# TRAINING VISUAL LANGUAGE MODELS WITH OBJECT DETECTION: GROUNDED CHANGE DESCRIPTIONS IN SATELLITE IMAGES

*João Luis Prado, Syrielle Montariol, Javiera Castillo-Navarro, Devis Tuia, Antoine Bosselut*

Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland

## ABSTRACT

Recently, generalist Vision Language Models (VLMs) have shown exceptional progress in tasks previously dominated by specialized computer vision models. This becomes more prevalent when visual grounding capabilities, such as the ability to reason over input text and image to generate bounding boxes around objects, are required. However, how these capabilities transfer to specialized domains such as remote sensing remains understudied, despite the recent increase in specialized models for Earth observation. In this work, we evaluate how grounding visual entities – by generating bounding-box coordinates – affects VLM performance in satellite imagery. To this end, we create two instruction-following tasks sourced from the xBD dataset, describing changes due to natural disasters observed in satellite images. We fine-tune several instances of MiniGPTv2, an open-source VLM with grounding capabilities, and evaluate their performance under the "grounded" vs. "not grounded" settings. We find that generating bounding boxes to refer to visual entities increases performance in tasks related to objects in the image, but only when the number of entities in the image is limited.

***Index Terms***— Vision-Language Models, Object Detection, Earth Observation

## 1. INTRODUCTION

Implementing Artificial Intelligence (AI) methods in Remote Sensing (RS) has the potential to enhance performance and enable new applications for various downstream tasks in environmental and geo-sciences, such as damage assessment, agricultural yield prediction, and land cover monitoring [1]. Among recent advances in AI, the augmentation of generalist large language models (LLMs) with image encoders paved the way for powerful generative Vision Language Models (VLMs), which achieve state-of-the-art performance in a wide range of tasks, such as image captioning and visual question answering [2]. A recent development in VLMs is the addition of *referential and grounding tasks* to the pre-training or fine-tuning data mixtures [3, 4, 5]. In such tasks, the model is generally required to generate or reason with bounding boxes surrounding objects or with segmentation maps, in order to match the referred visual entities with text. For example, a VLM might be asked to *"find all the taxi cars that aren't parked"* in a busy city scene. The model needs to answer by generating bounding box coordinates. Preliminary studies indicate that exposure to such grounded tasks during training may help mitigate hallucinations [4] and even increase performance in non-grounding tasks.

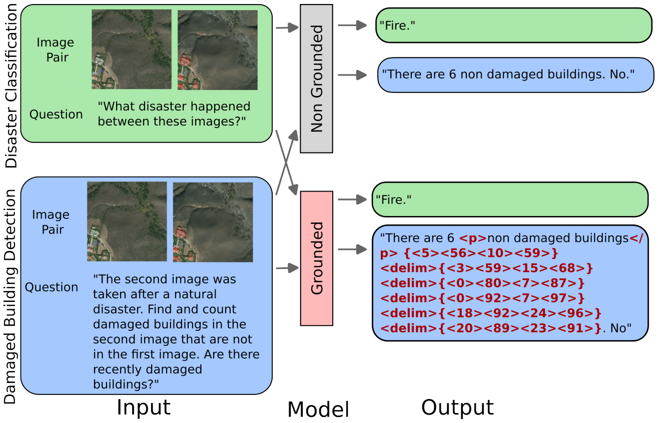

Disaster Classification
Image Pair
Question
"What disaster happened between these images?"
Damaged Building Detection
Image Pair
Question
"The second image was taken after a natural disaster. Find and count damaged buildings in the second image that are not in the first image. Are there recently damaged buildings?"
Non Grounded
Grounded
"Fire."
"There are 6 non damaged buildings. No."
"Fire."
"There are 6 <p>non damaged buildings</p> {<5><56><10><59>} <delim>{<3><59><15><68>} <delim>{<0><80><7><87>} <delim>{<0><92><7><97>} <delim>{<18><92><24><96>} <delim>{<20><89><23><91>}. No"
Input
Model
Output

