order embedding $\overrightarrow{v_t^2}$ of the target word t equals $\overrightarrow{v_t^2} = (sim(\overrightarrow{v_t}, \overrightarrow{v_{t_1}}), \ldots, sim(\overrightarrow{v_t}, \overrightarrow{v_{t_{|V|}}}))$ where $\overrightarrow{v_{t_i}}$ is the initial vector representing target word t_i, and V is the set of terms in the vocab-

ulary. This way, each word is represented in terms of its similarity to all other terms in the semantic space. Since the vectors obtained from different time periods once again have corresponding dimensions, these second-order embeddings can once again by compared using cosine similarity scores. As there are many words in a vocabulary, this once again inflates the number of dimensions a the second-order embedding. To avoid this, Hamilton et al. (2016a) build second-order vectors from the k nearest neighbors of the target words, which reduces the vector representation to k dimensions. The cosine distance between two such second-order embeddings of a target word from different time periods is called the *local neighborhood distance* (LND). Hamilton et al. (2016a,b) observe that different methods of constructing word embeddings can reflect different properties. The first-order vectors reflect a words distributional use, while the second-order embeddings reflect a words use in relation to other words of the vocabulary. These methods therefore naturally capture different properties of words usage. Methods using first-order vectors considerably reflect semantic changes word usage, which they call *linguistic drift*. LSCD methods using second-order vectors reflect cultural changes in language use, which they call *cultural shift*.

2.6.2 Token-based systems for LSCD

Since contextualized token embeddings allow for context-specific representations, one could take the contextualized token embeddings from usages at different time periods. This is an advantage, because there is no need to align the vectors in a joint space (Tahmasebi et al., 2021).

Giulianelli et al. (2020) use the contextualized token embeddings of BERT to compare the use of target words at different time periods. They analyse the 100 words annotated with semantic shift scores by Gulordava and Baroni (2011) as target words, and take context sentences from the COHA data set (Davies, 2012). For each target word, they extract N contextualized embeddings from N sentences containing these target words from BERT. These representations are aggregated to a representation matrix, where each row contains the normalized contextual embedding of the target word. They cluster this representation matrix using k-means to automatically distinguish the different usages of each word, which they call *usage types*. By counting the number of occurrences of each usage type k in a given time interval they obtain frequency distributions for each time interval. When normalized, these can be seen as probability distributions over each usage type. The semantic change scores are quantified by measuring the change in the frequency distribution of each of the usages in the clusters.

Giulianelli et al. (2022) experiment with *grammatical profiling*, which measures the distance between distributions of grammatical parameters. The counts of each morphological features of a target word (such as tense and verbform) in a diachronic corpus can be used to construct a vector representing a grammatical profile. These vector representations can as an additional tool to existing methods that use for instance contextualized token embeddings. Their work shows

that grammatical profiling can be helpful to detect semantic change, although simpler, type-based systems still outperform this method.

Using pre-trained models for LSCD without pre-training on different time periods does raise some objections. Firstly, pre-trained models can be influenced by information that does not stem from the time period under investigation. Since temporal information is not taken into account, a (contextualized) embedding of a text uttered in the year 1900 will still be identical to that of a text uttered in the year 2000. A second issue is that the large amount of pre-training data could be dominating the corpus that is under investigation. A solution to this is to fine-tune the model further on the corpora of interest. However, with smaller data sets, this may not impact the model parameters enough (Kutuzov et al., 2022).

Schlechtweg et al. (2019) conducted a large study of over nine different LSCD systems, and concluded that the system using SkipGram vector space models for each epoch, aligned using Orthogonal Procrustes (OP), and measured using Cosine Distance (CD) outperforms all other approaches. Therefore, this `SGNS+OP+CD` system will be used to perform LSCD in this thesis.

2.7 Conclusion

As the real world changes, so does language, its lexicon, and the meaning of words. Linguistic changes are reflected in data that models of NLP are trained on. This enables us to use models LSCD systems, to analyze corpora and learn new things about semantic change. At the same time, and for the exact same reasons, NLP models are also impacted by the dynamic nature of language.

In the next chapter, I further delve into why state-of-the-art models of NLP are not designed to remain up-to-date due a static language modeling paradigm, which can ultimately result in the failure of temporal generalization. NLP models are used in various daily life applications and scientific research. In order to NLP models to remain up-to-date, they should be equipped to learn and represent language as a dynamic phenomenon.

Chapter 3

Temporal Generalization

Over the past few years, language models have made tremendous strides in NLP, achieving state-of-the-art performance on a wide variety of language tasks. Language models like BERT and GPT-3 have become ubiquitos in many NLP applications, from chatbots, to machine translation to sentiment analysis. Despite their impressive performance, concerns have been raised that the performances of language models deteriorate over time due to their static architectures (Dhingra et al., 2022; Lazaridou et al., 2021; Loureiro et al., 2022a; Jang et al., 2022). As we've seen in the previous chapter, language is a complex and dynamic system that is constantly evolving, with new words and phrases emerging, others falling out of use, and shifts in meaning and usages over time. Yet, most language models are trained using *a static language modeling paradigm*, as Bender et al. (2021) described it. This can lead to performance of language models deteriorating over time, as the language it was initially trained on diverges from the more recent language it is being applied to. *Temporal generalization* refers to a language model's ability to process and understand language from unseen data from different time periods.

In this chapter, I first explain the current static language modeling paradigm, and how this is reflected in the prevailing methods used in NLP to design and evaluate language models. Next, I outline how temporal generalization and performance deterioration are commonly measured. Furthermore, I discuss the shortcomings in measuring temporal generalization, and other problems that come into play when trying to solve these issues. Lastly, I address how this temporal generalization relates to semantic change. Table 3.1 summarizes the relevant terms and concepts.

Term	Definition
Temporally shifted data	Data that does not overlap in time with the training data of the language model
Temporal generalization	A language model's ability to generalize well to future data (Lazaridou et al., 2021)
Temporal misalignment	When a language model is trained on data from one time period and tested or deployed on data from another time period (Luu et al., 2022).
Temporal calibration	The ability to connect information to the appropriate time period (Dhingra et al., 2022)
Temporal adaptation	Retraining (or continually pre-training) with unlabelled data that mostly overlaps in time (Su et al., 2022)
Temporal performance deterioration	When a language model performance is lower when tested on temporally shifted data compared to data that overlaps in time with the language models training data (Lazaridou et al., 2021)
Domain adaptation	Retraining (or continually pre-training) with domain specific unlabelled data (Agarwal and Nenkova, 2022; Su et al., 2022)
Averaging	The phenomenon that when there are conflicting facts in the training/test data, the model becomes less certain of either fact (Dhingra et al., 2022)
Forgetting	When a model fails to memorize facts that were valid in periods of time that are underrepresented in the training data (Dhingra et al., 2022).
Static language modelling paradigm	The view that language is static, resulting in practices where language models are trained and tested on data from the same time period.

Table 3.1: Some key concepts and their definition

3.1 Static language modelling paradigm

Bender et al. (2021) first described the trend in NLP to tweak performance by only making language models larger and larger, "following an adage of 'there's no data like more data'". Rather than improving and carefully selecting the training data sets, state-of-the-art models are dominantly trained on more data. And rather than refining the model architectures, the models are enlarged with more parameters. To get an idea of this, table 3.2 displays some popular LLMs and their sizes. Min et al. (2021) have shown that success of LLMs can indeed be largely attributed to the tremendous increase in training data size. Increasing the size of the training corpora has a positive impact on model performances, but still they do not necessarily perform better on tasks containing out-of-domain texts (Min et al., 2021). At the same time, language models trained on diverse data sets are more likely to produce incorrect answers (Min et al., 2021). This puts into question to what extent the architectures and training objectives of state-of-the-art language models truly contribute significantly to the models success.

A consequence of the trend to use ever more data and parameters is that pre-training requires a lot of computational costs, which in turn have large financial and environmental costs (Bender et al., 2021; Biesialska et al., 2020). It is therefore a costly solution to pre-train regularly from scratch to surpass the problem of performance deterioration over time. This solution would also only undermine the idea and point of pre-training in the first place. In order to address these challenges, researchers are exploring new approaches to training LMs that are mode adaptive and responsive to change in language over time. These researchers advocate a shift from the static language modeling paradigm to one of *continual learning*, where models are trained to learn from a continuous stream of data (Biesialska et al., 2020).

The static language modeling paradigm is also apparent in the approaches used to evaluate language models. Luu et al. (2022) explain that models are commonly trained and tested on data from overlapping time periods. While in practice, models of NLP are first trained on data from one time period, whereafter they are used in real life applications that naturally concern data from later time periods. This phenomenon that LMs are trained on data from one time period, and applied to data from other time periods, is what Luu et al. (2022) call *temporal misalignment*. Luu et al. (2022) showed that temporal misalignment has strong effects on performance deterioration for eight downstream tasks. They also showed that temporal (domain) adaptation by continued pretraining can improve performance, but that this effect is rather small compared to task specific fine-tuning on data overlapping with the test period. Evaluating model performances on data overlapping with the training period follows a static language modeling paradigm because it is assumes that language does not change over time. It assumes that performance can be adequately measured by test data that overlaps in time with the training data, and that the performance would remain the same ("generalizes") when applied to temporally shifted data. This can result in misleading results, because the effect of temporal

misalignment is not tested.

Model	Class	Memory	# param.	Reference
ALBERT (base)	MLM	16 GB	12M	Lan et al. (2019)
ALBERT (large)	MLM	16 GB	18M	Lan et al. (2019)
BERT (base)	MLM	16 GB	108 M	Devlin et al. (2018)
BERT (large)	MLM	16 GB	334M	Devlin et al. (2018)
BART	Enc-Dec.	160 GB	$\sim$ 370M	Lewis et al. (2020)
GPT	Autoregressive	unk	117M	Radford et al. (2018)
GPT-2	Autoregressive	40 GB	1.542 B	Radford et al. (2019)
GPT-3	Autoregressive	unk	175B	Brown et al. (2020)
T5	Enc-Dec.	750GB	11B	Raffel et al. (2022)
RoBERTa	MLM	160 GB	340M	Liu et al. (2019)

Table 3.2: Statistics popular LLMs

3.2 Temporal calibration

Dhingra et al. (2022) observe that state-of-the-art language models are generally poor at connecting factual information to the time period it applies to. For example, the capital of Alaska will not likely change in the near future, while the president of the United States will likely change faster. However, a model does not take into account such information in downstream tasks like question answering. To a question like, 'what is the capital of Alaska?', the model may by chance output the correct answer, but not because the model has understanding of the temporal scope this fact applies to (Dhingra et al., 2022).

The ability to connect information to the appropriate time period is called *temporal calibration*. Language models are likely to be queried about facts outside the temporal scope of their training data. While it may seem undesirable for a model to guess the answer to questions about the future, in many cases it is perfectly reasonable to assume that the future will be like the present: for example, in twenty years the capital of Alaska is unlikely to change, while it is nearly impossible to predict who the governor of Alaska will be in twenty years from now. Ideally, the confidence with which the model responds to such queries should reflect this difficulty.

The failure of models at temporal calibration may result in problems like *averaging* and *forgetting* (Biesialska et al., 2020; Dhingra et al., 2022). Averaging can happen when language models are trained on conflicting information because facts change over time. When language modelling architectures ignore temporal metadata, this can lead to an averaging effect where the model has low confidence in any of the correct answers. *Forgetting* happens when a language model fails to memorize facts that were true in underrepresented periods of time, resulting in performance degradation when asked questions about the more distant past.

