

Unsupervised Named Entity Resolution

Ioannis P. Klapaftis

Department of Computer Science
University of York, YO10 5DD
York, United Kingdom
giannis@cs.york.ac.uk

Suresh Manandhar

Department of Computer Science
University of York, YO10 5DD
York, United Kingdom
suresh@cs.york.ac.uk

Abstract—Resolving the ambiguity of person, organisation and location names is a challenging problem in the Natural Language Processing (NLP) area. This problem is usually formulated as a clustering problem, in which the target is to group mentions of the same entity into the same cluster. In this paper, we present a different approach based on the *Distributional Hypothesis* and edit distance, which associates an ambiguous entity to its corresponding entry in the Wikipedia knowledge base.

We experiment with two types of contextual features, i.e. bag-of-words and bigrams, as well as with another source of information, i.e. the edit distance between an entity mention and a Wikipedia article’s title. Our experiments show that the combination of these types of knowledge offers a superior performance than each one individually or any subset of them, in effect leading to the conclusion that they are able to capture non-overlapping information that is essential for this task.

Index Terms—Named Entity Resolution, Named Entity Ambiguity, Natural Language Processing

I. INTRODUCTION

Entity resolution is a highly challenging problem with a variety of applications. For instance, entity resolution is important for web search applications, where the target is to identify a particular named entity on the web along with information that describes that entity. In the same vein, entity resolution is essential from a security point of view, in which the target would be to identify the different entity mentions (first names, surnames, nicknames) of persons or organisations that exhibit a criminal behaviour. Additionally, the disambiguation of entities would possibly be beneficial for identifying different types of relations between persons, organisations, locations and other types of entities, in effect providing useful information for a variety of security-related tasks.

The disambiguation of an entity is a difficult task for two main reasons. Firstly, an entity can be mentioned in a variety of ways. For example, many organisations and people are often referred by their initials, i.e. *BA* for

British Airways or *MJ* for *Michael Jackson*. Secondly and most importantly, entity mentions are ambiguous, i.e. they might refer to different entities. For example, *BA* might also refer to *Bosnian Airlines*, while *MJ* might also refer to *Michael Jordan*. In the same vein, *Washington* might refer to the capital of USA, to a newspaper (*Washington daily*), or to *George Washington*.

In this paper, we present an unsupervised method for named entity resolution that associates a target ambiguous entity mention to its corresponding and unique knowledge base entry. Figure 1 shows the conceptual architecture of our method. The knowledge base consists of 818741 Wikipedia¹ articles, whose text and titles are indexed by the Lucene search engine².

Given a target entity mention and the corresponding article in which that mention appears, the proposed method queries Lucene in order to get a list of candidate Wikipedia entries. In the next step, the *Article Scoring Method* (Figure 1) calculates the similarity between each Wikipedia candidate entry and the target article (in which the target entity mention appears). Finally, the Wikipedia entry with the highest similarity to the target article is assigned to the target entity mention.

To calculate the similarity between each Wikipedia article and the input one, we combine three different approaches, i.e. *Bag-of-Words Scoring*, *Bigram Scoring* and *Edit Distance*. The first one associates an article with a vector of weighted features. These features are the words occurring in an article. The second associates an article with another vector of weighted features. These features are pairs of consecutive words (bigrams) occurring in an article. Finally, the third approach, *Edit Distance*, calculates the string distance between the input entity mention and the title of a Wikipedia article.

We evaluate our approach in a set of ambiguous entity

¹<http://www.wikipedia.com> [Accessed 01/03/2010]

²<http://lucene.apache.org/> [Accessed 20/02/2010]