**Fig. 1**: Pipeline with the "non grounded" and "grounded" models. We define two tasks: (i) Disaster classification (green boxes), where the model is expected to determine which event happened between images. Possible answers are: {"Flood", "Hurricane", "Volcano Eruption", "Fire", "Tsunami", "Earthquake" }. Note that in the post-disaster image, the bounding boxes are only added for illustrative purposes, and not actually given to the model as input. (ii) Damaged building detection (blue boxes), where the model is asked if there are damaged buildings in the second image. Before answering, it is tasked to either count the number of damaged buildings (non-grounded version) or count them and provide coordinates of their location (grounded version). Given the models' count of damaged buildings, possible answers are : {"There are $c$ damaged building(s). Yes.", "There are $c$ non-damaged building(s). No.", "There are no buildings. No."}.

However, matching the performance achieved by VLMs in generic computer vision datasets in the RS domain remains an open challenge due to the intrinsic differences between remote sensing imagery and the natural images on which VLMs are usually trained [6, 7]. In opposition to natural images, which tend to capture close-distance depictions of a scene or collections of salient subjects, remote sensing imagery is typically long-distance, often capturing kilometer-wide fields of view. It often presents only a sparse set of salient visual features, which makes the skill of understanding referring and grounding expressions both useful and challenging. To the best of our knowledge, GeoChat [8] is the only work having explored fine-tuning VLMs for RS with referential and grounding tasks.

In this work, we study how fine-tuning pre-trained VLMs with tasks requiring bounding box generation affects model performance on RS tasks. We use images and annotations of the xBD dataset [9], which contains pairs of high-resolution satellite images taken before and after a natural disaster. We adapt the original dataset into two instruction-following generative tasks (see Fig. 1): (i) *Disaster Classification* where the task is, given an image pair, to identify which disaster type happened among a set of possible natural disasters, and (ii) *Damaged Building Detection* where the task is to infer if there are compromised buildings by first counting building instances, generating the corresponding bounding boxes, and then generating a binary decision on the presence of damaged buildings. We expect specialized CV models to perform well in these tasks. Our work explores how VLMs can be adapted to this domain-specific setting, when given the capability to localize object by generating bounding boxes. We find that grounded models outperform their non grounded counterparts in damaged building detection, when the task is limited to samples with a low number of buildings.

## 2. PIPELINE

### 2.1. Dataset

The xBD dataset provides image pairs of regions before and after a natural disaster with object-level annotations of the destruction level of buildings. Our dataset is comprised of 11‘930 $\{1024 \times 1024$ image pair, instruction, answer$\}$ triplets, generated from 5‘965 image pairs sourced from the xBD dataset [9]. These triplets are separated in a training set and two test sets, one in-sample and one out-of-sample. The training set, consisting of 2‘952 images and the in-sample test set, consisting of 494 images pairs, are sourced from 11 disaster events, which are named “Tier 1” in the xBD dataset. The out-of-sample test set consists of 2‘519 images sourced from 8 other disaster events and is named “Tier 3” in the original dataset. We keep the original naming and splits for the test sets and retain only image pairs containing less than 50 destroyed buildings. Two tasks are defined:

- *Damaged building detection*: assessing whether there are buildings having suffered structural damage after a natural disaster or not, given a satellite images of a region. We formulate these instructions as a counting task followed by a binary classification task (*‘Yes’ / ‘No’*), expecting that counting may help the model reason about the image, even if performance in the counting sub-task is not evaluated. In the “grounded” case, the model is further required to generate bounding box coordinates of the identified damaged buildings. In both instances, we only evaluate the binary task of damaged building detection, and treat counting and coordinates generation as auxiliary tasks, only used to help the model’s reasoning.
- *Disaster classification*: identifying the natural disaster occurring between the provided pair of images. The training set and both test sets contained examples of 6 different event classes, sourced from 19 different events. We formulate this task without a bounding box generation subtask (grounding). It allows us to analyze the impact of jointly fine-tuning with a grounding task on the performance of non-grounded tasks.

