Finding News Citations for Wikipedia

Besnik Fetahu†, Katja Markert†, Wolfgang Nejdl†, Avishek Anand†

1L3S Research Center
Leibniz University of Hannover
Hannover, Germany
{fetahu,nejdl,anand}@L3S.de

2Institute of Computational Linguistics
Heidelberg University
Heidelberg, Germany
markert@cl.uni-heidelberg.de

ABSTRACT

An important editing policy in Wikipedia is to provide citations for added statements in Wikipedia pages, where statements can be arbitrary pieces of text, ranging from a sentence to a paragraph. In many cases citations are either outdated or missing altogether.

In this work we address the problem of finding and updating news citations for statements in entity pages. We propose a two-stage supervised approach for this problem. In the first step, we construct a classifier to find out whether statements need a news citation or other kinds of citations (web, book, journal, etc.). In the second step, we develop a news citation algorithm for Wikipedia statements, which recommends appropriate citations from a given news collection. Apart from IR techniques that use the statement to query the news collection, we also formalize three properties of an appropriate citation, namely: (i) the citation should entail the Wikipedia statement, (ii) the statement should be central to the citation, and (iii) the citation should be from an authoritative source.

We perform an extensive evaluation of both steps, using 20 million articles from a real-world news collection. Our results are quite promising, and show that we can perform this task with high precision and at scale.

1. INTRODUCTION

Wikipedia has become the most used Internet encyclopedia and, indeed, one of the most popular websites overall. In addition, due to Wikipedia’s inclusion into widely used applications such as Google KnowledgeGraph or Apple’s Siri system, its content will influence the knowledge and, potentially, the behavior of millions of users, even if they do not visit the Wikipedia site directly. Therefore, it is essential that its content is accurate and reliable.

In contrast to traditional encyclopedias, Wikipedia is not authored mainly by experts. Also, the articles are authored collaboratively by more than just a small number of contributors and the identity and expertise of authors is hard to verify. This leaves Wikipedia articles open to addition of inaccurate content, spamming or vandalism, and calls into question its reliability. A substantial number of reliability studies have compared Wikipedia against other reference works (such as the Encyclopedia Britannica or drug package information) or subjected them to expert review: The exhaustive survey in [15] concludes that the results of these studies have overall been favourable to Wikipedia when it comes to accuracy of facts, although some works (especially on medical articles) found errors of omission.

These surprisingly favorable results on the reliability of Wikipedia can in all probability be traced to a small number of Wikipedia editorial policies, one of which we are concerned with in this paper. The Verifiability policy requires Wikipedia contributors to support their additions with citations from authoritative external sources. In particular, Wikipedia policy states that “articles should be based on reliable, third-party, published sources with a reputation for fact-checking and accuracy.” This policy, on the one hand, guides contributors towards both neutrality and the importance of authoritative assessment and, on the other hand, allows Wikipedia core editors to identify unreliable articles more easily via a lack of such citations. Citations therefore play a crucial role in ensuring and upholding Wikipedia reliability.

For current and recent events, news citations are one of the most-used sources. Again, Wikipedia encourages the use of news outlets as citations: “news reporting from well-established news outlets is generally considered to be reliable for statements of fact.” As we show in Section 3, news are indeed the second-most widely used citation category in Wikipedia (with 1.88 million citations in our English Wikipedia snapshot) – however, around 26% of these are no longer available due to dead or redirected links. In addition, new information is added all the time and will need verification. For both these purposes, an automatic way of finding an authoritative news citation for any fact(s) one might wish to update, locate again or add would greatly facilitate Wikipedia editing and improve its reliability. Moreover, if no such citation can be found, it can guide contributors or core editors towards questioning their edits.

In this paper, we suggest such a method for automatic news citation discovery for Wikipedia. In particular, we make the following contributions: (i) We analyze for which type of Wikipedia statements a news citation is appropriate (in contrast to, for example, a scientific journal citation), taking into account the type and structure of entity the statement is about, as well as the language the statement is written in. We provide a supervised learning algorithm for statement classification into citation categories. (ii) We then develop a citation discovery algorithm which formalizes three properties of a good citation, namely that it entails the statement it describes and upholding Wikipedia reliability.
supports, that it is from an authoritative source and that the statement it supports is central to it. (iii) We establish a large-scale evaluation framework for citation discovery which uses crowdsourcing for measuring our approach’s precision.

To the best of our knowledge, this is the first work that automatically discovers citations for fine-grained Wikipedia statements. We show that news citations can be discovered with high precision, in large contemporary news collections. In particular, we with high accuracy recover the same or very similar citations as the ones originally given by Wikipedia contributors in the presence of numerous strong distractors or even find citations which are preferable to the original ones (as established via crowdsourcing).

2. PROBLEM DEFINITION AND APPROACH OUTLINE

In this section, we describe the terminology and problem definition for finding news citations for Wikipedia.

2.1 Terminology and Problem Definition

We operate on a specific snapshot of Wikipedia \( W \) where the text in each Wikipedia page \( e \in W \) is organized into sections denoted by \( \Psi(e) \). Additionally, entity pages are organized into a type structure, which is a directed-acyclic-graph (DAG) induced by the Wikipedia categories. This is routinely exploited by knowledge bases like YAGO (e.g. Barack Obama is a Person) and we leverage this type structure where each page \( e \) belongs to a set of types \( T(e) \). We, however, modify the original YAGO type structure to make it depth consistent as explained in Section 4.3.

2.1.1 Citations and Wikipedia Statements

Citation: In Wikipedia pages, any piece of text can be supported by a citation. The citation points to an external information source, such as a news article, blog, book or journal, that is considered as evidence for the fact mentioned in the text. Citations in Wikipedia are categorized into a predefined set of 16 citation categories viz. \( c = \{\text{web}, \text{news}, \text{books}, \text{journal}, \text{map}, \text{comic}, \text{court}, \text{press release}., . . . \} \). The distribution of the citation types is given in Figure 2.

Statement: We will refer to the piece of text from a Wikipedia page that has or needs a citation as a Wikipedia statement or simply a statement. In this work, we restrict statements to a single sentence or a sequence of sentences that occur between two consecutive citation markers or a citation marker and paragraph beginning/end. A citation marker is either an actual citation or a placeholder citation needed. We therefore leave the identification of statements to future work. We also do not consider finding evidence for partial sentences or clauses. Each statement \( s \) in a page \( e \) belongs to a section \( \psi \in \Psi(e) \), and the set of statements extracted from a section \( \psi \) of \( e \) is represented as \( S(e, \psi) \).

Anchors and Entities: Typically words or phrases in statements link to other Wikipedia pages which represent entities through anchors. We denote these links to other pages or entities starting from a statement \( s \) as \( \gamma(s) \), and \( T(s) = \{T(e) \mid e \in \gamma(s)\} \) the corresponding entity types.

2.1.2 Citation Finding Tasks

We posit that the following two tasks are integral to finding a citation for a Wikipedia statement.

Statement Categorization. For a statement \( s \) from a page \( e \) of an unknown citation category, the task aims to determine the correct citation category for \( s \).

\[ SC : f(s, e) \rightarrow c, \text{where } c \in \{\text{web, news, . . . }\} \] (1)

We want to categorize \( s \) as a news statement if it requires a news citation. This is based on the hypothesis that each statement typically has a preferred citation category, which we need to determine before making a high precision citation recommendation.