Understanding how facts relate to time can be seen as a prerequisite for

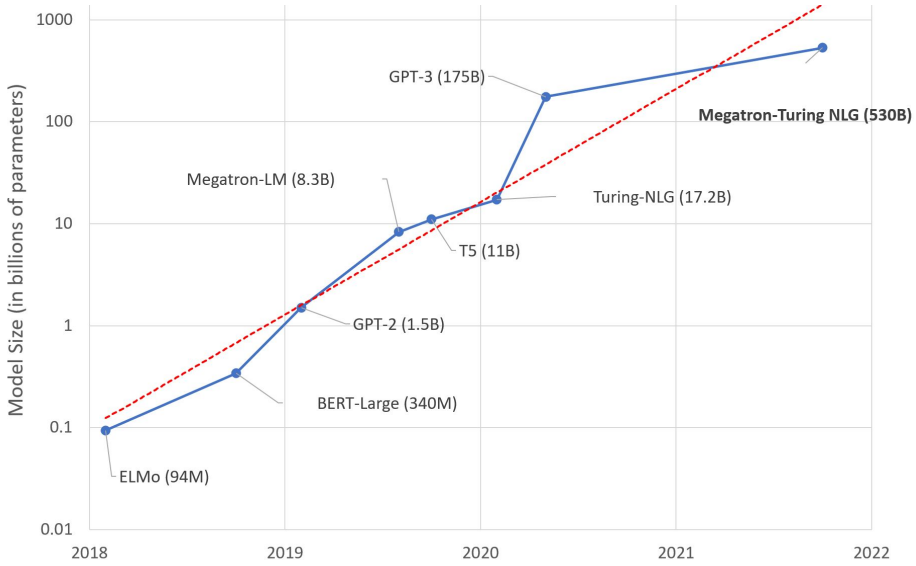

Figure 3.1: Number of parameters over time (from: https://huggingface.co/blog/large-language-models)

temporal generalization: if the aim is that the model should generalize its performance over time, it should be able to connect information to the correct time periods. If a language model performance degrades on question answering tasks, one may ask what causes the performance deterioration. Has the model not been exposed to the correct facts during training? Or is the model architecture simply not designed to connect information to the appropriate time periods? Surely, if the inability for temporal calibration impacts how LLMs process factual information, it is not unreasonable to expect that this inability also impacts how LLMs process semantic information.

A few attempts have been made to take the temporal dimension into account when training LLMs. Rosin et al. (2022) make an interesting attempt towards training "time-aware" language models. They propose to train what they call a 'temporal contextual language model', which uses the timestamp of a text as an additional context to the texts. This way, the model is not only trained to predict the text sequences based on the context words, but also on the time at which the text was written. They show a positive effect on the performance on the "sentence time prediction task" and also on semantic change detection.

Another attempt is of Loureiro et al. (2022a), who present a set of language models called TimeLMS that are specialized on diachronic twitter data. First, they pre-train a 'base' RoBERTa model on data up to 2019. Next, they continually train a new model from the base model every three months. The process of updating the base model follows the same procedure as the initial pre-training. While they do not actually make use of a continual learning approach, their

work allows the NLP community to use up-to-date LMs of any period of time, which can be useful to compare performance in the quest to alternatives to the current the static language modeling paradigm.

3.3 Performance deterioration

Lazaridou et al. (2021) describe *temporal generalization* as a model's ability to generalize well to future data from beyond their training period. By this they mean that the performance of a language model should remain consistent regardless of the time period it is tested on: if a model is capable of temporal generalization, performance should not deteriorate for data from beyond their pre-training period.

To inspect a LLM's capacity for temporal generalization, Lazaridou et al. (2021) measure the performance deterioration of a Transformer-XL over time on temporally shifted data. They measure the model's performances intrinsically by calculating the model's perplexity scores on texts from beyond the training period. The model's performances are also measured extrinsically for two downstream tasks: closed-book question answering and reading comprehension.

To measure the deterioration of a language model, Lazaridou et al. (2021) calculate the relative performance between two setups. In the `TIME-STRATIFIED`-setup the language model is trained on data until 2017 and tested on data from 2018-2019. The `CONTROL`-setup is trained and evaluated on text data from the overlapping time periods until 2019. The results of Lazaridou et al. (2021) show that relative perplexity increases most for (1) texts containing emerging new words that have rarely been used in the training period, (2) texts covering politics and sports, (3) proper nouns and numbers, and (4) open-class nouns. Another interesting result is that the model's performance on the closed-book question answering tasks decreases significantly over time, while the performance on the reading comprehension task remains the same.

Perplexity increase on texts containing emerging new words is not surprising, because emerging new words simply do not occur in the training period.[1] The perplexity increase on texts covering sports and politics, can be explained by the fact that such texts typically cover facts that are time-dependent. For example, who the president/world champion is of country X changes over time. This also explains why proper nouns and numbers yield higher perplexity scores, because the use of such words ('president Obama', 'population of 2 million') are typically time-dependent.

Each of the above results can be explained by the fact that the `CONTROL`-setup naturally lacks factual information on future data. Since the real world is changing, facts are changing as well, and these facts are reflected in the training data. This also explains why the performance deteriorates on the closed-book question answering task: the model simply has not been exposed to data containing the correct answer. Thus these results only show that the language

[1] About 27% of the unique words from the test period have never occurred in the training period.

model does not have access to the facts required to do the question answering task, or to predict future texts correctly. However, the results of Lazaridou et al. (2021) do not directly show that performance also deteriorates due to semantic language changes. Whether and how the performance is affected by semantic change will be addressed in section 3.4.

As mentioned in sections 2.3 and 3.3, it is common to measure performance of LLMs by examining perplexity scores, as Lazaridou et al. (2021) did as well. However, Röttger and Pierrehumbert (2021) point out that increased perplexity on texts does not necessarily imply that the model's performance on downstream tasks is truly deteriorating. They showed that even big changes in perplexity may lead to small changes in downstream task performance. The results of Lazaridou et al. (2021) on the reading comprehension task are an example of this: even though the `TIME-STRATIFIED`-setup showed increased perplexity scores, the performance on reading comprehension remained equally good. In this task, the language model has access to the information required to perform a task correctly. Apparently, increased perplexity scores do not impact the performance negatively in this case.

Agarwal and Nenkova (2022) also demonstrate that a language model's performance does not necessarily decrease for several downstream tasks.[2] They claim that the explanation for this is that the correctness labels of these tasks are not dependent on time. The only tasks for which significant performance deterioration was measured for all models was Domain Classification.

Increased perplexity scores do not give insight into *how* that language models are impacted on temporally shifted data, it merely shows *that* the models might be impacted. Lazaridou et al. (2021) do not elaborate on the cause of this perplexity increase. Is this increase merely a result of changing facts in the real world, that are not represented in the training corpora? Or does performance also deteriorate due to changes in language use, like semantic change, that is not reflected in the training corpora?

The fact that perplexity also increases for open-class nouns, could indicate that semantic change is also one of the causes of perplexity scores, since open-class nouns are typically more likely to change in meaning over time (Hamilton et al., 2016a).

3.4 Semantic change and performance deterioration

Several recent works have indicated that semantic change is closely related to temporal performance deterioration (Su et al., 2022; Ishihara et al., 2022). Su et al. (2022) examine the impact of semantically changing words on the performance of a language model, where performance is measured in terms of perplexity. They do so by showing that extra training on data containing semantically

[2]These tasks are: Named Entity Recognition, Truecasing, Sentiment Classification and Domain classification.

changed words, as opposed to just a random set of words, improves the performance of pre-trained language models significantly. Their method even yields performance improvement over domain adaptation methods on two different pre-trained language models and four data sets. The work of Su et al. (2022) indicates that language models suffer from performance deterioration due to lack of understanding of semantically shifted words. However, they only show this as a result of increase of perplexity scores. As discussed, perplexity scores are not necessarily an indication of performance deterioration on downstream language tasks. Therefore other evaluation metrics are needed to point out whether semantic change is a cause of performance deterioration on downstream tasks.

Ishihara et al. (2022) show a negative correlation between semantic change and a language model's perplexity for both the type-based Word2Vec model as well as the token-based RoBERTa. They show that a large time-series performance degradation occurs in the years when the so-called *semantic shift stability* is smaller. The degree of semantic shift is approximated by performing LSCD between Word2Vec models created from corpora of different time periods. A low degree of semantic shift between two time periods implies semantic shift stability between these time periods. They measure performance deterioration by comparing the performance of twelve different LMs varying by time-series. Again, they solely measure performance deterioration in terms of perplexity. Their approach to measuring semantic shift stability could support decision-making as to whether a model should be re-trained: when the semantic shift stability between two corpora is low, this should be an indication that the language model should be retrained, whereas high stability indicates that the language model performance would degrade less.

3.5 Conclusion

Performance deterioration occurs within a static language modelling paradigm where temporal misalignment between the training data and the test data is common. The performance of many state-of-the-art language models deteriorates on temporally shifted data, indicating that these models are incapable of temporal generalization. The incapability of temporal generalization in language models can be viewed from different angles. A model may lack factual knowledge of the world from time periods from beyond the training data, or a model may lack the ability of temporal calibration. There have also been indications that language model's lack of knowledge of semantically changed words can impact performance. However, the manner in which semantic change impacts the performance of language models on downstream tasks has not been researched as of yet.

So far, performance deterioration has been shown by comparing accuracies of different models on downstream tasks. These downstream tasks are not (directly) related to semantic change, but rather to factual and syntactic knowledge of a LLM. Performance deterioration is also commonly measured using perplexity scores, but as Röttger and Pierrehumbert (2021) pointed out,

increased perplexity does not necessarily indicate a performance deterioration on downstream tasks. Additionally, perplexity scores do not give insight into *how* the performance of LLMs degrades due to semantic change. So far, publications have focused on downstream tasks like named entity recognition and question answering, which primarily assess a model's capacity to process syntactic and factual information. This raises the question what the consequences are of temporal misalignment and of the static language modeling paradigm for using LLMs in downstream tasks. What happens in practice when LLMs are presented with temporally shifted data in which semantic change is present? Do the increased perplexity scores indeed mirror performance deterioration for data containing new language use? Or are LLMs capable of temporal generalization, and can they easily adapt to new uses of language?

These questions are exactly the gap that I aim to fill with this thesis. The main goal is to assess whether performance of LLMs is negatively impacted by semantic change and new language use. I especially aim to investigate this impact beyond perplexity scores, which have been the sole indicators of a relationship between semantic change and performance deterioration. This also allows to examine whether perplexity is a good indicator of performance deterioration on downstream tasks. I investigate this using the downstream task of contextualized word definition generation. This task not only allows to measure performance deterioration quantitatively through accuracy scores, but also provides human interpretable examples of how LLMs are impacted by language change when they are not up-to-date.

Chapter 4

Method

To assess whether performance of LLMs is negatively impacted by semantic change and new language use, I propose to use the downstream task of contextualized word definition generation. In particular, I examine the performance of `T5-base`, fine-tuned for the task of contextualized word definition generation, for a set of semantically stable target words, a set of semantically changing target words, and a set of emerging words. I first construct two corpora, such that the first overlaps in time with `T5-base`'s pre-training period, and the second consists of temporally shifted data. Next, I use a LSCD system to select (i) a set of changing words whose meaning has changed between the two time periods, and (ii) a set of stable words whose meaning has not changed between the two time periods. Third, I fine-tune `T5-base` for the task of contextualized definition generation, and test it on the sets of stable and changing target words. Moreover, I test the model on a set of *emerging words*. I collect human judgements to assess the correctness of the generated definitions. Performance deterioration of `T5-base` on the task of contextualized definition generation by comparing accuracy scores for each of the two time periods. This downstream task of definition generation not only enables to measure performance deterioration, but also allows for qualitative analysis of the generated content. This makes it possible to get human-interpretable insight into the implicit semantic information that the LLM represents of the words under investigation. Lastly, I conduct a qualitative analysis to get insight into the possible ways in which an outdated LLM fails at temporal generalization due to semantic change.

4.1 Model

The model under investigation is `T5` (Raffel et al., 2022). There are four reasons for investigating this particular model: (1) Since it is a sequence-to-sequence model, its training objectives are similar to that of definition generation models. (2) Huang et al. (2021) have proposed and evaluated a fine-tuning architecture for definition generation of `T5` which will be used in the experiments. (3) The

t_{start}	July 2015
t_{split}	May 2019
t_{end}	February 2023
C_1	corpus containing data between t_{start} - t_{split}
C_2	corpus containing data between t_{split} - t_{end}

Table 4.1: Terminology

time period from which the pre-training data originates is well documented.[1] (4) Since the model was published in 2019, and pre-trained on data until 2019, there exists enough temporally shifted data on which the model can be tested on temporal generalization. `T5` was pre-trained on the Colossal Clean Crawled Corpus (C4), which contains texts extracted from the web in April 2019 (Raffel et al., 2022).

4.2 Corpora

To detect whether and for which words semantic change has occurred between two time periods, LSCD systems require diachronic corpora C_1 and C_2 between which they can calculate semantic change scores. C_2 is collected such that consists of all documents published since t_{split} := May 2019, until t_{end} := February 2023. This covers data from 46 months in total. To keep the periods of both corpora equal, I set t_{start} := July 2015, such that C_1 also covers data of 46 months. The corpora are constructed from Tweets and Reddit posts and comments.

4.2.1 Twitter data set

To construct the Twitter data set, I use the pipeline developed by Loureiro et al. (2022a) to request Tweets using the academic Twitter API. They have published a pipeline that allows users to request tweets per month, filter out tweets by unauthorized users, and anonymify the user accounts. To request the most generic and random English tweets, they query tweets for English stop words. Loureiro et al. (2022b) retrieve 500 tweets for each hour of each day of each month. Due to licence constraints, I modified their sampling algorithm to only retrieve tweets for every four hours instead of every hour. This results in a total of 4,5 million tweets.