### 2.2. Models

We differentiate between “grounded” and “non grounded” model fine-tuning (Fig. 1). In the “non grounded” case, the model is trained to count the number of buildings in an image, then solve the damaged building detection task, while in the “grounded” case, the model generates bounding boxes coordinates for each damaged building before solving the task. The bounding box generations are expressed in free text with locator tokens “<p><\p>” and a list in the format $\{<x_1><y_1><x_2><y_2>\}$ <delim>, where $x_i$ and $y_i$ are respectively horizontal and vertical coordinates with origin at the top-left corner of the image. The first pair of coordinates corresponds to the top left corner, and the second pair to the bottom right corner of the bounding box. An example of grounded instruction is depicted in Fig. 1. Thus, during training, the input is the same for both models: the image pair and question for both tasks. The difference lies in the ground truth of the Buildings task used for fine-tuning the VLM, which contains bounding box coordinates in the grounded version of the dataset. Hence, the model learns to count the buildings and generate their coordinates before answering the question “*Are there recently damaged buildings?*”.

### 2.3. Experimental Settings

The model used in this work, MiniGPTv2 [10], is an open-source VLM building on Llama2 [11] and the EVA visual encoder [12], implementing referential grounding through numerically encoded bounding boxes. We use its instruction-tuned version.[1] Experiments are performed in a single A100 40 Gb RAM GPU. The learning rate evolves according to a cosine scheduler with 1‘000-step warm-up. Minimum and

[1] `https://github.com/Vision-CAIR/MiniGPT-4/`, “after stage-2” version.

**Table 1**: Building detection and disaster classifications F1 Score of MiniGPTv2 models trained without and with bounding box generation subtasks (grounding) against a maximum training set frequency classifier baseline. We report performance over two disjoint subsets of our evaluation sets, consisting of images with up to 3 objects (left) and images with more than 3 objects (right). Models correspond to instances of MiniGPTV2 trained to detect damaged buildings (with bounding box generation in the grounded case) and classify disasters jointly.†In the damaged building detection task, we report the performance of the grounded model trained on a subset of the train set. Bolded results correspond to model achieving best performance over all experiments in the evaluation category. We perform independent T-tests to identify the best model. $\star$: $p < 0.05$, $\star\star$: $p < 0.005$.

| | | Few buildings (0-3) | | Many buildings (4-50) | |
|---|---|---|---|---|---|
| Task | Model | Tier 1 (160) | Tier 3 (1139) | Tier 1 (334) | Tier 3 (1380) |
| Damaged Building Detection | Baseline (max freq.) | 41.18 | 34.50 | 38.49 | 29.52 |
| | Not grounded | 61.11 ± 2.51 | 51.54 ± 2.06 | **71.09**$^{\star\star}$ ± 1.45 | **62.64**$^{\star}$ ± 2.87 |
| | Grounded 3$^{\dagger}$ | 60.94 ± 2.65 | **54.55**$^{\star\star}$ ± 1.33 | 52.78 ± 1.65 | 45.92 ± 4.13 |
| | Grounded 10$^{\dagger}$ | 57.91 ± 4.51 | 44.61 ± 3.35 | 66.07 ± 1.77 | 53.85 ± 4.02 |
| Disaster Classification | Baseline (max freq.) | 9.52 | 3.43 | 4.61 | 5.52 |
| | Not grounded | 65.17 ± 1.64 | 22.32 ± 2.92 | **78.68**$^{\star}$ ± 2.29 | 28.95 ± 2.56 |
| | Grounded | **69.57**$^{\star\star}$ ± 0.57 | 22.54 ± 1.35 | 74.01 ± 1.02 | 25.60 ± 1.08 |

