Approach	Examples
Baseline	This paper presents a machine learning approach to the task of word sense disambiguation (WSD) based on lexical, syntactic and semantic features. Our approach relies on a rule-based classifier trained on manually annotated corpora. We show that our approach outperforms state-of-the-art WSD systems for English, German and Spanish.
Frequency	This paper presents a new approach to the automatic generation of referring expressions (REs) in natural language. The approach is based on the use of distributional semantics, which can be used to model the meaning of an utterance. We describe how we have constructed a data set for RE generation that has been manually annotated with respect to three dimensions: referential quality, fluency, and informativeness. In addition, we show how these dimensions correlate with human judgments about the quality of generated REs.
Contextual	This paper presents an overview of the work carried out within the EU FP7 European project Information Extraction (IE). In particular we focus on the main tasks involved in IE, i.e., text classification, named entity recognition and coreference resolution. Special attention will be paid to the issues related to multi-document summarization as well as cross-lingual information extraction which is very important for many natural language processing tasks such as machine translation and question answering. Finally some directions for future research are sketched which include the development of more advanced IE systems based on deep learning architectures and the creation of new IE evaluation corpora.
Contextual^2	We present a novel approach to the problem of automatically detecting and classifying semantic relations between named entities (NEs) in natural language text. In particular, we propose a new method for identifying NEs that are not explicitly mentioned in the text, i.e., those that do not appear in training data. The proposed method is based on an unsupervised clustering algorithm that uses word embeddings as features. Experimental results show that our method outperforms state-of-the-art methods by a large margin.

Table 2: Example output for each approach.

	Dev			Test		
	PPL↓	CM↑	CPL↓	PPL↓	CM↑	CPL↓
Baseline-all	19.97	21.85	82.59	22.76	16.22	102.03
Baseline-2	21.66	24.41	91.36	21.53	19.69	94.37
Baseline-3	21.08	22.75	88.75	21.06	19.01	92.06
Frequency-NoLSTM	19.97	22.87	83.18	21.23	19.09	88.38
Frequency	19.97	23.87	82.43	20.20	19.98	87.68
Context	23.47	24.86	96.64	23.21	18.20	102.11
Context^2	**19.66**	**24.94**	**77.54**	**19.81**	**20.12**	**82.43**

Table 3: Experiments results for automatic evaluation on abstracts. ↓ indicates lower is better and ↑ indicates higher is better. p-value < 0.001 for all scores over baseline based on statistical sign test (Dixon & Mood, 1946). Baseline-n means only n previous years' abstracts are used to fine-tune a non-temporal LM. We evaluated n from 1 to 10, and reported the best 2 ($n = 2$ and $n = 3$). Baseline-all means using the whole training set to fine-tune a non-temporal LM. PPL: perplexity score; CM: content meteor score; CPL: content perplexity score (See §4.4 for the detail of these metrics.)

	Topic	Topic New	Problem	Problem New	Method	Method New	Avg
Baseline-all	**100%**	8%	50%	**17%**	42%	25%	46%
Baseline-2	97%	0%	60%	0%	53%	3%	36%
Baseline-3	**100%**	0%	95%	**17%**	35%	12%	44%
Frequency	**100%**	**17%**	83%	0%	**100%**	25%	54%
Context^2	**100%**	**33%**	**100%**	**17%**	**100%**	**50%**	**63%**

Table 4: Experiments results for the human evaluation. See §4.6 for details of the criteria.

content words, we manually collect the non-content words as a stopwords list. During content words based evaluation, we filter out the stopwords, and the leftover tokens are naturally formulated into content words. The perplexity score evaluates fluency while the content words based metrics evaluate the adequacy of future research ideas since ideas are mainly represented by content words instead of non-content words.

Perplexity (PPL) We evaluate perplexity, which is calculated using the standard formula

$$PPL = 2^{-\frac{1}{M}\sum_i^N \log_2(p(x_i))}$$

where $p(x_i)$ is the token probability computed from the model and $M = \sum_i^N |x_i|$.

Content Perplexity (CPL) Perplexity is computed over all words equally, including non-content words. To better evaluate the benefit of the improved content word selection, we calculate perplexity on non-stop words. We call this *content perplexity*. This is computed by ignoring the stopword log probabilities, and only adding the non-stopword log probabilities together and dividing by the number

of non-stopwords instead of the total number of words[8]. For test data $D = x_1, ..., x_N$, the stopwords list is V_s, then the content perplexity CPL is computed by

$$CPL = 2^{-\frac{1}{M_s}\sum_i^N \log_2(p(x_i)I[x_i \notin V_s])}$$

where $p(x_i)$ is the token probability computed from the model and $M_s = \sum_i^N |x_i I[x_i \notin V_s]|$

Content Meteor (CM) This metric measures the match between model generated abstracts and real abstracts in the dev and test sets. We use 100 random seeds to generate $N_g = 100$ abstracts to compare with all abstracts in the dev or test set. After removing all the stopwords, we evaluate the Meteor score for the generated abstract only by the content words. Let $G = \{a_1, ..., a_i, ..., a_{N_g}\}$ be all generated abstracts with the stopwords removed, let $D = \{d_1, ..., d_j, ..., d_{N_h}\}$ be all abstracts in the dev or test set (N_h is the number of abstracts in dev or test set), we compute Content Meteor as:

$$CM = \frac{\sum_{i=1}^{N_g} \max_{j=1}^{N_h} Meteor(a_i, d_j)}{N_g}$$

4.5 Automatic Evaluation Results

Table 3 shows the automatic evaluation experiment results across all experimented models. Our proposed methods perform better than the GPT-2 baselines without temporal information on all automatic evaluation metrics.