4.2.2 Reddit data set

To construct the Reddit data set, I use the Pushshift Reddit API (Baumgartner et al., 2020). For each day between July 2015 to February 2023, I request at most 500 posts and 500 comments. Only the posts and comments that consist of

[1] Most publications do not mention this explicitly, and it has been quite a hassle to find out.

at least 10 words, and contain at least one English stop word (following Loureiro et al. (2022a)) from `nltk.stopwords` are included. This results in roughly a million posts and comments.

4.2.3 Data pre-processing and cleaning

The documents are split in sentences using `nltk`'s `sent_tokenize`. The sentences from Reddit are tokenized into words using `nltk`'s `TreebankWordTokenizer`. The sentences from Twitter are tokenized using the `nltk.TweetTokenizer` (Loper and Bird, 2002), and emoji's are removed from the texts. Words are stripped from punctuation and made lower case.

4.3 Lexical Semantic Change Detection

To determine what words have undergone semantic change since t_{split}, I use the `SGNS+OP+CD` system,, because Schlechtweg et al. (2019) concluded that this system outperforms (many) other systems for LSCD tasks in diachronic corpora (see section 2.6). Moreover, since the LSCD system needs to be trained on data from both C_1 and C_2, it is relatively computationally inexpensive to train the LSCD from scratch. The LSCD system can be used to calculate semantic change scores for a set of requested terms of the shared vocabulary of C_1 and C_2. Therefore, I first select a set of *candidate target words* for which the semantic change score is computed. Following Schlechtweg et al. (2019), the vector space models for C_1 and C_2 are constructed using the SkipGram with Negative Sampling, using a window size of 10, $k = 5$ and $t = None$. The minimum corpus frequency `minCount`, the threshold that determines whether the model includes a term in the vector space model, is set at 30. I also exclude terms by the filter rule which excludes (i) URLs (containing the sub-strings `https://` or `http://`) (ii) emoji's and (iii) non-Latin characters (specified by ascii characters starting in `\U` or `\u`).

The vector spaces of both corpora are aligned using Orthogonal Procrustes. The change score of a *candidate target word* is quantified using the cosine distance (CD) between the target word embeddings from the two models. Additionally, I compute the local neighborhood distance (LND) between both time periods.

4.3.1 Target word selection

The performance of `T5` is examined for three categories of target words: (1) changing words, (2) stable words, and (3) emerging words.

Changing target words

I use a data-driven approach to select a set of candidate changing words, inspired by Chen et al. (2021), who use trending scores as indicators of semantic change.

They determine the so-called trending score of a term in the vocabulary as follows:

$$score(w) = \frac{f_{w,C_2} - f_{w,C_1}}{f_{w,C_2} + k}$$

where f_{w,C_i} frequency of the word in the month of the corpus C_1 where the frequency was highest. k is a normalization term used to mitigate the frequency of highly-frequent terms in the recent data sets. Since the `minCount` is 30, I also set $k = 30$. This entails that the trending score is positive whenever the frequency of a target word is at least twice as high in C_2 compared to C_1. Instead of using the raw corpus frequencies, I use the document frequency of each word. Document frequency is a better indication of the trendiness of a word than sole occurrence frequency, as it might be the case that a word is used very frequently in a small number of posts which would make it less trendy.

The trending score is calculated for all terms in the shared vocabulary of C_1 and C_2 which have an entry in the WordNet database (Fellbaum, 1998)[2], and do not contain any digits (e.g. a term like '2022' is excluded). The terms with a trending score of at least 1 are considered as `candidate target words`. The top-20 words with the highest CD scores are selected as `changing target words`, proper nouns and abbreviations excluded.

Stable target words

To collect a set of `candidate stable words` that have not undergone semantic change since t_{split}, I randomly select 1000 words from the shared vocabulary of C_1 and C_2 that (i) have a WordNet entry, (ii) have a document frequency of at least `minCOunt` in both C_1 and C_2, and (iii) do not contain any digits. Next, I use the LSCD system to determine the CD scores for these candidate stable words. I select the set of `stable target words` such that the CD scores are both below 0.25. From this list, excluding proper nouns and abbreviations, 20 target words are randomly selected as `stable target words`.

Emerging target words

I collect a set of `emerging target words` that either (i) have a document frequency in C_2 of at least 50, while having a document frequency of 0 in C_1, or (ii) have a document frequency that is at least five times as much in C_2 compared to C_1. This resulted in a total of 1585 emerging words. Since newly emerging words are likely not present in the WordNet database, I manually select 20 words that (i) do not contain any digits, (ii) are not named entities (i.e. places, persons, brands) (iii) are not abbreviations.

[2] following Su et al. (2022)

4.4 Contextualized definition generation

Fine-tuning `T5` for the task of contextualized definition generation requires a data set containing ground truth data of words, usage examples, and their corresponding context-specific definitions. To fine-tune `T5` for definition generation, I use the architecture proposed by Huang et al. (2021).

4.4.1 Definitions data set

The Oxford data set is used to fine-tune `T5-base` for contextualized definition generation. The Oxford data set consists of triplets of target words, usage examples, and corresponding definitions from the Oxford Dictionary, collected by Gadetsky et al. (2018). These definitions and usage examples were written and picked out by experts, and are in British English.

The usage examples from the Oxford data set are picked out and verified by experts, which inherently differs from the experiments of this thesis, which uses a data-driven approach to collect example sentences for target words. Since longer sentences are more likely contain context relevant to the target word that short sentences, it is desirable to train the models on a data set that also has on average longer example sentences. Therefore I use the Oxford data set opposed to other obvious choices like WordNet; the usage examples of Oxford consist of 16.73 ± 9.53 words, while the WordNet usage examples consist of only 4.80 ± 3.43 words (Giulianelli et al., 2023). Moreover, the Oxford data set is a reasonable decision because it yields best NIST[3] score in Huang et al. (2021).

4.4.2 Fine-tuning `T5`

To fine-tune `T5-base` for contextualized definition generation, I follow the method proposed by Huang et al. (2021). To generate contextualized definitions, they fine-tune three `T5` models. The first model, `T5-base`, is fine-tuned to generate the n-best definitions for a given target word and context. The second model, `T5-specific` is meant as a specificity estimator. It is fine-tuned to generate a local context for a given target word $w*$, conditioned on the generated definition by the base model `T5-base`. The third model, `T5-general`, is meant to re-rank the n generated definitions for appropriate generality. This model is fine-tuned to generate a definition conditioned on a target without a local context.

The three models `T5-base`, `T5-specific` and `T5-general` are fine-tuned using the standard method proposed by Raffel et al. (2022). As input, the target words and corresponding context sentences are concatenated with the labels 'word: ...' and 'context: ...'. This concatenated string is then prompted to the encoder of `T5`, after sub-word segmentation by SentencePiece (Kudo and

[3]The NIST is evaluation metric measures the quality of text which is originates from in machine translation. The NIST score computes the n-gram precision for the generated text compared to a ground-truth text, weighed by the informativeness of the particular n-gram (Huang et al., 2021).

Richardson, 2018). The models are trained to minimize the cross-entropy loss for the generated output.

For each of the n generated definitions, the so-called *generation likelihood* $\mathcal{P}_{T5}$ is used to re-rank the n generated definitions. The generation likelihood of the definition D generated for the word $w*$ and the context C is defined as follows:

$$\mathcal{P}_{T5} := -\log(\mathbb{P}(D|C, w*)$$

`T5-specific` is used to estimate the over-specificity of the generated definitions. The specificity score $\mathcal{P}_{specific}$ is defined as

$$\mathcal{P}_{specific} := -\log \mathbb{P}(C|D)$$

`T5-general` is used to estimate the definition's under-specificity. The under-specificity score $\mathcal{P}_{general}$ is defined as:

$$\mathcal{P}_{general} := -\log \mathbb{P}(D|w*)$$

Finally, three metrics $\mathcal{P}_{T5}, \mathcal{P}_{specific}$ and $\mathcal{P}_{general}$ are used to re-rank the n-best definitions generated by `T5-base`, using a simple linear combination of these scores:

$$r = \alpha\mathcal{P}_{specific} + \beta\mathcal{P}_{general} + (1 - \alpha - \beta)\mathcal{P}_{general} \tag{4.1}$$

where $\alpha, \beta \in [0, 1] \subset \mathbb{R}$. The values of α and β are selected such that they yield the highest performance on the validation set. These values for α and β are then applied to the test set to select the final output definition. This pipeline, will henceforth be referred to as `T5-base-DG`.

4.4.3 Generating definitions for the target words

For each of the 60 target words (20 changing, 20 stable, 20 emerging), at most 100 example sentences were randomly sampled from each corpus. For some of the target words, either $C1$ or $C2$ did not contain 100 usages of the target word. Since the `minCount` was set at 30, each target word has at least 30 example sentences. For each example sentence and target word pair, a contextualized definition was generated using `T5-base-DG`.

Since the example usages for each target word are randomly sampled, the quality of the example sentences cannot be guaranteed. In contrast, the example usages from the Oxford data set on which the model was fine-tuned, were hand-picked by expert lexicographers. It can be the case that some of the example sentences are simply not informative enough so that the exact word sense can be deduced from the sentence. Another consequence of randomly sampling four sentences for evaluation, is that the set of sampled sentences may not display new usages of the target words, but "old" usages. Suppose that target word w has obtained an extra sense since t_{split}, it could in practice be the case that each of the four sampled sentences from C_2 correspond to the old sense of the target word.

4.5 Evaluation

`T5-base` is evaluated extrinsically on the definition generation task, and intrinsically using cross-entropy loss, perplexity, and pseudo-log likelihood.

4.5.1 Human evaluation

Since there are no ground truths available for the definitions corresponding to the example usages, human annotation is required to judge the correctness of the generated definitions for a given target word and example usage. To reduce the number of required annotations, I randomly sample 4 usage examples for each target word in each corpus. This results in a total of 160 instances of stable target words, 160 instances of changing target words, and 80 instances of emerging words (only from C_2); 400 in total.

Three human annotators (fluent English speakers) were asked to evaluate the correctness of the generated definitions. The annotators were presented with a total of 400 (word, example sentence, definition) triplets. For each triplet, the annotators judge correctness of the generated definition on a graded scale of $\in \{0, 1, 2, 3\} \in \mathbb{N}$, where score **3** corresponds to a completely correct definition, and **0** corresponds to an incorrect definition. A correct definition is defined as truthful and fluent. An additional special case for incorrect definitions is that of self-reference, in which case the annotators could assign the value **-10**. A definition is self-referring whenever it uses the target word itself to define the target word. The judgements were aggregated via majority vote, where the labels **-10, 0, 1** are considered *incorrect*, and the labels **2, 3** are considered *correct*. Full annotation guidelines can be found in appendix A.

The inter-rater agreement is measured using Krippendorff's α coefficient, which is a statistical measure used to assess the agreement or reliability among multiple raters or annotators when assigning categorical or ordinal values to items. It measures the extent of agreement beyond what would be expected by chance. The coefficient ranges from 0 to 1, with higher values indicating greater agreement.

4.5.2 Measuring performance deterioration

Using the human judgements as ground truths, I calculate the accuracy of `T5-base-DG` for the task of contextualized definition generation. Comparing the accuracies for the sentences from C_1 to the sentences from C_2, allows to measure performance deterioration. Recall that performance deterioration is the decline or decrease in the effectiveness or accuracy for a task of a LLMs over time. Thus the model `T5-base-DG` suffers from performance deterioration whenever its accuracy for the sentences from C_2 is lower than for the sentences from C_1. Furthermore, to get insight into how semantic change and emerging language use impacts performance, I calculate the accuracy scores for each category of target words (stable, changing, emerging).

Hypothesis 1 Since previous works have shown that LLMs suffer from performance deterioration over time (see 3.3), the first hypothesis is that the accuracy on the definition generation task will be higher for the example sentences from C_1 than for the example sentences from C_2.

Hypothesis 2 Since previous studies have indicated that there is a correlation between semantic change and performance deterioration (see 3.4), the second hypothesis is that this difference in accuracy between C_1 and C_2 is more prevalent for changing target words than for stable target words.

Hypothesis 3 Since emerging words are words that have newly entered or rapidly increased in frequency in C_2, it is likely that the `T5-base` has not been exposed to many training instances containing the emerging target word, making it more difficult to generate adequate definitions for them. The hypothesis is that accuracy on the emerging target words from C_2 will be lower than for the stable target words of C_2.