Citation Discovery. Given a (i) statement \( s \) found in page \( e \) and of category \( c = \text{news} \), and (ii) an external news collection \( N \), we define the citation discovery task as finding articles \( n \in N \) that serve as evidence for \( s \). We define the function \( FC \) which for \( s \) outputs the subset of articles that can be suggested for citation.

\[ FC : f(s, e, N) \rightarrow \{\text{correct}, \text{incorrect}\} \] (2)

2.2 Approach Overview

Figure 2 shows an overview of our approach. For an entity, we extract entity and type structure, and its statements and finally run the steps of statement categorization and citation discovery.

Statement Categorization–SC. In the first step, we predict the citation category of a Wikipedia statement \( s \) via supervised machine learning. We train a multi-class classification model, where the classes correspond to the citation categories \( c \).

Citation Discovery–FC. In the second step, for all news statements we find evidence for them via news articles. We retrieve candidate news articles from a news collection \( N \) through standard information retrieval methods with \( s \) serving as our query, and classify each candidate as either an appropriate citation for \( s \) or not.

3. WIKIPEDIA GROUND-TRUTH

3.1 Ground-Truth: Wikipedia News Statements

From a Wikipedia snapshot \( W \) (2015-07-01) we extract all statements and all citations associated with that statement.\[\] We extract 6.9 million statements with 8.8 million citations, from 1.65 million entities and 668k section types.

Citations are categorized into one of the categories \( c \) by the Wikipedia editors. However, sometimes the editors do not categorize a citation as news although they should do so. For example, in \( W \), its top–3 news domains BBC, NYTimes, Guardian, are often cited in categories other than news. Most of such violations by the editors occur when citing news under the category web, which often is a catch-all for almost any type of resource (news, book, etc.). In most cases such violations can be accurately corrected by applying two simple heuristics:

Majority Voting. Citations from the same domain URL are tagged with different categories. We resolve such cases based on majority voting. In case a domain is cited more often under the news category, then all citations to the same domain are changed to news.

URL Patterns. In this heuristic we look for patterns in the URL, specifically for ‘/news/’ and ‘/http://news/’. This rule is applied to web statements, and in case the URL matches one of the patterns, we change its category to news.

Table 1 shows the top–4 most frequent citation categories and the impact of our ground-truth curation rules. Rule application changes

As a statement can have different clauses, sometimes extracted citations only serve as evidence for part of the statement. We, however, do not distinguish at this level of granularity but assume that all associated citations support the whole statement.

Thus, the citation http://news.bbc.co.uk/1/hi/uk_politics/7433479.stm from the entity Liam Byrne has been categorized as web, although the more specific news category would have been appropriate.

https://en.wikipedia.org/wiki/Template:Citation_needed
the citation category for 1,652,619 citations, approximately 18% of all citations in W. The cells in the table show the number of statements that are changed from the category in the row to the category in the column table.

<table>
<thead>
<tr>
<th>book</th>
<th>journal</th>
<th>news</th>
<th>web</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>2,650</td>
<td>1,135</td>
<td>71,801</td>
</tr>
<tr>
<td>14,905</td>
<td>13,542</td>
<td>110,133</td>
<td>391,634</td>
</tr>
<tr>
<td>5,698</td>
<td>2,770</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>16,549</td>
<td>25,109</td>
<td>944,977</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 1: The cells show the number of statements that are changed from one category to another category after ground-truth curation.

We say that a statement is a news statement if it contains at least one news citation (after ground-truth curation). Figure 2 shows the statement distribution across the categories. It is evident that web and news are the two most popular categories, with 5.3 and 1.88 million citations, coming from 1.2 million and 436k entities, respectively.

3.2 Wikipedia News Collection

From the news statements, we extract the cited news articles and construct the Wikipedia news collection $N^W$, which serves as our ground-truth for the citation discovery task. We define $N_t \subseteq N^W$ as the set of articles cited from statements $s$ which come from entities of type $t$. With $N_s$ we denote the set of articles cited by $s$.

From the collection of news statements, we have 1.88 million citations to news articles (see above). We successfully crawled 1.5 million articles. The remaining 19% of citations point to non-existent articles (dead links, moved content etc.). Furthermore, some of the successfully crawled URLs point to the index pages. This can be noticed when we consider the article length (in terms of characters) in Figure 3. Filtering out articles that are below 200 characters, we are left with with 1.39 million articles, a decrease of 26% from the original 1.88 million news citations.

An additional issue we notice in $N^W$ are citations to non-English news articles. We find that 23% of articles in $N^W$ are in languages other than English, using Apache Tika4 for language detection.

4. STATEMENT CATEGORIZATION

In the statement categorization task, we are given a statement $s$ and the entity $e$ from which it is extracted. We compute features that exploit the language style of $s$ and the type and section structure of $e$ to categorize $s$ into one of the citation categories $c$. We learn a multi-class classifier (Section 4.3) with classes corresponding to citation categories $c$ and optimize for predicting news statements. Table 2 shows an overview of the feature list.

<table>
<thead>
<tr>
<th>feature</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>#verbs_attr</td>
<td>the number of verbs of attribution</td>
</tr>
<tr>
<td>#POS</td>
<td>the frequency of POS tags in $s$</td>
</tr>
<tr>
<td>$\lambda(s)$</td>
<td>temporal proximity of $s$ to time point</td>
</tr>
<tr>
<td>discourse</td>
<td>discourse annotations of $s$</td>
</tr>
<tr>
<td>#quotations</td>
<td>the frequency of quotations in $s$</td>
</tr>
<tr>
<td>$\theta(s, N_t)$</td>
<td>LM score of $s$</td>
</tr>
<tr>
<td>LDA($s, N_t$)</td>
<td>similarity of $s$ to a topic model</td>
</tr>
<tr>
<td>$p(s = news</td>
<td>\psi)$</td>
</tr>
<tr>
<td>$p(s = news</td>
<td>t)$</td>
</tr>
<tr>
<td>$p(s = news</td>
<td>t', \psi)$</td>
</tr>
</tbody>
</table>

Table 2: Feature list for statement categorization.

4.1 Statement Language-Style

We hypothesize that Wikipedia statements with news citations are similar to the language style of news, as they often paraphrase cited news articles. Different genres (such as news, recipes, sermons, FAQs, fiction . . .) differ in their linguistic properties as the different functions they fulfill influence linguistic form [6]. For example, we expect news reports (which center mostly on past events) to contain more past tense verbs than a recipe which gives instructions via verbs in the imperative. We use features that were successful in automatic genre classification including structural features via parts-of-speech as well as lexical surface cues [13].

Part of Speech Density. Frequency of part-of-speech (POS) tags, determined via the Stanford tagger, allows us to capture some of the structural properties of text. For example, news statements can be characterized by a high number of past tense verbs as well as

---

4.3 Task Overview.

In the statement categorization task, we are given a statement $s$ and the entity $e$ from which it is extracted. We compute features that exploit the language style of $s$ and the type and section structure of $e$ to categorize $s$ into one of the citation categories $c$. We learn a multi-class classifier (Section 4.3) with classes corresponding to citation categories $c$ and optimize for predicting news statements. Table 2 shows an overview of the feature list.