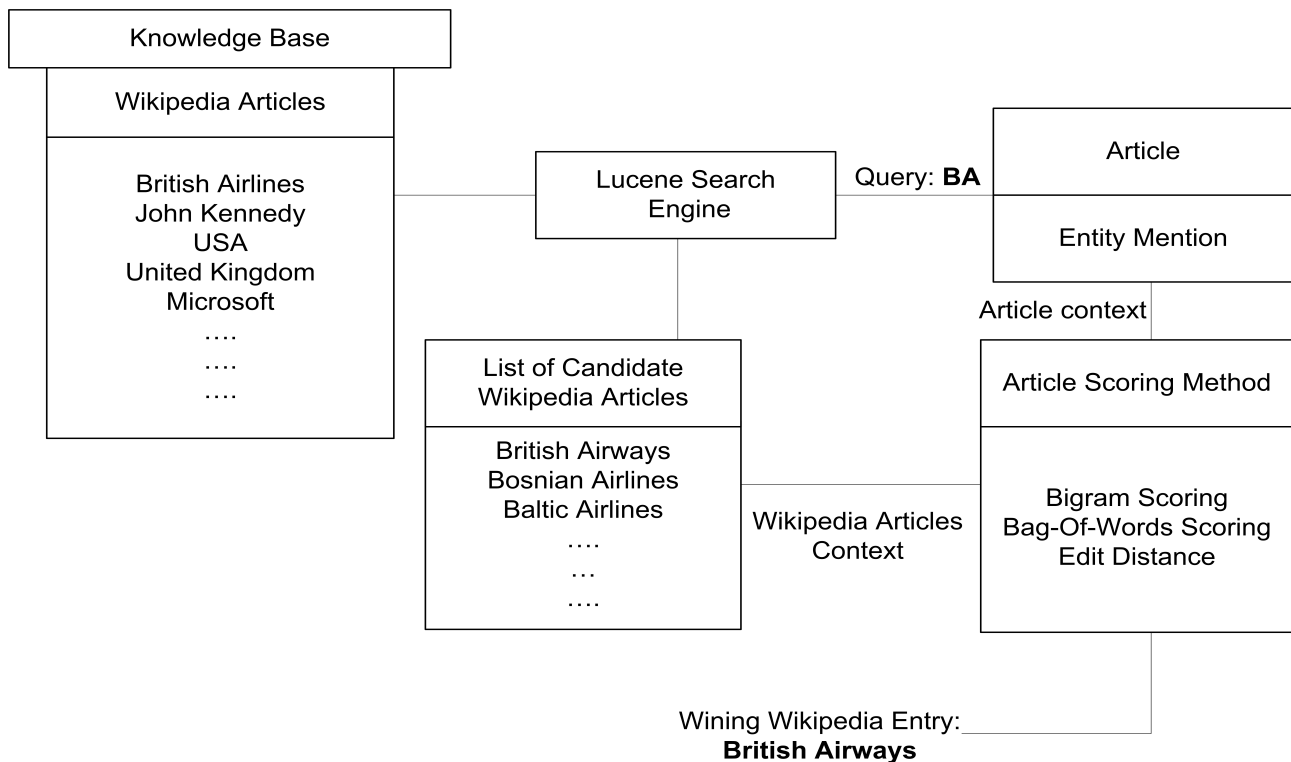


Fig. 1. Conceptual architecture of the proposed method

mentions that appear in a variety of news articles. Both the test dataset as well as the Wikipedia knowledge base were provided by the Text Analysis Conference (TAC) 2009 Knowledge Base Population Track (KBP) [1]. The evaluation shows that the combination of bigram, bag-of-words and edit distance scoring offers a superior performance than each one individually, in effect leading to the conclusion that these scoring methods provide non-overlapping information, essential for this task.

II. RELATED WORK

Resolving the ambiguity of named entity mentions is a relatively new area within Natural Language Processing, which has been tackled in the Web People Search (WePS) workshop series [2], as well as in the TAC-KBP 2009 challenge [1]. The main difference between WePS and TAC-KBP is that the former only focuses on the ambiguity of persons' names, while the latter also includes organizations and geopolitical entities.

Additionally, WePS formulates the problem of entity resolution as a clustering problem, in which the target is to cluster documents referring to the same entity. In contrast, TAC-KBP requires the association of an ambiguous entity mention to its corresponding entry in an already established Knowledge Base (KB). If an entry

does not exist in the knowledge base, then a potential system should return NIL. In our work, we do not deal with entities that do not have an associated entry in the KB. In contrast, we focus on named entities for which there is always an associated entry in KB.