4.6 Intrinsic evaluation

For intrinsic evaluation, I compute the cross-entropy loss (2.1) and perplexity scores (2.2) of `T5-base` for each of the example usages of each target word. I also calculate the pseudo-log likelihood (2.4) for each of the 400 annotated example usages. [4]

To get an indication how the appearance of a target word in the context sentence contributes to the sentence perplexity, I also calculate the cross entropy loss for the masked-word-prediction task:

$$\textsc{Loss}(w, c) := -\log \mathbb{P}(w_t | S_{\setminus w_t}) \tag{4.2}$$

This is computed by replacing the target word in the sentence with the special `<extra_id_1>` mask token, and computing the model's cross-entropy loss for predicting the target word in that position.

To assess whether perplexity scores of the input example sentences and performance on the downstream task of definition generation correlate, I calculate the correlation between the example sentence perplexity scores and the correctness of the (corresponding) generated definitions. I also calculate the correlation between the perplexity scores and the originating corpus.

Both cases concern a correlation between a numerical variable (perplexity scores) and a categorical variable (the correctness judgement / the corpus label). I calculate the *Point-Biserial correlation coefficient* r_{pb}, which is used when you have a dichotomous (two-level) categorical variable and a continuous variable. It measures the association between the two variables, and is computed by:

[4] Due to computational constraints, this is not done for the total of 8524 example usages for which the definitions are generated., but only for the 400 evaluated sentences.

$$r_{pb} = \frac{M_1 - M_0}{\sigma} \sqrt{\frac{N_1 N_0}{N(N-1)}} \tag{4.3}$$

Where M_1 and M_0 are the means of the continuous variable for the two groups defined by the binary variable (in this case, correctness label or the corpus label), σ is the standard deviation of the continuous variable, N_1 and N_0 are the sizes of the two groups defined by the binary variable, and N is the total sample size.

Spearman's correlation is a non-parametric measure that assesses the strength and direction of the monotonic relationship between variables, regardless of their specific distributions. Spearman's correlation is computed by:

$$\rho = 1 - \frac{6 \sum d_i^2}{n(n^2 - 1)} \tag{4.4}$$

Where d_i represents the difference between the ranks of each pair of observations, and n is the number of observations.

4.7 Qualitative analysis

Examining the generated definitions may give insight into the possible mistakes that a LLM can make in a generative task like contextualized definition generation when it is not up-to-date. Questions to consider when examining the generated output are:

- In case the generated definition is incorrect for a semantically changing word, does the output possibly display language use that is related to older usages of the target word?
- In case the generated definition is incorrect, does the output display language use that is related (either semantically or conceptually) to the input context sentence, or is the output unrelated?
- What kind of definitions are generated for target words which are neologisms? Since these words likely don't occur in the pre-training data, the model likely has not been exposed to prior information about these words. Do the generated definitions give insight into what information the model uses to generate definitions to these new words?
- Do the input sentences of the highest perplexity/cross-entropy loss yield correct or incorrect definitions? If not, do the generated definitions give insight that can help explain why the generated definitions are (in)correct?
- Do the generated definitions exhibit any form of bias?

Chapter 5

Results

To gain insight into how `T5-base` is affected by semantic change, I examined its performance via definition generation for a set of context sentences containing stable, changing, and emerging words. To collect these sentences, I created two corpora C_1 and C_2, such that C_1 overlaps in time with the model's training period, and C_2 is temporally shifted. With help of a LSCD system, I collected sentences containing semantically stable, semantically changing, and emerging words. The performance on the downstream task of definition generation should point out whether the model suffers from performance deterioration. Moreover, to get an insight into how semantic change influences performance deterioration, I compare the performance on the sentences containing semantically changed words and to those containing semantically stable words. The results are compared with the cross entropy loss and the perplexity scores of `T5-base`. This can confirm whether increased cross entropy loss and perplexity scores indicate performance deterioration of LLMs over time on the task of contextualized definition generation. Lastly, I conducted a qualitative analysis on the generated definitions themselves, to gain insight into how semantic change and new language use impact the behavior of a LLM on a generative task like definition generation in practice.

5.1 Target words

A total of 401 words from the shared vocabulary of C_1 and C_2 had a trending score of at least 1, and were considered *candidate changing words*. The top-20 trending words can be seen in table 5.1 below. The top-20 trending words with the highest CD change score, excluding abbreviations and proper nouns[1], are selected as *changing words*. The lowest change score of the top-20 changing target words was 0.72.

Many of the top trending words did not necessarily correspond to the highest cosine distance change scores. Of the top-20 trending words, only six had a

[1]These were: `crt, moa, lh, atp, cro, erica, burrow, hancock`

change score above 0.7. Of the 401 trending words, 30% (total 122) had a change score above 0.5 and 18% had a change score above 0.6 (total 72).[2] This shows that trendiness is an indicator of semantic change, but not a guarantee.[3]

Of the 1000 randomly selected words, 254 had a CD as well as LND scores below 0.25. Of these, 20 were randomly selected, which can be viewed in table 5.2. Let us get an impression of the kind of target words that appear in the experiments.

5.1.1 Changing target words

	Trending Word	CD	LND		Target Word	CD	LND
1	pandemic	0.90	0.47	1	corona	0.98	0.55
2	quarantine	0.88	0.34	2	lockdown	0.96	0.23
3	vaccine	0.45	0.02	3	manifesting	0.92	0.06
4	vaccinated	0.53	0.03	4	closeness	0.91	0.56
5	lockdown	0.96	0.23	5	pandemic	0.90	0.47
6	moots	0.37	0.09	6	quarantine	0.88	0.34
7	corona	0.98	0.55	7	navigator	0.86	0.26
8	distancing	0.83	0.31	8	distancing	0.83	0.31
9	vaccination	0.54	0.04	9	ape	0.81	0.21
10	virus	0.52	0.03	10	checkmate	0.79	0.26
11	airdrop	0.52	0.28	11	masking	0.78	0.20
12	yacht	0.72	0.28	12	peacock	0.78	0.19
13	masks	0.60	0.21	13	polygon	0.76	0.00
14	ukraine	0.35	0.01	14	anchor	0.75	0.39
15	vaccines	0.42	0.01	15	shanks	0.74	0.11
16	mandates	0.60	0.19	16	tracing	0.73	0.15
17	ukrainian	0.43	0.03	17	pinks	0.72	0.36
18	doge	0.51	0.06	18	moot	0.72	0.46
19	staking	0.61	0.13	19	hag	0.72	0.39
20	bodybuilding	0.68	0.24	20	yacht	0.72	0.28

Table 5.1: Top-20 trending (left) and changing words (right)

COVID-19 Many of the changing target words were related to the COVID-19 outbreak, e.g. CORONA, LOCKDOWN, PANDEMIC, QUARANTINE and DISTANCING. This is not surprising, as the COVID-19 outbreak happened by the end of 2019, which started after the pre-training period of `T5-base`. In C_1, *corona* was for instance used to refer to a Mexican beer brand, a or to city in the US, and has probably come to predominantly be used to refer to a virus since the

[2] opposed to 0.11% and 2.6% respectively for all words in the WordNet vocabulary.

[3] This puts into question the approach of Loureiro et al. (2022b), who only use trending scores as estimator of semantic shift.

COVID-19 outbreak. Before the pandemic, the word DISTANCING was probably primarily used to describe the act of establishing distance to other people from an emotional motivation, whereas in C_2 it had started to be used to describe the act of establishing distance between other people in order to not infect them.

Manifesting Another example of the emergence of a new sense is the word MANIFESTING. This word has likely received a high semantic change score because of the emergence of a new sense: a definition was added to the Urban Dictionary on December 6^{th} of 2020[4], defining it as 'a term used by subliminal users meaning to hope for a desire until it comes true using the law of attraction'. In C_1, MANIFESTING was probably used as simply the present participle of the verb TO MANIFEST, which describes the process of making something visible or apparent (OED, 2023c).

Checkmate Unlike CORONA, and MANIFESTING, the word CHECKMATE does probably not owe its high change score to emergence of a sense, but rather to the (re)domination of an already existing sense. This is likely due to the renewed popularity of the game of chess in recent years.[5] In C_1, the word CHECKMATE was used literally in the context of a chess game, as well as metaphorically to express a situation where someone or something is in a position of power or advantage over another, for instance in contexts concerning sports, arguments or debates. Thanks to the increasing popularity of the game of chess, the literal sense of CHECKMATE was probably more frequent C_2.

Some of the top-20 changing target words have likely received a high change score because of a growing prevalence of the word due to cultural events: POLYGON likely owes its high change score to the increasing use of the word to refer to an online gaming platform; SHANKS likely owes its high change score as it is used to refer to a Japanese Manga character in C_2; HAG owes its semantic change score as it's frequently used to refer to the Dutch football coach *Eric Ten Hag*: 40% of the sentences in C_2 contain the n-gram TEN HAG, compared to less than 1% in C_1.

5.1.2 Emerging target words

Like for the changing target words, many of the emerging target words concern covid-related words: COVID, COVIDIOTS, PLANDEMIC, VAXED, COVID, SPREADER, VAX, ANTI-VAX. Other emerging words are related to gender identity, such as NON-BINARY, FEMBOY, SAPPHIC, TRANSPHOBE. A particularly interesting emerging word is GOATED, which is an example of grammatization from the noun GOAT, which was initially an abbreviation for GREATEST OF ALL TIME. Emerging words that originate from blends are COPIUM (COPE +

[4]See https://www.urbandictionary.com/define.php?term=manifesting

[5]See https://www.chess.com/blog/CHESScom/chess-is-booming-and-our-servers-are-struggling

	Word	CD Score	LND Score		Emerging word
1	look	0.12	0.00	1	copium
2	lose	0.12	0.00	2	covidiots
3	player	0.13	0.00	3	plandemic
4	morning	0.14	0.01	4	vaxed
5	population	0.17	0.00	5	gatekeeping
6	option	0.17	0.00	6	grifting
7	idea	0.17	0.00	7	gaslight
8	settings	0.18	0.00	8	non-binary
9	opinions	0.18	0.00	9	femboy
10	statement	0.19	0.00	10	quarantining
11	families	0.20	0.00	11	covid
12	realise	0.20	0.00	12	transphobe
13	community	0.22	0.01	13	simp
14	asparagus	0.22	0.00	14	wokeness
15	art	0.22	0.00	15	sapphic
16	talks	0.22	0.00	16	spreader
17	beginning	0.22	0.00	17	goated
18	outcome	0.22	0.00	18	k-pop
19	groceries	0.22	0.02	19	vax
20	performance	0.22	0.02	20	anti-vax

Table 5.2: Stable and emerging target words

OPIUM), COVIDIOTS (COVID + IDIOT), and PLANDEMIC (PLAN + PANDEMIC). The word GASLIGHT is an example of a neologism that has resulted from the concatenation of two already existing words. Other notable examples not selected as target words, include the acronym PROD for PRODUCT, and the abbreviation IMA for I'M GOING TO.

5.2 Definition generation

The performance on the contextualized word definition task is evaluated by human judgements for a total 400 randomly sampled sentences for each target word. Krippendorff's α inter-rater agreement is 0.62.[6] First, I discuss the resulting accuracies for the task and whether these show performance deterioration. Next, the correctness of the generated definitions for each example sentence is compared with the perplexity scores for the corresponding example sentences. I discuss some examples to get insight into what kind of definitions `T5-base-DG` constructs for the temporally shifted sentences containing semantically changed and emerging words.

[6] And 0.68 if we reduce the labels to four, mapping the -10 judgement to 0; incorrect.

5.2.1 Performance deterioration

The accuracies on the definition generation task per category can be viewed in table 5.3. The performance deterioration of `T5-base-DG` between C_1 and C_2 is 19%, which indicates that `T5-base-DG` fails at temporal generalization. This confirms hypothesis 1 (see 4.5.2). The performance deterioration is especially present for semantically changed words, with a decrease in performance of 36.7%, compared to 7.5% for the stable target words, which confirms hypothesis 2. Still, performance deterioration is also present for sentences of stable words. This could indicate that performance can deteriorate over time even when semantic change is not present. Another possible explanation is that the context words, rather than stable target words, have undergone semantic change, which could also impact the performance. As expected, the performance on the task of contextualized definition generation in C_2 is also tremendously worse for the emerging words compared to the stable and changing words, confirming hypothesis 3. These results also persist when aggregating the judgements by consensus voting, which can be found in Appendix A.3.