maximum learning rates are respectively set to $10^{-5}$ and $8 \cdot 10^{-5}$. We follow MiniGPTv2 default parameters for LoRA fine-tuning [13]. We evaluate models in three independently seeded trials with class-weighted F1 Score on both Tier 1 (in-sample) and Tier 3 (out-of-sample) test sets, to infer how grounding affects the models' generalization capability. We compare with a simple baseline consisting in choosing the most frequent class in the full training set. In order to study the influence of the distribution of number of buildings in the training on the models' performance, we train MiniGPTV2 on subsets of the training data containing images with up to $n \in \{1, 3, 5, 10, 50\}$ buildings. We denote a model trained on data containing up to $n$ buildings as "Grounded $n$" or "Not Grounded $n$". If $n$ is not specified, the model is trained on the full training set. Note that all training runs are performed for the same number of iterations (16'794 steps), even when training on subsets of the train set.

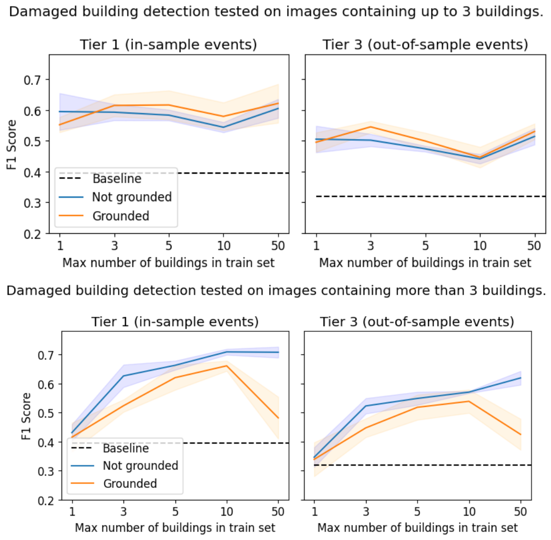


**Fig. 2**: F1 Scores on Tier 1 (in-sample disasters) and Tier 3 (out-of-sample disasters) test sets for different training set filters (horizontal axis) in (top) subset of test sets with up to 3 buildings and (bottom) subset of test sets with at least 4 and up to 50 buildings.

## 3. RESULTS

The class-weighted F1 Score of MiniGPTv2 instances trained with non-grounding and grounding instructions are depicted in Table 1, along with a maximum frequency classifier baseline. Disaster classification and building detection results are averaged over all runs. As expected, fine-tuned MiniGPTv2 models systematically perform better in the Tier 1 test set, as the domain gap is small with respect to events from Tier 3. The performance gap is much higher in the Disaster task, showing that generalization to unseen locations and disasters is challenging. For the Buildings task, we compare two grounded models with their non-grounded counterpart: Grounded 3 and Grounded 10 (trained on a subset of the training set containing at most 3 and 10 buildings respectively). The Grounded 3 model outperforms its non-grounded counterpart in the Building task only when we restrict evaluation to the subset of images containing up to 3 images. The drop in performance of models generating bounding boxes when evaluated with images with many objects can be explained by the fact that the datasets used for MiniGPTv2 pre-training and fine-tuning contain a smaller number of entities than the xBD dataset. Indeed, we note that our training set contains 11 objects per image on average, while RefCOCO and Ref-