<table>
<thead>
<tr>
<th>feature</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>#verbs_attr</td>
<td>the number of verbs of attribution</td>
</tr>
<tr>
<td>#POS</td>
<td>the frequency of POS tags in $s$</td>
</tr>
<tr>
<td>$\lambda(s)$</td>
<td>temporal proximity of $s$ to time point</td>
</tr>
<tr>
<td>discourse</td>
<td>discourse annotations of $s$</td>
</tr>
<tr>
<td>#quotations</td>
<td>the frequency of quotations in $s$</td>
</tr>
<tr>
<td>$\theta(s, N_t)$</td>
<td>LM score of $s$</td>
</tr>
<tr>
<td>LDA($s, N_t$)</td>
<td>similarity of $s$ to a topic model</td>
</tr>
<tr>
<td>$p(s = news</td>
<td>\psi)$</td>
</tr>
<tr>
<td>$p(s = news</td>
<td>t)$</td>
</tr>
<tr>
<td>$p(s = news</td>
<td>t', \psi)$</td>
</tr>
</tbody>
</table>

Table 2: Feature list for statement categorization.
proper names. We normalize the POS tag frequency w.r.t the sum of all tags in a statement, to account for varying statement length.

**Verbs of Attribution and Quotation Marks.** News articles often report statements by persons of repute, witnesses or other sources. We approximate this by two features: Firstly, we count verbs of attribution in s, via a list of 92 such verbs (claim, tell etc) with POS tag VB* and normalize w.r.t the total number of VB*.

Secondly, we use quotation marks as a potential indicator of paraphrasing. The feature simply counts the number of quotation marks in s, normalized w.r.t the statement length.

**Temporal Proximity** $\lambda(s)$. Most Wikipedia statements with news citations refer to relatively recent events, i.e. events close to the time of the Wikipedia snapshot. Therefore, we use temporal expressions such as dates and years as distinguishing features for news statements. We use a set of hand-crafted regular expression rules to extract temporal expressions like HeidelTime [23] and Stanford’s CoreNLP [11] module news statement ratio that belong to a section or type, respectively.

(1) such as dates and years as distinguishing features for phrasing verbs of attribution in sources. We approximate this by two features: Firstly, we count often report statements by persons of repute, witnesses or other proper names. We normalize the POS tag frequency w.r.t the sum of all tags in a statement, to account for varying statement length.

**Discourse Analysis.** We use discourse connectives to annotate the statements s with explicit discourse relations based on an approach proposed by Pittler and Church [19]. The annotations belong to the categories {temporal, contingency, comparison, expansion}, following the Penn Discourse Treebank annotation [20]. Some of the explicit discourse relations are particularly interesting (i.e., temporal) as they represent a common language construct used in news articles that report event sequences. The features are boolean indicators on whether s contains a specific explicit discourse relation.

**Language Model and Topic Model Scoring.** As surface lexical features have been shown to be efficient in genre recognition [22], we compute n-gram (up to n=3) language models with Kenner-Ney smoothing (LM) from news articles $N_t$ and compute the score $\theta(s, N_t)$. This score shows how likely s can be constructed from the LM. Similarly, we compute topic models using the LDA framework [6], where the score is the jaccard similarity between s and the topic terms from $N_t$.

### 4.2 Entity-Structure Based Features

Determining if a statement requires a news citation solely on language style is not always feasible. We exploit the entity structure of e and compute the probability of statements having a news citation given its types $T(e)$ and sections $\Psi(e)$.

**Section-Type Probability.** A good indicator of the likelihood that a statement s requires a news citation is the entity type it belongs to and the section type that it appears in. For instance, for type Politician, news statements have higher density in section ‘Early Life and Career’ as these tend to be more reflected in news. To avoid over-fitting we filter out entity types with fewer than 10 statements. Similarly, we filter out section with fewer than 10 statements, and in which they belong to the same citation category.

We compute the conditional probability of having a news citation for s for an entity type $t \in T(e)$ given a section type $\psi$.

$$p(s = \text{news}|t, \psi) = \frac{\sum_{e \in e \in \text{W} \land e \in T(e)} \sum_{s \in s(e) \land s \in \text{news}} 1_s \text{ of news}}{\sum_{e \in e \in \text{W} \land e \in T(e)} |S(e, \psi)|}$$

The $p(s = \text{news}|t, \psi)$ probability is likely to be a sparse feature, so we compute type and section news-priors. We compute section $p(s = \text{news}|\psi)$ and type news-priors $p(s = \text{news}|t)$ based on the news statement ratio that belong to a section or type, respectively.

Since s is associated with an entity e, which has a set of types $T(e)$, we aggregate the computed type news-priors and the section-type joint probability into their min, max and avg probabilities.

**Type Co-Occurrence.** From the entity types $T(s)$ and $T(e)$ we measure the likelihood of type co-occurrence in news. The probability simply counts the co-occurrence between t and t’ in news statements with respect to their total co-occurrences. Examples of highly co-occurring types in news are Politician and Organization.

$$p(t', t) = \frac{\sum_{s \in e \in \text{W} \land e \in T(e)} \sum_{s \in S(e) \land t \in T(e)} 1_s \text{ of news}}{\sum_{s \in e \in \text{W} \land e \in T(e)} \sum_{s \in S(e)} 1_t \in T(s)}$$

### 4.3 Learning Framework

**Learning Setup.** Wikipedia consists of a highly diverse set of entities. A model trained on all entities is unlikely to work. For example, the types Location and Politician represent two highly divergent groups with regard to entity page structure, the statements they contain and the way they are reported in news.

We therefore learn SC for individual types in the YAGO type taxonomy. The advantages of type specific functions $SC$ is that they are trained on homogeneous entities, which helps the models predict with greater accuracy. We take only types that have more than 1000 entity instances, resulting in 672 types. The types are organized from very broad types such as (owl:Thing) to very specific types like Serie_A_Players.

To utilize the specialization and generalization in a principled manner we transform the YAGO type taxonomy (DAG) into a hierarchical DAG. This is utilized later on in order to find the right level of type granularity for learning $SC$.

We assume that the hierarchy is rooted at owl:Thing and all internal nodes are depth-consistent, i.e. all paths from the root to the node are of the same length. We obtain this by a simple heuristic whereby for every child type $\rightarrow$ parent type we remove edges where the parent’s depth level in the taxonomy is higher than the minimum level from other parent nodes.

With this hierarchical type-taxonomy, we can determine the optimal level of type granularity such that we have optimal performance in categorizing statements. For learning the type specific $SC$, we keep 10% of entity instances for evaluation and the remainder for training. It is important to note that when we learn SC for a given type, the training instances are sampled through stratified sampling from all its children types.

**Learning Model.** The functions $SC$ represent multi-class classifiers with classes corresponding to the citation categories. Since we want to predict the news category $c$ = ‘news’ with high accuracy, one question is why we do not pose this as a binary classification problem, where a statement is categorized as news or not. We used the multi-class classifiers because they give us a more balanced distribution when compared to merging all non-news statements into a single category.