Unexpectedly, the accuracy for the sentences containing changing words in C_1 compared to those of stable words in C_1, is also substantially lower. This will be further elaborated on in the discussion. The drastic differences in performance for the emerging and changing words compared to the stable words, indicates that semantic change and performance deterioration are related. Moreover, the accuracy decrease between C_1 and C_2 in each of the categories implies that that `T5-base-DG` largely fails at temporal generalization. The accuracy scores compare with those of Huang et al. (2021), who report an accuracy of 45%.[7]

Category	C_1	C_2	$C_1 \cup C_2$
stable	**66.25%**	**61.25%**	**63.75%**
changing	**37.5%**	23.75%	30.625%
stable + changing	**52.5%**	42.5%	47.5%
emerging	-	8.75%	8.75%
total	**52.5%**	31.25%	-

Table 5.3: Accuracy on the contextualized definition generation task

[7]Notably, Huang et al. (2021) annotate the definitions using eight different categories: (1) over-specified, (2) self-reference, (3) wrong part-of-speech, (4) under-specified, (5) opposite, (6) similar semantics, (7) incorrect, and (8) correct. They do not specify how they aggregate the human judgements.

5.3 Perplexity scores

Perplexity v.s. corpus Table 5.4 shows the average cross entropy loss and perplexity of the 8524 sampled sentences, and the average cross entropy loss for masked target word prediction in the example sentences. For simplicity sake, let STABLE SENTENCES, CHANGING SENTENCES and EMERGING SENTENCES denote the example sentences containing stable target words, changing target words, and emerging target words respectively. On average, the scores are highest for the STABLE SENTENCES of C_1. Figure 5.1 displays how the scores are distributed in each category. Table 5.5 shows that a significant negative correlation was measured between the perplexity scores and the time period. However, this correlation of -0.02 is considered extremely weak.

Category	Corpus	Sentence `CE loss`	Sentence `PPL`	Word prediction `CE loss`
stable	C_1	**0.76**	**2.40**	11.34
	C_2	0.73	2.30	**11.69**
changing	C_1	0.69	2.18	10.69
	C_2	0.70	2.19	11.40
emerging	C_2	0.69	2.24	10.82

Table 5.4: Average scores

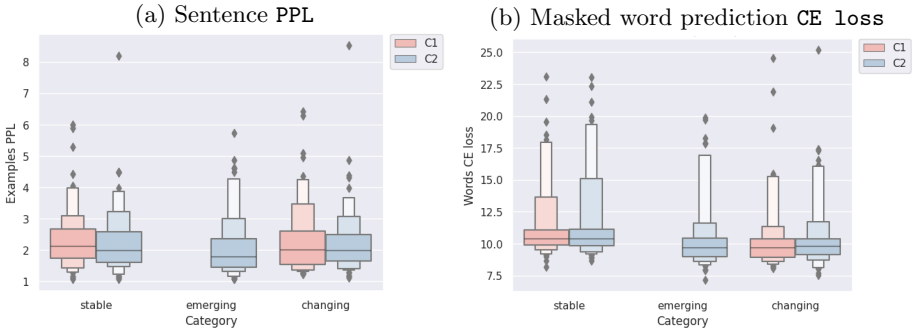

Figure 5.1: Scores per corpus

	Point Biserial	Spearman
Word prediction `CE loss`	0.03 (0.002)	**-0.02 (0.088)**
Sentences `CE loss`	-0.03 (0.006)	-0.02 (0.045)
Sentences `PPL`	-0.03 (0.007)	-0.02 (0.045)

Table 5.5: Correlations between the corpus and each scoring type

Perplexity v.s. performance Figure 5.2 displays the score distributions for the annotated 400 example sentences per category and judged correctness. The figure also displays the pseudo-log likelihood scores (2.4) for the example sentences.

For simplicity sake, let the (IN)CORRECT SENTENCES denote the 'example sentences for which the definition generation model has generated definitions judged (in)correctly'. The plot shows that the CORRECT SENTENCES on average indeed have lower perplexities. However, some of the sentences with the highest perplexity and pseudo-log-likelihoods still yielded correct definitions. We can see that the model has generated correct definitions for sentences whose perplexity is relatively high. These examples will be further discussed in section 5.3.1.

Table 5.6 shows the correlation scores between the correctness of the definitions and the cross entropy loss and perplexity for the corresponding example sentences. A significant negative correlation is observed between the sentence perplexity and the correctness of the generated definition. These correlations of -0.04, -0.03 and -0.02 are considered very weak.

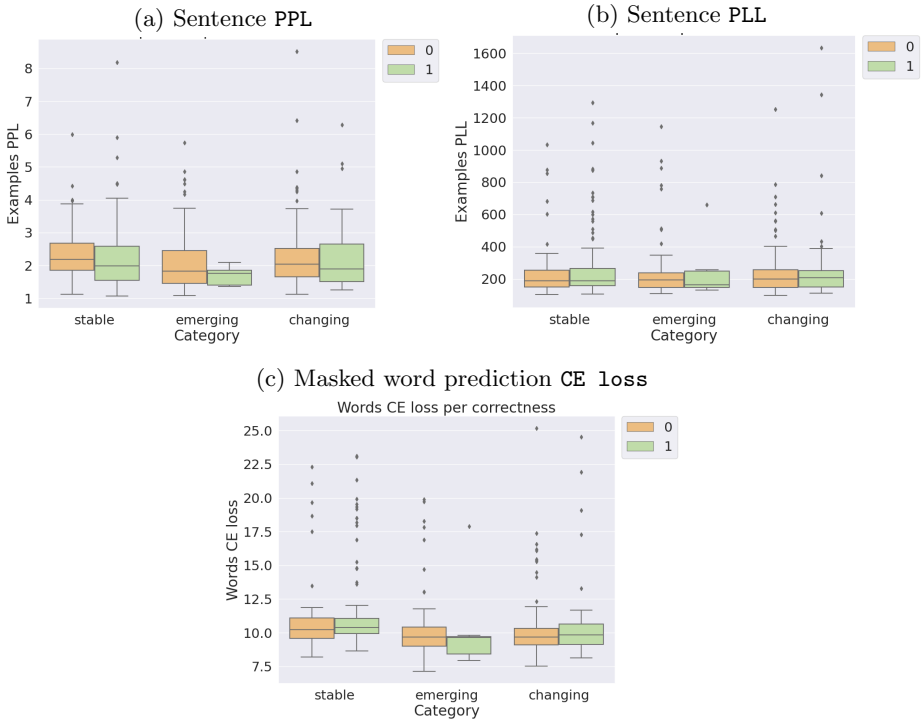

Figure 5.2: Scores per correctness label

	Point Biserial	Spearman
Word prediction cross-entropy loss	0.13 (0.009)	0.14 (0.000)
Sentences cross-entropy loss	-0.03 (0.551)	-0.04 (0.336)
Sentences PPL	-0.02 (0.752)	-0.04 (0.336)

Table 5.6: Correlations between correctness and each scoring type

	Target	LOOK
a)	**Example** C_1	in heroes and generals there you have limited sight as gunner but you can *look* out to see more with the danger of getting shot.
	Definition	expect something to happen or be the case
	Judgement	2, 2, 3
b)	**Example** C_2	*looks* like russia had exposed to the world that they are not a military might.
	Definition	have the appearance of being
	Judgement	3, 3, 3

Table 5.7: Correct definitions for the word LOOK

5.3.1 Qualitative Analysis

First, I go over some examples that illustrate how `T5-base-DG` is capable of generating context-specific definitions for the words and contexts of interest. Second, I discuss some examples where `T5-base-DG` fails to generate sufficiently correct contextualized definitions. Third, I illustrate how the incorrect definitions for the changing and emerging target words reflect a LLMs lack of temporal generalization. Lastly, I discuss some of the sentences with high perplexity scores, and illustrate that these high scores do not necessarily imply performance deterioration for the downstream task of contextualized word definition generation.

Stable target word *look* The examples of the stable target word LOOK in table 5.7 show that `T5-base-DG` is capable of generating context-specific definitions. In *a*), the word LOOK is used in the context of LOOK OUT TO SEE..., for which the generated definition is "expect something to happen or be the case". A different use of the word LOOK is displayed in *b*), as it's used in the context 'looks like...', for which the generated definition is 'have the appearance of being'. Both definitions are judged to be correct by all three annotators.

Stable target word *option* In example *c*) for the stable word *option*, we see that `T5-base-DG` has predicted a remarkably precise, fluent and correct definition, namely "an item in a list or other collection of data displayed on

	Target	OPTION
	Example C_2	selecting the aa *option* is now in loadout screen
c)		as opposed to being a modification.
	Definition	an item in a list or other collection of data displayed on the screen which can be selected
	Judgement	3, 3, 3
	Example C_2	also if an address is required then is there another
d)		*option* other than providing my address?
	Definition	an idea or way of doing something
	Judgement	1, 1, 1

Table 5.8: Examples of the stable word OPTION

	Target	LOCKDOWN
	Example C_1	imagine being on *lockdown* up until you get
e)		to college ... boom !
	Definition	a state of isolation or seclusion by the police
	Judgement	3, 3, 2
	Example C_2	this *lockdown* needs to end foreal my son is so bored
f)		and lonely now he is demanding for a baby sister ...
	Definition	a period of isolation or separation from other people
	Judgement	3, 2, 3

Table 5.9: Correct definitions for LOCKDOWN

the screen which can be selected". Example *d*) does show that a different sense of the word OPTION is detected in a different context. However, in *d*), the definition 'an idea or way of doing something' is not judged sufficiently correct by the annotators.

Changing target word lockdown Table 5.9 shows two examples of correctly generated definitions for the changing target word LOCKDOWN. These are 'a state of isolation or seclusion by the police' for the example sentence of C_1, and 'a period of isolation or separation from other people' for the example sentence of C_2. Table 5.10 shows two examples of *incorrectly* generated definitions for LOCKDOWN. In example *e*), the definition 'an act of closing something down' is judged insufficient by two of the three annotators. An explanation for this is that the word LOCKDOWN does not refer to an *act*, but rather a situation. Example *f*) even displays a definition for a word sense which is completely unrelated, 'a withdrawal of troops and supplies from the area where they are locked up'.

Table 5.11 shows three more changing target words and their generated definitions. These changing target words CORONA, MANIFESTING, and POLYGON

	Target	LOCKDOWN
g)	**Example** C_1	in the middle of lecture my prof says get the hell out of here now there 's going to be a *lockdown* in 10 minutes
	Definition	an act of closing something down
	Judgement	1, 1, 3
h)	**Example** C_2	if we don't enter a substantial *lockdown* in the next 48h we should all be v worried .
	Definition	a withdrawal of troops and supplies from the area where they are locked up
	Judgement	0, 0, 0

Table 5.10: Incorrect definitions for LOCKDOWN

have each obtained new usages in C_2. The generated definitions are by and large incorrect for C_1 as well as C_2. These three target words illustrate three different phenomena.

Changing target word corona CORONA illustrates that `T5-base-DG` generates incorrect definitions for sentences from C_1 both and C_2, but that the incorrectness of these definitions have a different basis. Namely, the definitions for C_1 do display knowledge of semantic relatedness to the intended sense of the target word, while the definitions for sentences of C_2 reflect completely unrelated semantic properties. CORONA in C_1 concerns three example sentences in which it refers to a Mexican beer, and one usage where it refers to the city of Corona ('I live in corona...'). In each of the cases, the generated definitions are semantically related to the intended sense of the target word, but untruthful. The definitions for the example sentences of C_2 show relatively divergent definitions. Except for 'a cigar', these definitions share some (indirect) semantic similarity with some of the words used in the context sentences for which they were generated. The definition 'a cigar' is likely generated because CORONA is also a brand of Havana cigar (OED, 2023a), but in this was not the correct sense in this particular context.

Changing target word manifesting MANIFESTING is an example of a changing target word where `T5-base-DG` does generate a correct definition for a sentence in C_2, while this sentence displays a usage of the (relatively) new sense of the target word: the definition 'making a public mention of something' may be an instance of successful temporal generalization, as the definition is a proper description of the newly emerged word sense of MANIFESTING.

Changing target word polygon The definitions for the target word POLYGON illustrate another type of mistake that `T5-base-DG` makes when presented with semantic change in temporally shifted data. The word POLYGON has a

high semantic change score because its new sense refers to an online gaming platform. The generated definitions for C_2 reflect semantic similarity to the original sense of the word, like 'more than three dimensional elements', or 'many-dimensional'. Other definitions also show some semantic similarity with the online entertainment website, using words like SYSTEM and COMPUTER, but the generated definitions are not truthful.