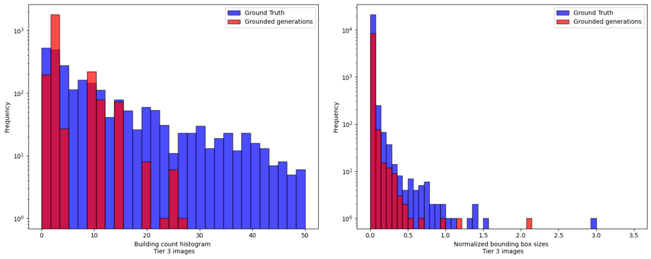
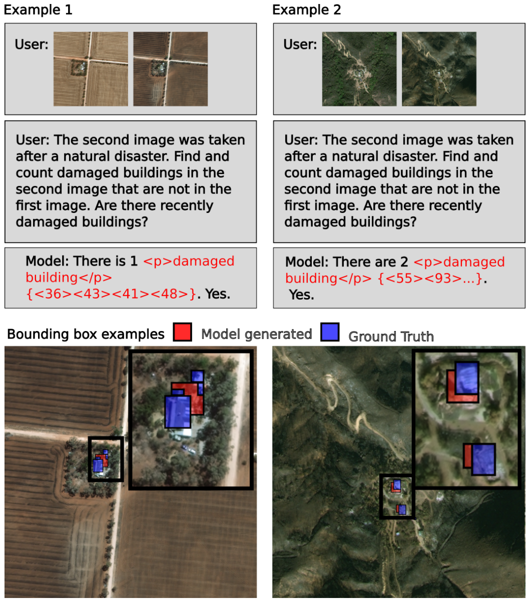


**Fig. 3**: (left) Histogram of buildings to be grounded per image pair in the Tier 3 test set (blue) as generated by a model trained on the full dataset (orange). (right) Histogram of normalized bounding box sizes (full image size corresponds to 100) generated in the Tier 3 test set (blue) and generated by a model trained on the full dataset (orange). Distributions in the train set and Tier 1 test set are very similar.

COCO+ [14], on which MiniGPTv2 is trained, contains only 2.5 objects per image on average. Moreover, Fig. 3 (left) shows that, even when trained and evaluated with up to 50 entities per image, grounded models tend to underestimate the number of entities. We also observe that the grounded model usually only generates small bounding boxes, as depicted in Fig. 3 (right); the model underestimates the size of objects, and is unable to identify the largest ones. The performance in the Disaster task does not show a clear trend when comparing grounded and non-grounded models. This result is expected, since the Disaster task is not directly influenced by the additional grounding information carried by the bounding box coordinates.

We further investigate how grounded and non grounded models' performance in the Building task vary when trained over different training set subsets. Fig. 2 shows the class-weighted F1 Score across models trained on filtered training sets. We observe that grounded models are better than non-grounded models in samples containing few buildings. However, when we evaluate on images with 4 or more objects, the trend reverses and non-grounded models perform better.

**Qualitative assessment of grounded generations.** We provide two examples of prompts and generated answer on the Building task after grounded fine-tuning. The first example (Fig. 4 left) shows a common failure mode of the model. The associated disaster is a bushfire in Australia, with burned crops visible in the post-disaster image. The grounded model is not able to discern the four closely packed buildings, and outputs a single bounding box accounting for the center of the entity-dense region. The model generates fewer bounding boxes than expected, confirming the skewed bounding box distribution depicted in Fig. 3. The second example (Fig. 4 right) depicts images of the 2017 California wildfires. The model correctly counts the buildings and outputs bounding boxes overlapping with ground truth buildings, successfully grounding the textual reference to the visual object.



User: The second image was taken after a natural disaster. Find and count damaged buildings in the second image that are not in the first image. Are there recently damaged buildings?

Model: There is 1 <p>damaged building</p> {<36><43><41><48>}. Yes.

User: The second image was taken after a natural disaster. Find and count damaged buildings in the second image that are not in the first image. Are there recently damaged buildings?

Model: There are 2 <p>damaged building</p> {<55><93>...}. Yes.

**Fig. 4**: (Top) Two examples of model generations for the Damaged Building Detection task with grounding. The user provides an image pair before and after the disaster and a query. The model answers with a collection of bounding boxes containing identified (non-)damaged buildings and an answer to the query. (Bottom) Image after disaster overlayed with generated bounding boxes (red) and ground truth bounding boxes (blue). Black outlines correspond to zooms in regions with several buildings.