Mann and Yarowsky [3] developed an unsupervised method for personal name disambiguation that exploits a rich set of biographic features, such as birth date, birth place and others. A manually constructed fact pair (e.g. *Barrack Obama, 1961*) is given as a query to a web search engine, and the sentences returned from that search are processed in order to generate a set of patterns matching the initial query. These patterns are then applied to extract biographical details from the data. In the next step, each ambiguous entity mention is associated with a vector of features, i.e. the extracted biographical facts, and a clustering process is applied to the constructed vectors. In each iteration of the clustering, the two most similar clusters are merged to produce a new cluster. Their algorithm iterates until all ambiguous entity mentions have been assigned to a cluster.

Bagga and Baldwin [4] proposed a method that relies on the vector space model. In their work each ambiguous entity mention is associated with a vector of weighted features. These features are the words that occur within a

55 token window around the ambiguous entity mention. Given a set of ambiguous entity mentions, the similarity between their corresponding features vectors is calculated, and if the similarity value is above a predefined threshold, then these entity mentions are considered to be co-referent. Evaluation of their method was performed on the *John Smith* corpus that consists of 197 articles talking about 35 different *John Smiths*.

Gooi and Allan [5] re-implemented the Bagga and Baldwin approach [4] and compared it against their method, which performs agglomerative clustering on the extracted feature vectors. Their conclusion was that agglomerative clustering performs better in most of the experiments they have conducted.

III. METHOD

Let E be an entity mention in an article A . In Figure 1, E is *BA*, while A is an article containing information about potential strikes of *British Airways* staff. Our target is to disambiguate E and associate it with its corresponding Wikipedia entry, i.e. *British Airways*.

A. Lucene Interface

The first step of our method is to get a list of Wikipedia articles that could potentially contain the entry associated with the input entity mention. Given that searching for the entity mention in the text of all the Wikipedia articles is a time-consuming task, we use Lucene to index these articles in order to speed-up the search process.

Specifically, the input query to Lucene search engine is the target entity mention, while the output of Lucene is a list of documents, $L = D_0 \dots D_n$, ranked by their relevance to the input query. The score of Wikipedia article D_i for entity mention E is the cosine distance between the document and query vectors. The features of these vectors are stemmed words occurring in the document and the query, weighted by *Term Frequency/Inverse Document Frequency* (TF.IDF). We only consider the top n documents returned by Lucene, where n was set equal to one thousand documents.

B. Scoring Method

The target at this stage is to identify the Wikipedia Article that most likely explains the target entity mention by exploiting: (1) the distributional properties of the context surrounding the entity mention, and (2) the edit distance between the target entity mention and a candidate Wikipedia article. To calculate the similarity between each Wikipedia article and the input one, we

Bigram	Frequency
BA_have	1
have_announce	1
announce_the	1
the_cancellation	1
cancellation_of	1
of_two	1
two_flight	1

TABLE I
EXTRACTED BIGRAMS

combine three different approaches, i.e. *Bigram Scoring*, *Bag-of-Words Scoring* and *Edit Distance*.

$$\begin{aligned} sim(A, D_i) &= p_0 * BG(A, D_i) \\ &+ p_1 * BW(A, D_i) \\ &+ p_2 * \frac{1}{ED(E, T(D_i))} \end{aligned} \quad (1)$$

Equation 1 defines the similarity between article A , i.e. the context of the input entity mention, and a Wikipedia article D_i , where $BG(A, D_i)$ is the similarity between A and D_i according to the bigram scoring, $BW(A, D_i)$ is the similarity between A and D_i according to the bag-of-words scoring, and $ED(A, T(D_i))$ is the edit distance between the title $T(D_i)$ of D_i and the entity mention E . Parameters p_0, p_1, p_2 reflect the confidence that should be assigned to its similarity approach.