Finally, we opt for Random Forests (RF) [7] as our supervised machine learning model. We experimented with other models, but the differences in performance are marginal, and RF have superior learning time. We train the models on the full feature set in Table 2

### 5. CITATION DISCOVERY

For the citation discovery task, we follow the citation policy guidelines in Wikipedia and single out three key properties on what makes a good citation.

1. the statement should be entailed by the cited news article
2. the statement should be central in the cited news article
3. the cited news article should be from an authoritative source

We approach the citation discovery for news statements as follows. We use statement \( s \) as a query (see Section 5.1) to retrieve the top-\( k \) news articles from \( \mathcal{N} \) as citation candidates for \( s \). We then classify the candidate citations as either ‘correct’ or ‘incorrect’, depending on whether they meet the above criteria of a good citation.

In order to do so, we compute features for each pair \( (s, n_i) \), w.r.t. the individual sentences of a news article \( n_i \). The feature vectors become the following \( \langle s, [\sigma_1^j, \sigma_2^j, \ldots, \sigma_{\ell}^j] \rangle \), where \( \sigma_i^j \) represents the \( j \)-th sentence from \( n_i \).

Since the number of sentences \( \sigma_i \) varies across news articles, we aggregate the individually computed features at sentence level into the corresponding min, max, average, weighted average, and exponential decay function scores as shown below.

\[
(s, \min_j F(\sigma_i^j), \max_j F(\sigma_i^j), \text{Avg}(F(\sigma_i)), \sum_a \frac{1}{j} \ast F_i; \sum F_j^7, \ldots )
\]

where \( F \) is a feature from the complete feature list in Table 3.

5.1 Query Construction

We use the statement text as query which can vary from a sentence to a paragraph. One way to improve the likelihood of obtaining good citation candidates from top-\( k \) articles is through query construction approaches (QC). It has been shown that in similar cases where the query corresponds to a sentence or paragraph, QC approaches are necessary to increase the accuracy of IR models. Henzinger et al. [13] propose several QC approaches that weigh query terms based on the \( tf-idf \) score.

We experimented with different QC approaches from [13] and their impact on finding news articles in \( \mathcal{N}^{\text{W}} \). We found that QCALBase performed best and use it in the remainder of the paper. In QCALBase, the terms extracted from the statements are weighted based on \( tf-idf \), with \( tf \) and \( idf \) are computed w.r.t. the other statements under consideration.

In principle, one should consider all retrieved articles from the result set. However, this is not only computationally expensive for our subsequent learning step but also unbalances our training set. To determine a reasonable retrieval depth, we experimented with 1000 randomly chosen statement queries with QC and determined the hit-rate at retrieval depth \( k \), i.e. whether the cited article is retrieved in the top-\( k \) articles.

Figure 4 shows the hit-rate in top-1000 with top 50 ranked query terms and with divergence from randomness query similarity measure [1] for our random sample of 1000 news statements.

![Figure 4: Hit-rate of articles in \( \mathcal{N}^{\text{W}} \) up to rank 1000 (x-axis) for 1000 news statements, respectively QCALBase queries.](image)

We focus on the top-100 retrieved news articles as potential citations for \( s \), as the achieved hit-rate beyond the top-100 shows only minor improvement. In Figure 5 we also note that the hit-rate does not go beyond 50%. We found that most of the news articles that are not retrieved are either missing or non-English articles in \( \mathcal{N}^{\text{W}} \).

5.2 Textual Entailment Features

As the citation is supposed to give close evidence for the statement’s content, in the ideal case the cited news article should fully entail the statement, i.e. the statement should be derivable from the news article. The recognition of textual entailment has been the study of extensive research in the last 10 years; cf [8] for an overview. A full treatment of entailment needs extensive world knowledge and inference rules; we here restrict ourselves to much simpler lexical and syntactic similarity methods used in baseline entailment systems and leave the extensions to future work [10].

IR Baseline Features. We use the retrieval model as a pre-filter to find candidate news articles as citations for \( s \). The retrieval model also provides us with two possible features for the learning model: firstly, a matching score of \( n_i \) for query \( s \), where the score corresponds to the divergence from randomness query similarity measure [1]. Secondly, the retrieval rank of \( n_i \). We use the IR model as our baseline and hence refer to them as baselines features.

Tree Kernel Similarity. Lexical similarity measures in many cases fail to capture the joint semantic and syntactic similarity. For this purpose, we consider the tree kernel similarity measure proposed in [14]. We first compute the dependency parse trees of \( s \) and \( \sigma_i^j \) using the Stanford tagger [24], and then compute the tree kernel, \( K(s, \sigma_i^j) \). Tree kernel similarity through the dependency parse tree measures the maximum matching subtrees between \( s \) and \( \sigma_i^j \), where the matching subtrees have the same syntactic and semantic meaning. We refer the reader to [14] for details.

LM & Topic Model Scoring. From an article \( n_i \), we compute a unigram LM and compute \( \theta(s, n_i) \) as the likelihood of \( s \) being generated from the computed LM. In addition, we compute \( n \)-gram LM (with \( n \) up to 3) from articles in \( N_i \), and compute the score \( \theta^\text{n}(s, N_i) \) accordingly.

Similarly, we compute LDA topic models [6] for entity types, specifically from articles in \( N_i \). This follows the intuition that content usually is clustered around specific topics, i.e. for type Politician most discussions are centered around politics, career, etc. The topic score is the Jaccard similarity between \( n_i \) and the topic terms.

5.3 Centrality Features

Similarity to most central news sentence. As described above we compute similarity features between \( s \) and sentences in \( n_i \). However, some sentences in \( n_i \) are more central than others. Hence, the computed features between the pairs \( \langle s, [\sigma_1^j, \sigma_2^j, \ldots, \sigma_{\ell}^j] \rangle \), do not have uniform weight. Therefore, we find the most central sentence \( \sigma_i^j \) in \( n_i \) and distinguish the computed entailment/similarity features between \( s \) and \( \sigma_i^j \).

We compute centrality of a sentence in \( n_i \) through the TextRank approach introduced in [17]. We first construct a graph \( G = (V, E) \) from \( n_i \), where \( V \) corresponds to the sentences of \( n_i \), with edges in \( E \) weighted with the Jaccard similarity between any two sentences, in this case \( \sigma_i^j \in V \). Computation of centrality for any vertex \( \sigma_i^j \) is similar to that of PageRank, with slight changes accounting for the weighted edges between vertices.

\[
\Gamma(\sigma_i) = (1 - d) + d \ast \sum_{\sigma_j \in \text{Out}(\sigma_i)} \frac{\mathbf{J}(\sigma_i, \sigma_j)}{\sum_{\sigma_k \in \text{Out}(\sigma_j)} \mathbf{J}(\sigma_j, \sigma_k)} \Gamma(\sigma_j)
\]

[10] Off-the-shelf entailment systems exist but are too slow to use at scale.
where $d$ is the damping factor ($d = 0.85$), a common value in PageRank computation. The computation converges if the difference in the score of $\Gamma(s)$ in two consecutive iterations is small.