## 4. CONCLUSION

In this work, we use MiniGPTv2, a VLM fine-tuned with tasks related to object detection, to evaluate how visual grounding through the generation of bounding box coordinates interleaved with text affects performance on a RS task related to object detection: damaged building detection. We find that grounded generations do not outperform non-grounded outputs in general, but that identifying the precise location of objects – through generating bounding boxes – helps the model when the number of objects is low. It is explained by the fact that the original training set of MiniGPTv2 over-exposes the model to samples with few entities.

VLMs such as MiniGPTv2 are known for their high generalization ability, thanks to their extensive fine-tuning on a very large set of image-text pairs. However, the performance of the Disaster classification task sharply drops from the in-sample to the out-of-sample test sets. The model uses spurious correlations to solve the task, relying on contextual features (e.g. the type of terrain) as a proxy to identify the disaster. We hypothesize that this is due to a heavy lack of similar images during pre-training. Thus, in specialized domains such as RS, further fine-tuning VLMs on similar images should be explored to account for this domain gap.

## 5. ACKNOWLEDGEMENTS

We gratefully acknowledge the support of the Swiss National Science Foundation (No. 215390), Innosuisse (PFFS-21-29), the EPFL Science Seed Fund, the EPFL Center for Imaging, Sony Group Corporation, and the Allen Institute for AI.

## 6. REFERENCES

[1] Gustau Camps-Valls, Devis Tuia, Xiao Xiang Zhu, and Markus Reichstein (Editors), *Deep learning for the Earth Sciences: A comprehensive approach to remote sensing, climate science and geosciences*, Wiley & Sons, 2021.

[2] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi, “Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models,” 2023.

[3] Haoxuan You, Haotian Zhang, Zhe Gan, Xianzhi Du, Bowen Zhang, Zirui Wang, Liangliang Cao, Shih-Fu Chang, and Yinfei Yang, “Ferret: Refer and ground anything anywhere at any granularity,” 2023.

[4] Keqin Chen, Zhao Zhang, Weili Zeng, Richong Zhang, Feng Zhu, and Rui Zhao, “Shikra: Unleashing multimodal llm’s referential dialogue magic,” 2023.

[5] Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei, “Kosmos-2: Grounding multimodal large language models to the world,” 2023.

[6] Fan Liu, Delong Chen, Zhangqingyun Guan, Xiaocong Zhou, Jiale Zhu, and Jun Zhou, “Remoteclip: A vision language foundation model for remote sensing,” 2023.

[7] Yuan Hu, Jianlong Yuan, Congcong Wen, Xiaonan Lu, and Xiang Li, “Rsgpt: A remote sensing vision language model and benchmark,” 2023.

[8] Kartik Kuckreja, Muhammad Sohail Danish, Muzammal Naseer, Abhijit Das, Salman Khan, and Fahad Shahbaz Khan, “Geochat: Grounded large vision-language model for remote sensing,” 2023.

[9] Ritwik Gupta, Richard Hosfelt, Sandra Sajeev, Nirav Patel, Bryce Goodman, Jigar Doshi, Eric Heim, Howie Choset, and Matthew Gaston, “xbd: A dataset for assessing building damage from satellite imagery,” 2019.

[10] Jun Chen, Deyao Zhu, Xiaoqian Shen, Xiang Li, Zechun Liu, Pengchuan Zhang, Raghuraman Krishnamoorthi, Vikas Chandra, Yunyang Xiong, and Mohamed Elhoseiny, “Minigpt-v2: large language model as a unified interface for vision-language multi-task learning,” 2023.

[11] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom, “Llama 2: Open foundation and fine-tuned chat models,” 2023.

[12] Yuxin Fang, Wen Wang, Binhui Xie, Quan Sun, Ledell Wu, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao, “Eva: Exploring the limits of masked visual representation learning at scale,” 2022.

[13] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen, “Lora: Low-rank adaptation of large language models,” 2021.

[14] Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara L. Berg, “Referitgame: Referring to objects in photographs of natural scenes.,” in *EMNLP*, Alessandro Moschitti, Bo Pang, and Walter Daelemans, Eds. 2014, pp. 787–798, ACL.