	CORONA C_1	Correct?
1.	a cocktail made with aromatic spices and fruit juice	0
2.	a deep red or yellowish-brown colour	0
3.	a cold drink served with drinks such as fruit or vegetables	0
4.	a small lake or valley	0
	CORONA C_2	
1.	the identification of a kite or other mammal by its markings and colours	0
2.	a cigar	0
3.	a divinely conferred blessing or beneficence	0
4.	a strain of arnoviruses found in many tropical and subtropical areas	0
	MANIFESTING C_1	
1.	appearing in the body as a sign of illness or injury	1
2.	exhibiting or making visible signs	1
3.	(in whistleblowing) an arrangement of the hands to show someone who has been playing	0
4.	the action or sound of a manifesto	0
	MANIFESTING C_2	
1.	perceptible by the word of god	0
2.	the action of revealing something	0
3.	making a public mention of something	1
4.	the action of clearly showing someone or something	0
	POLYGON C_1	
1.	a solid or cylindrical object having at least three straight sides and angles	1
2.	more than three dimensional parts or elements	0
3.	a very large number or amount	0
4.	a word or phrase used by several people	0
	POLYGON C_2	
1.	a three-dimensional recreation in which players use two or more lines to move around one another	0
2.	many-dimensional	0
3.	denoting a conceptual system in which data is represented by two or more discrete units	0
4.	a computer graphic or display device that supports several different configurations	0

Table 5.11: Changing target words and their generated definitions

Emerging target words Out of the 80 examples of emerging words, only 7 of the generated definitions were judged to be correct. Two of these are for GASLIGHT, and two are for ANTI-VAX.

The two correctly generated definitions for ANTI-VAX are (1) 'antipathy or aversion to vax', and (2) 'a person who has no vaccinations or is actively anti-viral'. The first definition is correct, as it is fluent and factual. Consider however that the model incorrectly defines VAX in all cases, with definitions like 'a disease caused by an infection of the vagina', and 'ask for or obtain as a vaex'. In contrast, definition (2) is surprisingly correct, apart from the fact that the term 'anti-viral' is slightly ambiguous. The correctness of this definition can be explained by the informativeness of the example sentence that was provided, which was: 'swagenknecht okay go ahead you call this guy anti-vax because he is not vaccinated!'. This example sentence is largely a definitial sentence itself, as it explicitly states why a person is anti-vax. Incorrectly generated for ANTI-VAX are (3) 'exaggerated or anti-vox', and (4) 'hostile or obnoxious'.

Correct definitions for GASLIGHT were (1) 'the light of a gaslamp', and (2) 'manipulate (someone) by psychological means into doubting their feelings'. Both definitions refer to a different sense of the word GASLIGHT; the first being the traditional use of GASLIGHT, while the latter corresponds to the emerged sense, which is defined correctly by the generated definition. Contrary to ANTI-VAX, the example sentences of GASLIGHT are not as informative that the model can copy the definition from the example sentence. An explanation for this is that the term GASLIGHT, and its corresponding emerging sense of 'manipulate (someone) by psychological means into doubting their feelings' is not completely new, as it originates from the British theater play 'Gas Light' of 1938, and was added to the Urban Dictionary in 2009[8]. This makes it likely that this sense of GASLIGHT was already used in the pre-training corpus of `T5-base`.

More examples of incorrectly generated definitions can be viewed in table 5.12 below. These expose some other common blunders that `T5-base-DG` makes. Firstly, many of the definitions for the emerging words, `T5-base-DG` itself produces definitions with non-existing new words as well. This was the case for VAEX when defining VAX, for ANTI-VOX when definition ANTI-VAX, PLANDELIA and PLANDISONE when defining PLANDEMIC, and A-FEMBOY to define FEMBOY.

Second, many of the emerging words trigger some weak or strong form of self reference. This seems to happen more than for the stable target words. This was the case for GATEKEEPING, SPREADER, PLANDEMIC, NON-BINARY, FEMBOY, WOKENESS and QUARANTINING.

Third, some of the (incorrect) generated definitions reflect an implicit polarity (positivity or negativity) towards the target word. This polarity seems to be inferred from the provided context (the example sentence). For instance, the generated definitions for the word SIMP are considerably negative: 'a weak or ineffectual person', 'a stupid or contemptible person', 'a servile or impudent

[8] See: https://www.urbandictionary.com/define.php?term=Gaslighting and https://www.washingtonpost.com/wellness/2022/04/15/gaslighting-definition-relationship-abuse-response/

woman', and 'an impudent or insincere man'. In fact, according to the online dictionary, the definition of SIMP is: 'a slang insult for men who are seen as too attentive and submissive to women, especially out of a failed hope of winning some entitled sexual attention or activity from them'.[9] Likewise, FEMBOY is defined as 'a lame or mischievous person', while in fact, it means 'a young, usually cisgender male who displays traditionally feminine characteristics'.[10] Thus, the model does catch on to the negative polarity of the context, however, attributes incorrect qualities to the word. Likewise, the definitions for COVID reflect negative ('a term of abuse') or positive ('a term of endearment') connotations, depending on the sentiment of the input context. Arguably, these definitions display bias towards the emerging words.

What about perplexity? No strong correlations were observed between the cross entropy loss and (pseudo-)perplexity scores. Neither between the time periods from which the sentences originate, nor with the correctness of the generated definitions. How so? The example sentence with the highest perplexity score is for the target word CLOSENESS: '@user closeness to yt supremacy is hella of a drug'. The generated definition for this target word, 'the quality of being attentive or lenient', is incorrect. Indeed, this example sentence does contain two out of vocabulary words: YT and HELLA, which may explain the high sentence perplexity.[11] The second highest perplexity was for the sentence: 'achieve this when you go to settings and disable these'. This sentence, however, does not contain any unknown words, and the generated definition for the stable target word SETTINGS is correct: 'the specified configuration of a computer or other electronic device'. For the masked-word-prediction-loss, 3 of the top-5 sentences with the highest scores were in fact correct. This shows that even the highest (pseudo-)perplexity scores are not indicators of the failure of temporal generalization.

[9] https://www.dictionary.com/e/slang/simp/

[10] https://www.dictionary.com/e/gender-sexuality/femboy/; https://www.urbandictionary.com/define.php?term=femboy&page=9

[11] YT is an acronym for WHITE, and HELLA, which is an acronym for HELL OF.

	COVID
1.	used as a general term of abuse
2.	divergence from sex in the sexual activity of women
3.	used as a term of endearment
4.	an entertaining or amusing person
	K-POP
1.	denoting a category of words in radio and television programmes that are intended to attract attention
2.	pop music or dance to a popular song of australian origin
3.	a style of popular music intended for people who are secretly seeking to attract attention
4.	relating to or denoting unrestrained folk music of us black origin
	FEMBOY
1.	a showy or frivolous woman
2.	a-femboy
3.	a person who shares popular misconceptions
4.	a lame or mischievous person
	COVIDIOTS
1.	any of the old world scottish precociously elected officers and pensioners
2.	a person who behaves in an unfriendly and cowardly manner
3.	a person who believes that their tastes or behaviour are superior to those of other people
4.	a person who is secretly willing to obey others
	PLANDEMIC
1.	of or relating to plandelia
2.	an outbreak of a plan demic
3.	a period of plandisone
4.	an act of spreading plandisone

Table 5.12: Emerging target words and (incorrectly) generated definitions

Chapter 6

Discussion

Performance deterioration The results show a performance deterioration of 19.2% for `T5-base-DG` on the task of contextualized definition generation. The accuracy for the task deteriorates drastically more for changing target words (36.7%) compared to the stable target words (7.5%). Furthermore, the accuracy of 8.7% on the task for the emerging words was exceptionally low. This indicates that `T5-base-DG` fails to generalize its capacity to generate context-specific definitions for a target word given a context over time. The results also strongly indicate semantic and lexical change are causes for the performance deterioration of `T5-base-DG` In turn, this could imply The three hypotheses described in 4.5.2 are confirmed.

A striking observation is that the performance on the changing target words in C_1 is also substantially worse than for the stable words in C_1. I initially expected that the accuracy for the definition generation task on the changing target words in C_1 would be comparable to the accuracy on the stable target words in C_1. So why is the accuracy considerably low for the changing target words in C_1? One possible explanation is that these changing words were *already* unstable in the time period of C_1. This aligns with knowledge that words undergoing semantic change typically go through polysemous stages before a dominant sense is established, and *the Law of Innovation*. These results could also be explained by the fact that the changing words were pre-selected according to their trending score. Recall that the trending score reflects the ratio between the frequencies of C_1 and C_2. It is likely that some of the trending words were relatively infrequent in C_1 compared to an average word in the English vocabulary. As a consequence, the pre-trained model `T5-base` may have been exposed to fewer training instances for these trending words in the first place, resulting in their representations to be of lower quality. This explanation that the semantically changed words are relatively infrequent would align well with the *Law of Conformity*, that frequent words tend to change more slowly than infrequent words. To confirm this idea, it would be interesting to examine whether changing words that were substantially more frequent before t_{split} correspond to higher accuracy for the definition generation task for C_1. Fu-

ture work should point out the performance deterioration on the semantically changing words already occurs even before the end of the pre-training period.

Perplexity Comparisons between cross-entropy loss and perplexity of `T5-base` show that these are in general not accurate indicators of an LLM's performance on the downstream task of definition generation. Some examples showed relatively high losses that did not impact the performance of `T5-base-DG`, and vice versa. This aligns with the findings of Röttger and Pierrehumbert (2021) who indeed state that high perplexity does not necessarily correspond to lower downstream task performance. This does put into question the legitimacy of related studies such as (Lazaridou et al., 2021) and Ishihara et al. (2022), as they largely base their conclusions on perplexity increase. Hence, this shows that developers of LLMs for downstream tasks should not rely on perplexity scores when reporting performance deterioration.

Generated definitions When examining the content of the generated definitions, multiple distinct consequences of the failure of temporal generalization can be identified. In some cases, `T5-base-DG` outputs content that is either semantically similar to the original word sense when presented with usages that did not exist prior. In other cases, `T5-base-DG` may output content in which too much of the provided context is adapted, resulting in overspecific and untruthful definitions. In the case of emerging words, `T5-base-DG` is more likely to output (1) sentences containing new neologisms, (2) content that relates to the polarity that the context conveys, and (3) self-referential language use. Moreover, some of the generated definitions display different kinds of bias towards certain groups or perspectives. This was specifically observed for the emerging words. However, a more in-depth and systematic approach is needed to determine whether such biases are actually characteristic of language change and temporal generalization. Interestingly, most generated definitions were fluent, while the factual information that they convey is incorrect. In some cases, a non-careful reader may be deceived into thinking it's a proper definition; this is what Mickus et al. (2019) call hallucination. This is important if we consider the real-life application of a definition generation model: users would use it to look up the meaning of words which they do not know yet. Naturally, users would not be able to verify whether the generated definitions are correct - otherwise, they would not need the tool in the first place. If it is indeed the case that temporally shifted data used as input for LLMs can result in incorrect or biased output, this could potentially have harmful effects on the hegemonic structures in societies.

Limitations of the method Importantly, there are some caveats to the method used. Firstly, the contexts provided for the definition generation model are not hand-picked by experts but are selected in a data-driven way. This approach risks the selection of low-quality contexts. The sentences can be of low quality in multiple ways; they may exhibit incorrect or affluent language

use, or they may exhibit insufficient information for a model, or a human for that matter, to be able to infer the intended word sense in the specific context. This might sabotage the performance of the definition generation model. Still, this shortcoming applies to each category of target words, stable as well as changing and emerging. Evaluating the model on more instances could reduce the risk of skewing results due to low-quality definitions. Secondly, the method of randomly selecting example sentences from the corpora C_1 and C_2 does not guarantee that the sample of sentences from C_2 truly exhibits instances of semantic change. It could well be the case that four sentences were sampled from C_2 which are actually "old" or "traditional" usages of that target word, rather than its new or shifted sense. To address this issue, future work could consider applying additional heuristics to select new usages of the target words. Thirdly, the experiments in this thesis were only conducted on the `T5-base` model and for one specific fine-tuning architecture. It could be the case that other pre-trained LLMs do not show as much performance deterioration on the task of contextualized definition generation. Future work should point out whether the results of this thesis generalize to other LLMs.