The doubly contextualized model has about 3 content meteor points improvement over the year agnostic baseline-all, and 5 points content perplexity improvement, which indicates the content better matches real future abstracts. This demonstrates that the doubly contextualized enables the model to generate content words that will be used in the future. The word frequency model also shows 2 points improvement in the content meteor and slightly better in the content perplexity. This indicates that by only adding the word frequency as bias, the model can increase the content matching slightly. We tried the accumulated baselines on the abstracts of the n most recent years and the performance of only training on the most recent abstracts cannot surpass proposed model.

From a fluency perspective, the word frequency model has the same perplexity as baseline-all. In contrast, the doubly contextualized model shows a larger improvement which indicates that it can enable the model to generate more fluent abstracts than the baselines for future abstracts. Without using the LSTM to model the temporal information, the model only considers a single previous year bias which hurts the performance. Without gating, although the model has a high content matching score, it has a lower fluency score because the model cannot recognize which tokens should be biased. This demonstrates the importance of the gating mechanism in the doubly contextualized model.

4.6 Human Evaluation

For a human evaluation, we randomly evaluate 100 generated abstracts for each approach. Since our temporal language generation task is to generate abstracts, we evaluate the abstracts with six different criteria, with criteria tailored to the abstract generation task. Table 4 shows the human evaluation results for all the experimental methods. We have six criteria for evaluation, which are divided into three abstract content types each with fluency and novelty aspects. Each criterion score is binary 0 or 1 for each abstract. We add all obtained scores together and divide them by the total gold scores to obtain the percentage of the human evaluation score. The human evaluators are NLP researchers. We conducted a blind evaluation, so the human evaluators did not know the approach for abstracts.

- **Topic: Is the topic clear and correct?** We check if an abstract has a fluent topic or background description without factual errors.
- **Topic New: Is the topic new?** We check if an abstract has a topic we have never seen before or matches the recent research topics.
- **Problem: Is the problem clear and correct?** We check if an abstract has a fluent problem description without factual errors.
- **Problem New: Is the problem new?** We check if an abstract introduces a problem that we have never seen or matches the recent research problems.

[8]Appendix C shows how the stopwords are curated.

- **Method: Is the method clear and correct?** We check if the abstract has a fluent method description without factual errors.
- **Method New: Is method new?** We check if an abstract proposes a method we have never seen or matches the recent approaches.

Note that "new" here does not mean completely new. Instead, it only means more related to "future abstracts" such as abstracts in our dev or test set. Apparently, the model cannot generate completely new topics, a new method, or a new problem that they have never seen during the training.

All of our proposed methods outperform the baseline when evaluated using the average score. The generated topics for all approaches are clear and correct, indicating that the GPT-2 baseline can adequately generate clear topics. However, topics are not necessarily new in the baseline approaches, whereas in our proposed approach 1/3 of the topics are new. Additionally, the problem and the method are not always clear and correct in the baseline, whereas our proposed approaches can have all generated new problems, and the methods are clear and correct. In our proposed approach 1/2 of the approaches are new, which shows that our proposed approaches have the ability to predict new trends for future research.

4.7 Case Study

Table 2 shows generated abstracts from all the approaches and Table 6 in Appendix shows more generated abstracts from all the approaches compared to the reference abstracts from ACL conferences. The baseline approach generates more general abstract content that does not contain many details or generate very traditional methods for NLP research. The example from `Context` shows that without gating, the generated output after 2-3 sentences is not related to the starting sentence because it may ignore the previous context, although new content is generated, which shows that the gating mechanism can help the model determine whether the next generated token should be depended on the historical documents or the previous context. In contrast, the $\texttt{Context}^2$ method generates more detailed content and content that is more related to recent research, such as word embeddings or neural networks, and later generated sentences are more coherent to the previous context, which balanced between considering the historical documents to generate new content or following the previous context to generate more coherent text.

5 Related Work

To the best of our knowledge, there is no prior work constructing language models for future text based on temporal historical documents. However, there is much work on language models with temporal information (Röttger & Pierrehumbert, 2021; Lazaridou et al., 2021; Hofmann et al., 2021; Agarwal & Nenkova, 2022; Loureiro et al., 2022). Huang & Paul (2019) worked on document classification using word-level temporal embeddings, and Röttger & Pierrehumbert (2021) adapts the pre-trained BERT models to domain and time. Lazaridou et al. (2021) evaluated the performance of language models on future text, in a setup similar to ours but did not construct any temporally aware models for future language modeling. Dhingra et al. (2022) conducted experiments with temporal language models for question answering. Hofmann et al. (2021) modeled temporal and social information together by modifying BERT with a latent Gaussian process. Rosin et al. (2022) concatenated time tokens to text sequences and introduced time masking using masked language modeling to make a time-aware BERT. However, none of the previous works are about building language models for future text based on temporal historical documents. In this paper, we fill this gap and propose future language models that can generate texts that are more related to future content, which can be applied to many future forecasting areas.

6 Conclusion

In this paper, we introduce the task of future language modeling and propose a series of future language models. We evaluate our models on abstracts in NLP. The proposed approaches outperform the baseline non-temporal language models across all automatic evaluation metrics and human evaluation on generating content related to the future text based on temporal historical documents.