ABSTRACT
In this paper we present an approach for extracting and linking entities from short and noisy microblog posts. We describe a diverse set of approaches based on the Semanticizer, an open-source entity linking framework developed at the University of Amsterdam, adapted to the task of the #Microposts2015 challenge. We consider alternatives for dealing with ambiguity that can help in the named entity extraction and linking processes. We retrieve entity candidates from multiple sources and process them in a four-step pipeline. Results show that we correctly manage to identify entity mentions (our best run attains an F1 score of 0.809 in terms of the strong mention match metric), but subsequent steps prove to be more challenging for our approach.

Keywords
Named entity extraction; Named entity linking; Social media

1. INTRODUCTION
This paper describes our participation in the named entity extraction and linking challenge at #Microposts2015. Information extraction from microblog posts is an emerging research area which presents a series of problems for the natural language processing community due to the shortness, informality and noisy lexical nature of the content. Extracting entities from tweets is a complex process typically performed in a sequential fashion. As a first step, named entity recognition (NER) aims to detect mentions that refer to entities, e.g., names of people, locations, organizations or products (also known as entity detection), and subsequently to classify the mentions into predefined categories (entity typing). After NER, named entity linking (NEL) is performed: linking the identified mentions to entries in a knowledge base (KB). Due to its richness in semantic content and coverage, Wikipedia is a commonly used KB for linking mentions to entities, or deciding when a mention refers to an entity that is not in the KB, in which case it is referenced by a NIL identifier. DBpedia aims to extract structured information from Wikipedia, and combines this information into a huge, cross-domain knowledge graph which provides explicit structure between concepts and the relations among them.

Our participation in this challenge revolves around the existing open-source entity linking software developed at the University of Amsterdam. We use Semanticizer\(^1\), a state-of-the-art entity linking framework. So far Semanticizer has been successfully employed in linking entities in search engine queries \(^1\) and in linking entities in short documents in streaming scenarios \(^6\). Moreover, it has been further extended to deal with additional types of data like television subtitles \(^3\). In what follows we explain how we use Semanticizer for the task at hand, and describe each of our submitted runs to the competition.

2. SYSTEM ARCHITECTURE
Our system processes each incoming tweet in four stages: mention detection, entity disambiguation and typing, NIL identification and clustering, and overlap resolution. We explain each stage in turn.

Mention detection: The first step aims to identify all entity mentions in the input text, and is oriented towards high recall. We take the union of the output of two mention identification methods:

- **Semanticizer**: the state-of-the-art system performs lexical matching of entities’ surface forms. These surface forms are derived from the KB, and comprise anchor texts that refer to Wikipedia pages, disambiguation and redirect pages, and page titles as described in Table 1. For this, we use two instances of Semanticizer, running on two Wikipedia dumps: one dated May 2014 (the version used to build DBpedia 3.9), and a more recent one, dated February 2015.

- **NER**: For identifying entity mentions that do not exist in Wikipedia, i.e., out of KB entities, we employ a state-of-the-art named entity recognizer, previously applied to finding mentions of emerging entities on Twitter \(^2\). We train five different NER models, three using the ground truth data from the Microposts challenges (2013 through 2015), one using pseudo-ground truth (generated by linking tweets as in \(^2\)) and one trained on all data.

Given the candidate mentions identified by NER and Semanticizer, we include a binary feature to express whether the mention has been detected by both systems (\( \geq 2 \) mentions corresponding to the same KB entity), which we refer to as a strong mention.

Our system processes each incoming tweet in four stages: mention detection, entity disambiguation and typing, NIL identification and clustering, and overlap resolution. We explain each stage in turn.

*Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

\(^{1}\)https://github.com/semanticize/semanticizer
We expect the NER runs to be superior to Semanticizer or ES-only (non-linked mentions get a fixed score). We then find the highest scoring path through the graph using dynamic programming, and cast the disambiguation step of identifying the correct entity for a mention as a learning to rank problem.

Next to the features in Table 1, we use additional full-text search features. We index Wikipedia using ElasticSearch (ES), and issue queries dependent contextualization of streaming data. In ECIR 2014. Springer, 2014.

We have presented a system that performs entity mention detection, disambiguation and clustering on short and noisy text by drawing candidates from multiple sources and combining them. We observe that our simple NER and ES runs perform better than our more complex runs. We believe that more robust methods are needed to deal with the errors introduced at each step of the pipeline. For future work we plan on improving mention detection with additional Semanticizer features.

### Acknowledgements
This research was partially supported by the Netherlands Organisation for Scientific Research (NWO) under project numbers 727.011.005, SEED and 640.006.013. DADAISM; Amsterdam Data Science, and the Dutch national program COMMIT.

### REFERENCES