**Relative Entity Frequency.** The importance of $e$ in $n_i$ is crucial when finding citations for $s$. This importance is partially mirrored simply in how often $e$ is mentioned in $n_i$. However, another genre-specific property of news is its inverted pyramid structure, i.e. the most important information is mentioned at the beginning of the article. We therefore measure relative entity frequency of $e$ in $n_i$, based on an approach described in [10]. It attributes higher weight to entities appearing in the top paragraphs of $n_i$, where the weight follows an exponential decay function.

$$
\phi(e, n_i) = \frac{|\rho(e, n_i)|}{|\rho(n_i)|} \sum_{\rho \in \rho(n_i)} \left( \frac{tf(e, \rho)}{\sum_{e' \neq e} tf(e', \rho)} \right)^{\frac{1}{2}}
$$

where $\rho$ represents a news paragraph from $n$ and $\rho(n_i)$ indicates the set of all paragraphs. $tf(e, \rho)$ indicates the frequency of $e$ in $\rho$. With $|\rho(e, n_i)|$ and $|\rho(n_i)|$ we indicate the number of paragraphs in which entity $e$ occurs and the total number of paragraphs.

Additionally we consider the relative entity frequency for entities in $e \in \gamma(s)$ and measure the minimum, maximum and average relative entity frequency scores.

### 5.4 News-Domain Authority Features

Wikipedia’s editing policy distinguishes clearly between more and less-established news outlets and prefers the former (see the Introduction). We therefore compute the authority of news domains w.r.t entity types and sections. We will denote the domain of the news article referred from $s$ as $D[s]$, and with $D$ any arbitrary domain.

**Type-Domain Authority.** Authority of news domains is non-uniformly distributed across types. For types such as Politician the authority of domains like BBC is higher than for types such as Athletes, where a domain specialized in sports news is more likely to be authoritative. We capture the type-domain authority as follows:

$$
p(D|t) = \sum_{D \subseteq \text{owl:Thing}} \sum_{\text{OWL:Thing} \subseteq D[t]} D = D[t] \frac{D[s]}{D[s]}
$$

**Section-Domain Authority.** We measure the authority of domains associated to certain entity sections. The density of news references across sections varies heavily. Therefore, it is natural to consider the authority of news domains for a given section.

$$
p(D|\psi) = \sum_{D \subseteq \text{owl:Thing}} \sum_{\text{OWL:Thing} \subseteq D[\psi]} D = D[\psi] \frac{D[s]}{D[s]}
$$

Note that these features compute news outlet authority with regard to current Wikipedia usage, which we seek to re-create. An alternative we intend to look at in future work is to measure authoritativeness via Wikipedia-external measures of news outlets, such as page visits or interlinkage.

### 6. STATEMENT CATEGORIZATION EVALUATION

Here we describe the evaluation of our approach for SC. Since we consider a type taxonomy, we have a hierarchy of models. Each statement belongs to an entity, which in turn is a child to a type (node) in the hierarchy. Consequently, we construct each model from training instances (statements) that are its children. We focus on two aspects of our approach (i) performance of models at varying depths, and (ii) performance of various feature classes.

We provide the detailed results for the statement categorization task and the corresponding ground-truth data at the paper URL.

#### 6.1 Experimental Setup

**Setup.** We consider 672 entity types from our Yago taxonomy, for which we learn individual SC models. We consider types that have more than 1000 entity instances. The level of granularity in the YAGO taxonomy has a maximum depth of 20, while the root type is owl:Thing containing all possible entities.

**Train/Test.** We learn the SC models using up to 90% of the entity instances of a type $t$ as training set, and the remainder of 10% for evaluation. We use stratified sampling to pick entities of type $t$ and its subtypes for the train and test set. We train and test SC models over 6 million statements coming from 1.3 million entities.

**Metrics.** We evaluate the performance of SC with precision $P$, recall $R$ and $F1$. A statement is categorized correctly if the predicted category corresponds to the ground-truth.

### 6.2 Results and Discussion

The following discussion focuses on the results for the statement categorization task for the news category. Due to space constraints we report the first three type levels in the Yago taxonomy, specifically the immediate child Legal Actor Geo of owl:Thing. The results for the remainder of the types are accessible at the URI.

Table 4 shows the results for SC models evaluated over 61k entities and trained with up to 550k entities, depending on the training size $\tau \in [1\%, 90\%]$. The results for this type represent more than 47% of the total set of entities in our evaluation dataset.

For readability we remove the wordnet prefix from the types and their numerical ID values.
The overall performance of $SC$ for all types for $\tau = 90\%$ measured through macro-average and micro-average precision are 0.48 and 0.57. Since a statement belongs to multiple types $T(s)$, we decide the category of $s$ based on majority as categorized from the individual $SC$ models.

6.2.1 Level of Type Granularity

As expected, we observe that model performance depends on the type level (cf. Table 4). A unified model from heterogeneous training instances performs poorly: the $SC$ model for the main type Legal Actor Geo achieves a precision $P=0.527$ with high variance across its subtypes. Comparing the types at depth level 3, the difference in terms of precision can go as high as 15% between Legal Actor Geo and the best performing subtype preserver.

At higher depths, performance of the $SC$ models often improves significantly as the instances belonging to a given type become more homogeneous. For example, the fine grained entity type wcat Italian footballers has a precision of $P=0.87$ and recall of $R=0.58$, which constitutes a 50% precision and a 26% recall improvement over its parent type Person. However, the performance improvement is not monotonically increasing. In some fine-grained types, there is in fact a performance reduction which can be attributed to over-fitting. This suggests that there is indeed a sweet spot in terms of choice of the best performing model for an instance. We observed that the instances that are children of person showed best performances between levels 5 and 8.

Our models perform poorly for types such as location since location pages have a lower news density. We again observe that news articles are usually centered around people and its instances benefit the most from our approach. We also observe that the performance of our approach is sensitive to the type hierarchy. The choice of YAGO as a taxonomy is due its fine-grained types. How ever, there exist many long-tail entities that are direct descendants from the higher levels and fail to leverage the homogeneity of fine-grained types. We also perform poorly on such instances.

In the YAGO taxonomy, the entities are distributed normally with a mean at depth level 8, which contains around 36% of entities. The long tail with types lower than depth level 8 accounts for 28% of entities in the YAGO taxonomy.

We focussed on the category news in our discussion and in Table 4. Performance of $SC$ models for the categories $c = \{\text{web}, \text{book}, \text{journal}\}$ and type person is $P=0.62$ and $R=0.59$, $P=0.29$ and $R=0.69$, and $P=0.25$ and $R=0.26$, respectively. The relatively high score for the web category can be attributed to the high density of statements of category web, accounting for more than 54% of the total statements. Hence, by always choosing web as the category of a statement we get an average precision of 0.54.

6.2.2 Convergence and Feature Ablation

Convergence. We measure the amount of training data required for the models to converge to optimal performance. Figure 5 shows the learning curve for some of the types reported in Table 4. We see that $SC$ models achieve nearly optimal performance early on with a sample of 7% to 10%, however, small improvements of up to a maximum of 5% are seen with the increase of training data.