General remarks Overall, contextualized definition generation seems a promising task to assess a model's capacity for temporal generalization when semantic change and lexical change are present. This task could not only be used to assess whether performance deterioration is present but could, of course, also be used to test whether an LLM *is* capable of temporal generalization. For instance, this task could be used to examine how effective temporal domain adaptation is for keeping an LLM's semantic knowledge, rather than their factual knowledge, up-to-date. The results showed that the pre-trained LLM `T5-base`, fine-tuned for definition generation, is bad at temporal generalization, and that this is likely due to the semantic change that is present in the temporally shifted data. These results only concern the generalization of the capacity for generating contextualized definitions. Still, the fact that the model fails to generalize its capacity to process semantic information onto future data does raise concerns about its capacity to generalize different but related capabilities. Surely, if generative LLMs struggle more to generate accurate definitions for input in which semantic change is present, it is likely that other generative LLMs, fine-tuned for generative applications such as text summarization and chatbots, are also impacted by semantic change in temporally shifted data. Hence, if models of similar applications fail comparably at temporal generalization, this could have a real impact on human knowledge and, with this, on society.

Chapter 7

Conclusion

In this thesis, I proposed and experimented with a method to assess the capacity for temporal generalization via the downstream task of definition generation. So far, the capacity of LLMs for temporal generalization has been assessed using downstream tasks that primarily assess a model's capacity to process factual and syntactic information in texts. Definition generation, on the other hand, assesses more directly how a model processes target words semantically. Assessing LLMs via definition generation also allows for an intuitive and human interpretable output that helps see what the possible consequences are in practice when a model fails at temporal generalization. This work is the first to use the task of contextualized definition generation to assess how semantic change influences temporal generalization of LLMs.

For this thesis, I showed that `T5-base`, fine-tuned for definition generation, indeed suffers from performance deterioration, especially when semantic or lexical change is present in the temporally shifted test data. Another interesting result is that the performance on either time periods is worse for semantically changing words, indicating that semantic change impacts performance even when a model is tested on data overlapping with the pre-training period. I also collected two corpora of Reddit and Twitter data. All data and code will be made publicly available.[1]

Future work should point out whether these findings also apply to other LLMs and other downstream tasks. Furthermore, I confirmed that high perplexity is not a reliable indicator of performance deterioration on downstream tasks. My results underline the importance of designing adequate, task-specific methods that can assess whether an LLM deteriorates over time. Qualitative analysis demonstrates the potential of outdated models to produce biased output. Future work should point out whether models indeed produce more biased output in temporally misaligned setups. Overall, definition generation can be a promising task to assess a model's capacity for temporal generalization with respect to semantic and lexical change.

[1] https://github.com/IrisLuden/Thesis-TemporalGeneralization

Bibliography

Agarwal, O. and Nenkova, A. (2022). Temporal effects on pre-trained models for language processing tasks. *Transactions of the Association for Computational Linguistics*, 10:904–921. https://doi.org/10.1162/tacl_a_00497.

Almeida, F. and Xexéo, G. (2019). Word Embeddings: A Survey. *CoRR*, abs/1901.09069. http://arxiv.org/abs/1901.09069.

Baumgartner, J., Zannettou, S., Keegan, B., Squire, M., and Blackburn, J. (2020). The Pushshift Reddit Dataset. *Proceedings of the International AAAI Conference on Web and Social Media*, 14:830–839. https://doi.org/10.1609/icwsm.v14i1.7347.

Bender, E. M., Gebru, T., McMillan-Major, A., and Shmitchell, S. (2021). On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 610–623. https://doi.org/10.1145/3442188.3445922.

Biesialska, M., Biesialska, K., and Costa-jussà, M. R. (2020). Continual Lifelong Learning in Natural Language Processing: A Survey. *CoRR*, abs/2012.09823. https://arxiv.org/abs/2012.09823.

Boleda, G. (2020). Distributional Semantics and Linguistic Theory. *Annual Review of Linguistics*, 6(1):213–234. https://doi.org/10.1146/annurev-linguistics-011619-030303.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. (2020). Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901. https://ailab-ua.github.io/courses/resources/GPT3_Brown_2020.pdf.

Chalmers, D. J. (2002). On Sense and Intension. *Philosophical Perspectives*, 16:135–182. http://www.jstor.org/stable/3840912.

Chen, S., Neves, L., and Solorio, T. (2021). Mitigating Temporal-Drift: A simple approach to keep NER models crisp. In *Proceedings of the Ninth*

International Workshop on Natural Language Processing for Social Media, page 163–169, Online. Association for Computational Linguistics. https://aclanthology.org/2021.socialnlp-1.14.

Davies, M. (2012). Expanding horizons in historical linguistics with the 400-million word Corpus of Historical American English. *Corpora*, 7(2):121–157.

Del Tredici, M., Fernández, R., and Boleda, G. (2019). Short-Term Meaning Shift: A Distributional Exploration. In *Proceedings of NAACL-HLT*, pages 2069–2075. https://aclanthology.org/N19-1210.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv e-prints*, page arXiv:1810.04805. https://ui.adsabs.harvard.edu/abs/2018arXiv181004805D.

Dhingra, B., Cole, J. R., Eisenschlos, J. M., Gillick, D., Eisenstein, J., and Cohen, W. W. (2022). Time-aware language models as temporal knowledge bases. *Transactions of the Association for Computational Linguistics*, 10:257–273. https://doi.org/10.1162/tacl_a_00459.

Dubossarsky, H., Hengchen, S., Tahmasebi, N., and Schlechtweg, D. (2019). Time-Out: Temporal referencing for robust modeling of lexical semantic change. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, page 457–470, Florence, Italy. Association for Computational Linguistics. https://aclanthology.org/P19-1044.

Eger, S. and Mehler, A. (2017). On the linearity of semantic change: Investigating meaning variation via dynamic graph models. *CoRR*, abs/1704.02497. http://arxiv.org/abs/1704.02497.

Eisenstein, J. (2019). *Introduction to Natural Language Processing*. MIT Press. Google-Books-ID: 72yuDwAAQBAJ, https://books.google.nl/books?hl=en&lr=&id=72yuDwAAQBAJ&oi=fnd&pg=PR5&ots=gWcPZ64lq-&sig=lbuZOy2XFX18wIUA89-U7h5zsYs&redir_esc=y#v=onepage&q&f=false.

Fellbaum, C. (1998). *WordNet: An Electronic Lexical Database*. Bradford Books. https://mitpress.mit.edu/9780262561167/.

Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., and Ruppin, E. (2001). Placing search in context: the concept revisited. In *Proceedings of the 10th international conference on World Wide Web*, page 406–414, Hong Kong Hong Kong. ACM. https://dl.acm.org/doi/10.1145/371920.372094.

Gadetsky, A., Yakubovskiy, I., and Vetrov, D. (2018). Conditional Generators of Words Definitions. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 266–271. https://aclanthology.org/P18-2043/.

Giulianelli, M., Del Tredici, M., and Fernández, R. (2020). Analysing Lexical Semantic Change with Contextualised Word Representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, page 3960–3973. http://arxiv.org/abs/2004.14118.

Giulianelli, M., Kutuzov, A., and Pivovarova, L. (2022). Do not fire the linguist: Grammatical profiles help language models detect semantic change. In *Proceedings of the 3rd Workshop on Computational Approaches to Historical Language Change*. The Association for Computational Linguistics. http://arxiv.org/abs/2204.05717.

Giulianelli, M., Luden, I., Fernandez, R., and Kutuzov, A. (2023). Interpretable word sense representations via definition generation: The case of semantic change analysis. *arXiv preprint arXiv:2305.11993*. https://arxiv.org/pdf/2305.11993.pdf.

Gulordava, K. and Baroni, M. (2011). A distributional similarity approach to the detection of semantic change in the Google Books Ngram corpus. In *Proceedings of the GEMS 2011 workshop on geometrical models of natural language semantics*, pages 67–71. https://aclanthology.org/W11-2508.pdf.

Hamilton, W. L., Leskovec, J., and Jurafsky, D. (2016a). Cultural Shift or Linguistic Drift? Comparing Two Computational Measures of Semantic Change. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2116–2121, Austin, Texas. Association for Computational Linguistics. https://aclanthology.org/D16-1229.

Hamilton, W. L., Leskovec, J., and Jurafsky, D. (2016b). Diachronic word embeddings reveal statistical laws of semantic change. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1489–1501. https://aclanthology.org/P16-1141.pdf.

Huang, H., Kajiwara, T., and Arase, Y. (2021). Definition Modelling for Appropriate Specificity. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2499–2509, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics. https://aclanthology.org/2021.emnlp-main.194.

Ishihara, S., Takahashi, H., and Shirai, H. (2022). Semantic Shift Stability: Efficient Way to Detect Performance Degradation of Word Embeddings and Pre-trained Language Models. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing*, page 205–216, Online only. Association for Computational Linguistics. https://aclanthology.org/2022.aacl-main.17.

Jang, J., Ye, S., Lee, C., Yang, S., Shin, J., Han, J., Kim, G., and Seo, M. (2022). TemporalWiki: A Lifelong Benchmark for Training and Evaluating Ever-Evolving Language Models. *arXiv e-prints*, pages arXiv–2204. arXiv:2204.14211 [cs], http://arxiv.org/abs/2204.14211.

Jatowt, A. and Duh, K. (2014). A framework for analyzing semantic change of words across time. In *IEEE/ACM Joint Conference on Digital Libraries*, pages 229–238. IEEE. https://ieeexplore.ieee.org/abstract/document/6970173.

Jurgens, D., Mohammad, S., Turney, P., and Holyoak, K. (2012). SemEval-2012 Task 2: Measuring Degrees of Relational Similarity. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, page 356–364, Montréal, Canada. Association for Computational Linguistics. https://aclanthology.org/S12-1047.

Kilgarriff, A. (1997). I don't believe in word senses. *Computers and the Humanities*, 31(2):91–113. https://doi.org/10.1023/A:1000583911091.

Kim, Y., Chiu, Y., Hanaki, K., Hegde, D., and Petrov, S. (2014). Temporal Analysis of Language through Neural Language Models. *CoRR*, abs/1405.3515. http://arxiv.org/abs/1405.3515.

Kudo, T. and Richardson, J. (2018). SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *EMNLP 2018*, page 66.

Kulkarni, V., Al-Rfou, R., Perozzi, B., and Skiena, S. (2015). Statistically significant detection of linguistic change. In *Proceedings of the 24th International Conference on World Wide Web*, WWW '15, page 625–635, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee. https://doi.org/10.1145/2736277.2741627.

Kurtyigit, S., Park, M., Schlechtweg, D., Kuhn, J., and Schulte im Walde, S. (2021). Lexical Semantic Change Discovery. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6985–6998, Online. Association for Computational Linguistics. https://aclanthology.org/2021.acl-long.543.

Kutuzov, A., Øvrelid, L., Szymanski, T., and Velldal, E. (2018). Diachronic word embeddings and semantic shifts: a survey. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1384–1397. https://doi.org/10.48550/arXiv.1806.03537.

Kutuzov, A., Velldal, E., and Øvrelid, L. (2022). Contextualized embeddings for semantic change detection: Lessons learned. *Northern European Journal*

of Language Technology, 8(11). https://nejlt.ep.liu.se/article/view/3478.

Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., and Soricut, R. (2019). ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. https://arxiv.org/abs/1909.11942.

Lazaridou, A., Kuncoro, A., Gribovskaya, E., Agrawal, D., Liska, A., Terzi, T., Gimenez, M., de Masson d'Autume, C., Kocisky, T., Ruder, S., et al. (2021). Mind the gap: Assessing temporal generalization in neural language models. *Advances in Neural Information Processing Systems*, 34:29348–29363. https://proceedings.neurips.cc/paper/2021/hash/f5bf0ba0a17ef18f9607774722f5698c-Abstract.html.

Lehrer, A. (2003). Understanding trendy neologisms. *Italian Journal of Linguistics*, 15:369–382.

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., and Zettlemoyer, L. (2020). BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics. https://aclanthology.org/2020.acl-main.703.

Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., and Neubig, G. (2023). Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. *ACM Computing Surveys*, 55(9):195:1–195:35.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv e-prints*, page arXiv:1907.11692. https://ui.adsabs.harvard.edu/abs/2019arXiv190711692L.

Loper, E. and Bird, S. (2002). NLTK: The Natural Language Toolkit. In *Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics - Volume 1*, ETMTNLP '02, page 63–70, USA. Association for Computational Linguistics. https://doi.org/10.3115/1118108.1118117.

Loureiro, D., Barbieri, F., Neves, L., Espinosa Anke, L., and Camacho-collados, J. (2022a). TimeLMs: Diachronic language models from Twitter. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 251–260, Dublin, Ireland. Association for Computational Linguistics. https://aclanthology.org/2022.acl-demo.25.