Ablation. We apply a feature ablation test for the different features groups from Table 2. Figure 6 shows the results for the feature groups language style, and entity structure. The highest gain is achieved with the feature group entity structure, which reveals the challenging nature of the task where language style features cannot be applied alone.

7. CITATION DISCOVERY EVALUATION

In this section, we evaluate the citation discovery task for news statements. We perform an extensive evaluation for approximately 22k news statements and discover citations from a real-word news collection with 20 million articles in a timespan of two years.

7.1 Statement and News Collection

We limit ourselves to the subset of news statements with citations to news articles in $N^{W}$ from 2013 to 2015. The resulting set contains 22k news statements with 27k news article citations in $N^{W}$. We denote this temporal slice of news articles in $N^{W}$ by $N^{13-15}$. As finding the right citation from this preselected collection is easier than the realistic scenario of finding a citation among all possible news, we also collected all English news articles from the period [2013-08, 2015-08] from the GDelt project. We call the resulting high-coverage dataset $N^{G}$.

We merge $N^{G}$ with $N^{W}_{13-15}$ and call the resulting dataset $N = N^{W}_{13-15} \cup N^{G}$. The set $N$ contains around 20 million news articles. $N^{13-15}$ accounts for less than 1% in $N$, making the correct articles hard to find.

7.2 Evaluation Strategies

Evaluation Strategy E1: In this scenario, we, for each news statement $s$, only consider the pairs $(s, n)$, where $n \in N_s$ as correct and all other possible citations as incorrect. This allows for fully automatic evaluation but is only a lower bound for FC, as there can be additional articles that are relevant for $s$ but do not exist in $N_s$. We therefore also consider a variant E1+FP, where we consider $n' \notin N_s$ as additional correct citations if the similarity (based on the jaccard similarity) to one of the articles in $N_s$ is above 0.8.

Evaluation Strategy E2: E2 assesses the true performance of FC. In this case, apart from already existing citations for $s$ from $N_s$, we assess through crowd-sourcing the appropriateness as citations of articles $n \in N \land n \notin N_s$.\[^{13}\]

[^13]: A statement can have more than one citation.
We set up the crowd-sourcing experiment for E2 as follows. For a statement \( s \) and an article \( n \), we ask the crowd to compare \( n \) with the ground-truth article \( n^* \) and answer the question ‘Which of the two shown news articles is an appropriate citation for the statement?’. The workers are shown \( s \) as well as \( n \) and the ground-truth article in random order without an indication which one is the ground truth. We provide the following response options: (i) first, (ii) second, (iii) both, (iv) none, and (v) insufficient info. We deployed the experiment in CrowdFlower\(^{17}\) and chose only high quality workers to ensure the reliability of our experiment\(^{18}\). We removed workers who did not spend the minimum amount of two minutes to assess the appropriateness of a citation.

We collect three judgments per question. We count citations as correct which are ground-truth articles or articles which the majority of workers judge as appropriate citations.

### 7.3 Experimental Setup

**Retrieval model.** We use the retrieval model in \(^{[1]}\) via the implementation provided by Solr\(^{19}\). We use the top-100 news articles per query/statement returned by the retrieval model for feature extraction and learning approach.

**Learning Setup.** We learn classifiers specific to entity types for a total of 83 types. We limit to types that have news statements in the date range 2013-2015 and with at least 100 entity instances. From our set of 22k statements, we randomly sample statements from each entity type if they have more than 1000 instances, otherwise we take all statements. Training and testing data consist of from each entity type if they have more than 1000 instances, otherwise we take all statements.

**Learning Approach.** We learn the FC models as supervised binary classification models using random forests RF\(^{[7]}\). We predict \( (s, n) \) ∈ ‘correct’, ‘incorrect’, i.e. if a candidate news article is an appropriate citation for \( s \) or not. We optimize for the ‘correct’ class. The correct labels in training and automatic evaluation E1 are all part of \( N_{10}^{≤15} \), which makes up only 1% of our news collection \( N \). Therefore, we learn FC as a cost-sensitive classifier.

**Metrics.** We evaluate performance of FC models via precision \( P \), recall \( R \), and F1 score.

### Baselines

We consider two baselines (B1 and B2) for this task. For B1, we use the divergence from randomness model \(^{[1]}\) to retrieve news articles from \( N \) for \( s \) and simply suggest the top–1 article as citation. In B2 we learn a supervised model based on the IR baseline features (see Table 3).

### 7.4 Results and Discussion

Table 5 shows the results for all evaluation strategies for the citation discovery task. We only display detailed results for the top–10 best performing entity types out of the 83 types in our evaluation. The results in each row in Table 5 show the best performance we achieve for the individual types, while varying the variables such as the training sample size and feature number. We show results with a maximum of 60% training sample size.

We report additionally the overall performance of FC models across all 83 types through macro and micro average in the last rows in Table 5. The detailed results are accessible at the paper URL\(^{20}\).

#### 7.4.1 E1: Automated Evaluation

Table 4 shows the evaluation results for strategy E1 in the third column.

Results for E1 are encouraging given the fact that in top–100 news candidates retrieved from \( N \) only 1% of the news are ‘correct’ (on average one relevant citation in \( N_{10}^{≤15} \) per statement). Furthermore, as shown in Figure 4, the highest recall we get at top–100 is on average around 45%.

We achieve the best performance in terms of precision for the entity type football player, with precision \( P=0.80 \) and a recall of \( R=0.30 \). For F1 the best performing type in this setup is the entity type player with \( F1=0.57 \).

Using the evaluation strategy E1+FP, we consider as relevant all false positive (FP) articles which are highly similar to the ground-truth articles \( N \) (above 0.8 similarity). Even though the FP articles do not exist in our ground-truth, the high similarity to the ground-truth article are strong indicators of them being relevant citations. Using this strategy, the results improve for some of the types with up to 8% in terms of precision. For type entertainer we have an increase of 11%. In absolute numbers, by considering the highly-similar FP articles as relevant we gain an additional 757 news articles out of 12,877, i.e. an additional 6% news citations.

Baselines B1 and B2 show the difficulty of the citation discovery task. In particular, we show that standard IR models struggle with this task. Choosing only the top–1 article for citation (B1) achieves only up to \( P=0.37 \) (since we take always top–1, all evaluation metrics are the same).

#### Table 4: Results for the statement classification for entities of type yagoLegalActorGeo. Results are aggregated for the different sample ranges \( \tau \) and shown at different levels of entity types in the YAGO type hierarchy.

<table>
<thead>
<tr>
<th>Level</th>
<th>Parent Type</th>
<th>Child Type</th>
<th>( 1 ≤ \tau ≤ 10 )</th>
<th>( 10 ≤ \tau ≤ 50 )</th>
<th>( 50 ≤ \tau ≤ 90 )</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>P</td>
<td>R</td>
<td>F1</td>
</tr>
<tr>
<td>L.0</td>
<td>owl:Thing</td>
<td>Legal Actor Geo</td>
<td>0.48</td>
<td>0.36</td>
<td>0.41</td>
</tr>
<tr>
<td>L.1</td>
<td>Legal Actor</td>
<td>location</td>
<td>0.51</td>
<td>0.34</td>
<td>0.41</td>
</tr>
<tr>
<td></td>
<td></td>
<td>region</td>
<td>0.30</td>
<td>0.29</td>
<td>0.29</td>
</tr>
<tr>
<td>L.2</td>
<td>location</td>
<td>point</td>
<td>0.30</td>
<td>0.14</td>
<td>0.38</td>
</tr>
<tr>
<td></td>
<td></td>
<td>person</td>
<td>0.53</td>
<td>0.36</td>
<td>0.43</td>
</tr>
<tr>
<td>L.3</td>
<td>person</td>
<td>preserver</td>
<td>0.63</td>
<td>0.31</td>
<td>0.42</td>
</tr>
<tr>
<td></td>
<td></td>
<td>authority</td>
<td>0.53</td>
<td>0.20</td>
<td>0.29</td>
</tr>
<tr>
<td></td>
<td></td>
<td>contestant</td>
<td>0.59</td>
<td>0.43</td>
<td>0.50</td>
</tr>
<tr>
<td></td>
<td></td>
<td>leader</td>
<td>0.53</td>
<td>0.26</td>
<td>0.34</td>
</tr>
<tr>
<td></td>
<td></td>
<td>wc:Living people</td>
<td>0.55</td>
<td>0.37</td>
<td>0.44</td>
</tr>
</tbody>
</table>

\( ^{17} \)https://www.crowdflower.com

\( ^{18} \)We select workers with the highest quality as provided by the CrowdFlower platform.

\( ^{19} \)http://lucene.apache.org/solr/
Regarding B2, we see that we cannot learn well using only the IR baseline features, and perform even worse than using B1.

### 7.4.2 E2: Automated+Crowdsourced Evaluation

For E2, we report results after re-evaluating performance of FC models via gathering judgements for false positive (FP) news articles suggested as citations for s. We evaluate 11,803 false positive news article citation candidates for the top–10 entity types in Table 5 from 6.9k news statements. As reported above, workers could choose between both ground truth and our suggestion being correct, one of them or neither. The inter-rater agreement between workers was 64%. Table 6 shows how these false positives were assessed.

<table>
<thead>
<tr>
<th>Type</th>
<th>P</th>
<th>R</th>
<th>F1</th>
<th>P</th>
<th>R</th>
<th>F1</th>
<th>P</th>
<th>R</th>
<th>F1</th>
<th>#feat.</th>
<th>%train</th>
</tr>
</thead>
<tbody>
<tr>
<td>player</td>
<td>0.37</td>
<td>0.36</td>
<td>0.37</td>
<td>0.31</td>
<td>0.28</td>
<td>0.29</td>
<td>0.67</td>
<td>0.46</td>
<td>0.55</td>
<td>0.71 (5.63%)</td>
<td>0.85 (21.18%)</td>
</tr>
<tr>
<td>entertainer</td>
<td>0.32</td>
<td>0.31</td>
<td>0.31</td>
<td>0.16</td>
<td>0.18</td>
<td>0.17</td>
<td>0.70</td>
<td>0.33</td>
<td>0.45</td>
<td>0.78 (10.26%)</td>
<td>0.90 (22.22%)</td>
</tr>
<tr>
<td>causal agent</td>
<td>0.26</td>
<td>0.26</td>
<td>0.26</td>
<td>0.17</td>
<td>0.21</td>
<td>0.19</td>
<td>0.73</td>
<td>0.28</td>
<td>0.41</td>
<td>0.77 (5.19%)</td>
<td>0.88 (17.05%)</td>
</tr>
<tr>
<td>location</td>
<td>0.21</td>
<td>0.19</td>
<td>0.20</td>
<td>0.21</td>
<td>0.23</td>
<td>0.22</td>
<td>0.55</td>
<td>0.26</td>
<td>0.35</td>
<td>0.62 (11.29%)</td>
<td>0.83 (33.73%)</td>
</tr>
<tr>
<td>artist</td>
<td>0.31</td>
<td>0.31</td>
<td>0.31</td>
<td>0.24</td>
<td>0.27</td>
<td>0.28</td>
<td>0.67</td>
<td>0.21</td>
<td>0.32</td>
<td>0.67 (31.31%)</td>
<td>0.85 (21.18%)</td>
</tr>
<tr>
<td>football player</td>
<td>0.31</td>
<td>0.31</td>
<td>0.31</td>
<td>0.29</td>
<td>0.38</td>
<td>0.33</td>
<td>0.80</td>
<td>0.30</td>
<td>0.43</td>
<td>0.80</td>
<td>0.90 (11.11%)</td>
</tr>
<tr>
<td>wcet Living people</td>
<td>0.27</td>
<td>0.26</td>
<td>0.26</td>
<td>0.21</td>
<td>0.18</td>
<td>0.2</td>
<td>0.67</td>
<td>0.23</td>
<td>0.34</td>
<td>0.70 (4.29%)</td>
<td>0.85 (21.18%)</td>
</tr>
<tr>
<td>creator</td>
<td>0.34</td>
<td>0.32</td>
<td>0.33</td>
<td>0.25</td>
<td>0.24</td>
<td>0.24</td>
<td>0.74</td>
<td>0.25</td>
<td>0.38</td>
<td>0.74</td>
<td>0.91 (18.68%)</td>
</tr>
<tr>
<td>organism</td>
<td>0.29</td>
<td>0.28</td>
<td>0.28</td>
<td>0.22</td>
<td>0.19</td>
<td>0.2</td>
<td>0.69</td>
<td>0.30</td>
<td>0.41</td>
<td>0.70 (1.43%)</td>
<td>0.83 (16.87%)</td>
</tr>
<tr>
<td>person</td>
<td>0.26</td>
<td>0.24</td>
<td>0.25</td>
<td>0.21</td>
<td>0.23</td>
<td>0.22</td>
<td>0.64</td>
<td>0.35</td>
<td>0.46</td>
<td>0.66 (3.03%)</td>
<td>0.85 (24.71%)</td>
</tr>
<tr>
<td>macro-average</td>
<td>0.24</td>
<td>0.18</td>
<td></td>
<td>0.62</td>
<td></td>
<td></td>
<td>0.65 (4.6%)</td>
<td></td>
<td></td>
<td>0.82 (24.40%)</td>
<td></td>
</tr>
<tr>
<td>micro-average</td>
<td>0.25</td>
<td>0.21</td>
<td></td>
<td>0.67</td>
<td></td>
<td></td>
<td>0.71 (5.5%)</td>
<td></td>
<td></td>
<td>0.86 (22.00%)</td>
<td></td>
</tr>
</tbody>
</table>

Table 5: Top–10 best performing entity types for the FC task. E1+FP and E2 columns show the improvement for P over E1. Right most column shows the configuration with which we learn the FC models.

Statement Categorization. We set up statement categorization as a majority voting categorization. For each statement and the type specific classifiers SC we predict the category and pick the category that has the majority of votes. In contrast to the statement categorization in Section 6, where the original task aimed at showing for which types this task can be performed accurately, we now aim to set up citation discovery in an automated manner.

Based on the ground-truth, 340 out of the 1000 statements were news statements. We categorize 368 as news statements, out of which 263 are correct, i.e. P=0.72 and R=0.77. It is interesting to see that we can leverage additional information through majority voting, where for the same statement and its associated types we can predict with high accuracy the citation category label of s.