Loureiro, D., D'Souza, A., Muhajab, A. N., White, I. A., Wong, G., Espinosa-Anke, L., Neves, L., Barbieri, F., and Camacho-Collados, J. (2022b). TempoWiC: An evaluation benchmark for detecting meaning shift in social media.

In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 3353–3359, Gyeongju, Republic of Korea. International Committee on Computational Linguistics. https://aclanthology.org/2022.coling-1.296.

Luu, K., Khashabi, D., Gururangan, S., Mandyam, K., and Smith, N. A. (2022). Time Waits for No One! Analysis and Challenges of Temporal Misalignment. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5944–5958, Seattle, United States. Association for Computational Linguistics. https://aclanthology.org/2022.naacl-main.435.

Miaschi, A. and Dell'Orletta, F. (2020). Contextual and Non-Contextual Word Embeddings: an in-depth Linguistic Investigation. In *Proceedings of the 5th Workshop on Representation Learning for NLP*, page 110–119, Online. Association for Computational Linguistics. https://aclanthology.org/2020.repl4nlp-1.15.

Mickus, T., Paperno, D., and Constant, M. (2019). Mark my Word: A Sequence-to-Sequence Approach to Definition Modeling. *DL4NLP 2019*, page 1. https://aclanthology.org/W19-6201.

Mikolov, T., Chen, K., Corrado, G. S., and Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. http://arxiv.org/abs/1301.3781.

Min, B., Ross, H., Sulem, E., Veyseh, A. P. B., Nguyen, T. H., Sainz, O., Agirre, E., Heintz, I., and Roth, D. (2021). Recent Advances in Natural Language Processing via Large Pre-Trained Language Models: A Survey. *CoRR*, abs/2111.01243. https://arxiv.org/abs/2111.01243.

Noraset, T., Liang, C., Birnbaum, L., and Downey, D. (2017). Definition modeling: Learning to define word embeddings in natural language. In *Thirty-First AAAI Conference on Artificial Intelligence*. https://dl.acm.org/doi/abs/10.5555/3298023.3298042.

OED (2023a). corona, n.2. In *OED Online*. Oxford University Press. https://www.oed.com/view/Entry/41772?rskey=RYacYY&result=2#eid Accessed 15 May 2023.

OED (2023b). hacker, n. In *OED Online*. Oxford University Press. https://www.oed.com/view/Entry/83045?rskey=TdieDR&result=1 Accessed 7 May 2023.

OED (2023c). manifest, v. In *OED Online*. Oxford University Press. https://www.oed.com/view/Entry/113484?rskey=hwp8XR&result=1& Accessed 15 May 2023.

OED (2023d). viral, adj. In *OED Online*. Oxford University Press. https://www.oed.com/view/Entry/223706?redirectedFrom=viral, Accessed 5 May 2023.

OED (2023e). woke, adj.2. In *OED Online*. Oxford University Press. https://www.oed.com/view/Entry/58068747?rskey=uIbWEi&result=3&isAdvanced=false#eid Accessed 7 May 2023.

Osborne, M., Lall, A., and Van Durme, B. (2014). Exponential reservoir sampling for streaming language models. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 687–692. http://aclanthology.lst.uni-saarland.de/P14-2112/.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543. https://aclanthology.org/D14-1162.pdf.

Qiu, X., Sun, T., Xu, Y., Shao, Y., Dai, N., and Huang, X. (2020). Pre-trained models for natural language processing: A survey. *Science China Technological Sciences*, 63(10):1872–1897.

Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., et al. (2018). Improving language understanding by generative pre-training. https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al. (2019). Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9. https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. (2022). Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *J. Mach. Learn. Res.*, 21(1). https://dl.acm.org/doi/abs/10.5555/3455716.3455856.

Ranjan, N., Mundada, K., Phaltane, K., and Ahmad, S. (2016). A Survey on Techniques in NLP. *International Journal of Computer Applications*, 134(8):6–9. https://doi.org/10.5120/ijca2016907355.

Rosin, G. D., Guy, I., and Radinsky, K. (2022). Time Masking for Temporal Language Models. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*, WSDM '22, page 833–841, New York, NY, USA. Association for Computing Machinery. https://doi.org/10.1145/3488560.3498529.

Röttger, P. and Pierrehumbert, J. (2021). Temporal Adaptation of BERT and Performance on Downstream Document Classification: Insights from Social Media. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2400–2412, Punta Cana, Dominican Republic. Association for Computational Linguistics. https://aclanthology.org/2021.findings-emnlp.206.

Schlechtweg, D., Hätty, A., Del Tredici, M., and Schulte im Walde, S. (2019). A Wind of Change: Detecting and Evaluating Lexical Semantic Change across Times and Domains. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 732–746, Florence, Italy. Association for Computational Linguistics. https://aclanthology.org/P19-1072.

Schlechtweg, D., Tahmasebi, N., Hengchen, S., Dubossarsky, H., and McGillivray, B. (2021). DWUG: A large resource of diachronic word usage graphs in four languages. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7079–7091, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics. https://aclanthology.org/2021.emnlp-main.567.

Su, Z., Tang, Z., Guan, X., Li, J., Wu, L., and Zhang, M. (2022). Improving Temporal Generalization of Pre-trained Language Models with Lexical Semantic Change. *arXiv e-prints*, pages arXiv–2210. arXiv:2210.17127 [cs], http://arxiv.org/abs/2210.17127.

Sun, T.-X., Liu, X.-Y., Qiu, X.-P., and Huang, X.-J. (2022). Paradigm Shift in Natural Language Processing. *Machine Intelligence Research*, 19(3):169–183. https://doi.org/10.1007/s11633-022-1331-6.

Tahmasebi, N., Borin, L., and Jatowt, A. (2021). *Survey of computational approaches to lexical semantic change detection*, page 1–91. Language Science Press, Berlin. https://zenodo.org/record/5040302.

Traugott, E. C. (2017). Semantic change. In *Oxford research encyclopedia of linguistics*. https://doi.org/10.1093/acrefore/9780199384655.013.323.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. (2017). Attention is All you Need. *Advances in Neural Information Processing Systems*, 30. https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Wang, A., Pruksachatkun, Y., Nangia, N., Singh, A., Michael, J., Hill, F., Levy, O., and Bowman, S. (2019). SuperGLUE: A stickier benchmark for general-purpose language understanding systems. *Advances in neural information processing systems*, 32. https://proceedings.neurips.cc/paper_files/paper/2019/file/4496bf24afe7fab6f046bf4923da8de6-Paper.pdf.

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 *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, page 353–355, Brussels, Belgium. Association for Computational Linguistics. http://aclweb.org/anthology/W18-5446.

Wang, S., Zhou, W., and Jiang, C. (2020). A survey of word embeddings based on deep learning. *Computing*, 102(3):717–740. https://doi.org/10.1007/s00607-019-00768-7.

Zamora-Reina, F. D., Bravo-Marquez, F., and Schlechtweg, D. (2022). LSCDiscovery: A shared task on semantic change discovery and detection in Spanish. In *Proceedings of the 3rd Workshop on Computational Approaches to Historical Language Change*, page 149–164, Dublin, Ireland. Association for Computational Linguistics. https://aclanthology.org/2022.lchange-1.16.

Appendix A

Appendix

A.1 Annotation guidelines

You are provided with a spreadsheet with four columns: **Targets**, **Judgement**, **Example** and **Definition**. In every row, there is one English target word in the **Target** column, one example sentence in which this target word is used in the **Example** column, and one definition sentence or phrase in the **Definition** column. The definition has been generated by a large language model and it is a context-specific definition for the target word in the example sentence.

Words can have different meanings, depending on the context in which they are used. The possible meanings that a word in different contexts can have are called *senses*. A popular example is the word **bank**, which is a polysemous word:

> Sentence 1: I need to get some money from the *bank*
> Sentence 2: I'm walking along the river *bank*.

In sentence 1, the sense of the target word *bank* can be defined as "An institution that invests money deposited by customers or subscribers". In sentence 2, on the other hand, the target word *bank* refers to the sense that can be defined as "the sloping, vertical, or overhanging edge of a river or other watercourse".

Your task is to judge for each row whether the definition of the target word in the example sentence is correct. That is, the definitions must be:

- **Truthful**: i.e. should reflect exactly the sense in which the target word is occurring in the example sentence. Ideally, the definition should be specific enough so as not to mix with other senses, while general enough so as not to describe information of the example sentence that does not concern the target word.
- **Fluent** i.e., feeling like natural English sentence or phrase, without grammar errors, utterances broken mid-word, etc.

Task instructions

You have to fill in the **Judgements** column with one of five values:

0: The definition is incorrect; not truthful or not fluent

1: The definition is partially incorrect; it is either not truthful or not fluent, but it does reflect some information related to the sense of the target word in the example sentence

2: The definition is mostly correct; it is truthful and fluent, but could be better nuanced

3: The definition is correct.

-10: The definition is self-referential; i.e. refers back to the target word itself.

Example

The word *dress* can be used as a noun, or as a verb. Consider the following pairs of example sentences and (correct)definitions:

A Target word: *dress*
Example sentence: I am wearing my beautiful pink *dress*.
Definition: a one-piece garment, typically extending down over the legs in a skirt [1]

B Target word: *dress*
Example sentence: I want to *dress* up nicely for the party.
Definition: to clothe oneself

The definitions above are correct; they are fluent and truthful, and therefore you would judge them with a **3** in the 'judgements' column. If, however, the definition of B would be provided for the example sentence of A (or vice versa), the definitions would be *incorrect* for the target word in the example sentence, because it defines the wrong sense of the target word. In this case, you would judge with the score **0**.

Too specific or too general

Definitions can be too specific or too general to the context in which it is used. An example of a too specific definition for Example sentence A is:

a one-piece garment that is pink and beautiful

This definition is too specific because the sense of the target word *dress* does not necessarily require the dress to be pink nor beautiful, the adjectives in this sentence only specify what the color of the dress is. You should judge this with a **1**.

[1] The complete definition from the Online Oxford English Dictionary is:(https://www.oed.com/search?searchType=dictionary&q=dress&_searchBtn=Search)

An example of a too general definition for sentence 4 would be:

> a piece of cloth

This definition is too general for this example sentence, because 'a piece of cloth' can also describe many other objects, like a t-shirt or a towel, which are not possible in this sentence. You should judge the generated definitions with a **1**, as the definition is not sufficiently truthful.

Definitions could be fluent and truthful, but could be better nuanced, for example:

> a one-piece clothing, often worn by women and girls

This definition is truthful and fluent, and undoubtedly refers to the correct sense of the target word. However, it might be improved with some extra nuance or information. Therefore you would judge this definition by a **2**.

Self-reference

When self-reference occurs, the definition is considered *incorrect* and should receive the special label **-10**. An example of a self-referential definition is:

> Target word: self-conscious
> Definition: the state of being self-conscious

A.2 Annotations judgements

The total counts for all judgments for each category can be seen in the tables below. Table A.1 displays the counts of each judgements, and table A.1 displays these numbers as percentages. Each category consists of a total of $80 \cdot 3 = 240$ judgements .

Category	Corpus	-10	0	1	2	3
stable	C_1	19	38	30	46	107
	C_2	19	43	33	39	106
changing	C_1	22	100	27	37	54
	C_2	13	118	39	27	43
emerging	C_2	30	148	38	10	14

Table A.1: Number of judgements for each category (240 annotations per row)

Category	Corpus	-10	0	1	2	3
stable	C_1	7.9%	15.8%	12.5%	19.2%	44.6%
	C_2	7.9%	17.9%	13.8%	16.2%	44.2%
changing	C_1	9.2%	41.7%	11.2%	15.4%	22.5%
	C_2	5.4%	49.2%	16.2%	11.2%	17.9v
emerging	C_2	12.5%	61.7%	15.8%	4.2%	5.8%

Table A.2: Judgement percentages for each category

A.3 Consensus voting accuracy

Besides taking the majority votes to aggregate judgements, the consensus vote was also computed. In consensus voting, a definition is considered correct only when *all (three) annotators have judged it to be correct.* Recall that a judgement of 2 and 3 are considered correct, while the judgements -10, 0, 1 are considered incorrect. Aggregating the judgements by consensus voting generally displays the same trends as in majority voting (see 5.3), except that the stable target words of C_2 are judged to better than those of C_1. This would imply that the performance of `T5-base` does not deteriorate for sentences containing stable target words.

Category	C_1	C_2	$C_1 \cup C_2$
stable	**43.75%**	**47.50%**	**45.62%**
changing	**27.50%**	12.50%	20.00%
stable + changing	**35.625%**	30.0%	32.81%
emerging	-	2.50%	-
total	**35.625%**	20.83%	26.75%

Table A.3: Accuracy according to consensus vote