### Citation Discovery.

For the citation discovery task we ran it based on the generic FC model trained on statements belonging to all types, namely owl:Thing. We could use the type specific FC, with additional costs for computing type specific features.

In the second task, from the 368 statements classified as news statements, we ran the citation discovery model FC. We are able to suggest 549 news citations for 78 statements. Based on crowdsourcing evaluation, we suggest 346 relevant citations, i.e. a precision of P=0.63, out of which 200 citations are citations that were preferred over existing ones in the ground-truth. For 146 cases the citations we suggest are considered to be equally appropriate as the existing ones in the ground-truth, for 116 citations the ground-truth ones were preferred over the ones we suggested. Note that our FC models suggest citations for s only in case they fulfill the criteria in Section 6 thus, enforcing high accuracy.

9. RELATED WORK

**Citation Sources.** Ford et al. [12] analyze the citation behavior of Wikipedia editors with respect to their adherence to the citation guidelines. They investigate what types of sources are most often cited, i.e. primary, secondary and tertiary as defined in Wikipedia. They conclude that news are one of the top cited source type in secondary type, while they see a growing trend of primary sources due to their persistence on the web, contrary to the policies of preferring secondary sources. Luyt and Tan [15] analyze a subset of history pages and show that citations are biased towards a specific group of sources. [12] emphasize the importance of citations in Wikipedia as a means to ensure the quality of entity pages.

**Wikipedia Quality.** Anderka et al. [2] propose an approach to predict quality flaws in Wikipedia pages. A quality flaw in Wikipedia is usually annotated with specific cleanup tags. They train a model to predict quality flaws, where among the top–10 quality flaws they identify unreferenced, refimprove, primary sources as some of the most serious flaws. Our work is complementary to

### 8. PIPELINE EVALUATION

For the evaluation of both tasks in a pipeline scenario, we randomly sample 1000 statements from all categories and run the process of citation discovery through both steps. Each statement is associated with multiple entity types, as they are extracted from e where T(e) is a set of types. For the statement categorization task we perform the evaluation based in our ground-truth, for the citation discovery we evaluate the suggested citations as in evaluation strategy E2. Note, that here in the evaluation we pair we have a news article (that we suggest) and a resource that can be of any type including, book, web, journal.

```python
import pandas as pd
import numpy as np

# Read the evaluation data
evaluation_data = pd.read_csv('evaluation_data.csv')

# Measure the precision and recall
precision = evaluation_data['Precision'].mean()
recall = evaluation_data['Recall'].mean()

# Calculate the micro- and macro-averaged F1 scores
micro_f1 = (2 * precision * recall) / (precision + recall)
macro_f1 = np.mean(evaluation_data['F1']

# Print the results
print(f'Precision: {precision:.2f}, Recall: {recall:.2f}, Micro-F1: {micro_f1:.2f}, Macro-F1: {macro_f1:.2f}

# Additional analysis

# Compute the top-10 performing entity types
best_entity_types = (evaluation_data['Entity Type']

# Calculate the improvement over the baseline
improvement = (evaluation_data['Baseline'] - evaluation_data['Our Method'])

# Print the top-10 best performing entity types
print('Top-10 Best Performing Entity Types:

<table>
<thead>
<tr>
<th>Entity Type</th>
<th>Improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>person</td>
<td>0.67</td>
</tr>
<tr>
<td>football player</td>
<td>0.31</td>
</tr>
<tr>
<td>creator</td>
<td>-0.25</td>
</tr>
<tr>
<td>artist</td>
<td>0.24</td>
</tr>
<tr>
<td>location</td>
<td>0.21</td>
</tr>
<tr>
<td>causal agent</td>
<td>0.18</td>
</tr>
<tr>
<td>entertainer</td>
<td>0.17</td>
</tr>
<tr>
<td>player</td>
<td>0.16</td>
</tr>
<tr>
<td>wcet Living</td>
<td>0.15</td>
</tr>
<tr>
<td>organism</td>
<td>0.14</td>
</tr>
</tbody>
</table>
```
their since we aim at finding appropriate citations for Wikipedia statements, thereby improving the quality of Wikipedia pages.

**Wikipedia Enrichment.** Sauper and Barzilay \cite{Sauper2013} propose an approach to automatically generate complete entity pages for a specific entity type. The approach is trained on already-populated entity pages of a specific type by learning templates about the section structure at the type level. For a new entity page, they extract documents through Web search (with entity and section title as a query) and identify the most relevant paragraphs to add in a section. Fetahu et al. in \cite{Fetahu2012} proposed an approach for suggesting news articles for a Wikipedia entity and entity section. They first identify news articles that are important to an entity and in which the entity is salient, and further identify the most appropriate section to suggest the article. In case of a missing section, a new section is added by exploiting the section structure from the entity type.

This work differs from \cite{Sauper2013,Fetahu2012} as we do not add content or suggest news articles to a complete section in an entity page, but rather provide citations to already existing statements.

**Cumulative Citation Recommendation (CCR).** TREC introduced the CCR task in the Knowledge base acceleration track in 2012. For a stream of news and social media content and a target entity from a knowledge base (Wikipedia), the goal of the task is to generate a score for each document based on how pertinent it is to the input entity. Balog et al. \cite{Balog2012} propose approaches that find entity mentions in the document collection and rank them according to how central the entity is in the respective documents. This however is a filtering task for documents towards checking if they are relevant for a pre-defined set of entities. In contrast, in our task we aim at finding news citations as evidence for Wikipedia statements.

### 10. CONCLUSIONS

In this work we define and attempt to solve the automatic news citation discovery problem for Wikipedia. We define two tasks – sentence categorization and the citation discovery – towards finding the correct news citation for a given Wikipedia statement. For the sentence categorization task we learn a multi-class classifier to predict if a statement requires a news statement. Further, for the news citation discovery problem, we first find the likely candidates by a retrieval model over a real-world news collection followed by a binary classification for the most likely candidate. We find that statement categorization is a hard problem due to lack of context for the NLP-based features to perform well. However, the Wikipedia page and its type structure provides important cues towards accurate classification. On the other hand we perform well on the citation discovery task with 67% precision (for top-categories) on the automated evaluation, which further improves to over 80% when crowdsourced. This shows that we not only identify the correct ground truth articles present in Wikipedia, but in some cases our suggestions are a better fit compared to the sources in Wikipedia.

### Acknowledgments.

This work is funded by the ERC Advanced Grant ALEXANDRIA (grant no. 339233).

### 11. REFERENCES

\begin{thebibliography}{99}


\bibitem{Finkel2005} J. R. Finkel, T. Grenager, and C. D. Manning. Incorporating non-local information into information extraction systems by gibbs sampling. In \textit{43ed ACL}, 2005, USA.


\bibitem{Kate2008} R. J. Kate. A dependency-based word subsequence kernel. In \textit{2008 EMNLP}, Honolulu.


\bibitem{Püttler2009} E. Püttler and K. W. Church. Using word-sense disambiguation methods to classify web queries by intent. In \textit{2009 EMNLP}, Singapore.


\end{thebibliography}