1) *Bigram Scoring*: In our setting a bigram is a pair of two adjacent words. For example, given the sentence *BA has announced the cancelation of two flights*, we would extract the bigrams shown in Table I. Note that the words within the extracted bigrams are lemmatised. The main motivation for using bigrams as features is the fact that bigrams are less ambiguous than simple words, hence they have a higher discriminating ability. On the other hand, bigram features are more sparse. In our method, we only extract bigrams, whose words do not appear in a stop list.

To calculate the bigram similarity, $BG(A, D_i)$, between a Wikipedia article D_i and the input article A , we associate each one of them with a vector of features. These features are the extracted bigrams weighted by their frequency of occurrence in the corresponding article. Let V_A be the vector of bigrams for the input article A , and V_{D_i} the vector of bigrams for the Wikipedia article D_i . The similarity, $BG(A, D_i)$, is the cosine of their respective vectors (Equation 2, where n is the number of extracted bigrams).

$$BG(A, D_i) = \frac{\sum_{k=1}^n V_A(k) * V_{D_i}(k)}{\sqrt{\sum_{k=1}^n V_A(k)^2 * \sum_{k=1}^n V_{D_i}(k)^2}} \quad (2)$$

2) *Bag-of-Words Scoring*: The second scoring method for calculating the similarity between the input article A and a Wikipedia article D_i is based on a bag-of-words model. In this model, each input or KB article is associated with a vector of features, i.e. the nouns appearing in the articles. These features are weighted using the log-likelihood ratio (G^2) [6] that attempts to identify the most important nouns for a given article. Let U_A be the vector of nouns for the input article and U_{D_i} the vector of nouns for the Wikipedia article D_i . As in the bigram scoring method, the similarity, $BW(A, D_i)$, is the cosine of their respective vectors.

In the following, we describe the process of weighting the noun features of A (resp. D_i) using the G^2 ratio. Initially, article A (resp. D_i) is POS-tagged using the *GENIA* tagger [7]. Following the example in [8], [9], only nouns are kept and lemmatised, since they are less ambiguous than verbs, adverbs or adjectives.

Let rc be a large reference corpus. In our work we have used the British National Corpus³ (BNC). Our aim is to check if the distribution of a word w in A (resp. D_i), is similar to the distribution of w in rc , i.e. $p(w|A) = p(w|rc)$ (resp. $p(w|D_i) = p(w|rc)$) (null hypothesis). If that is true, G_2 will have a small value, hence w will be assigned a low weight as it offers a limited discriminating ability. In contrast, if the distribution of w in A (resp. D_i) is significantly different than its distribution in rc , then this would possibly mean that w is an important word, hence its G^2 weight would be relatively high.

For each word w , which appears in A (resp. D_i) we construct two contingency tables. The first one contains the observed values of w in A (resp. D_i) and rc (Table II). For example in Table II for the target entity mention BA , we observe that the word *airline* appears 213 times in A and 2439 times in rc . The remaining frequency of other words in A is 23729 and in rc 24038639.

The second (ET) contains the expected values under the model of independence (Table III). Given the two contingency tables, we can calculate G^2 (Equation 3), where n_{ij} is the i, j cell of OT and m_{ij} (Equation 4) is the i, j cell of ET, and $N = \sum_{i,j} n_{ij}$.

³The British National Corpus. Distributed by the Oxford University Computing Service.

Observed Values (OT)	Target Article (A)	Reference Corpus (rc) BNC
Freq. of <i>airline</i>	213 (n_{11})	2439 (n_{12})
Total Freq. of remaining words	23279 (n_{21})	24038639 (n_{22})

TABLE II
CONTINGENCY TABLE FOR OBSERVED VALUES (OT) FOR TARGET ARTICLE A AND CONTEXT WORD *airline*

Observed Values (OT)	Target Article (A)	Reference Corpus (rc) BNC
Freq. of <i>airline</i>	2.58 (m_{11})	2649.4 (m_{12})
Total Freq. of remaining words	23489.4 (m_{21})	2.403e7 (m_{22})

TABLE III
CONTINGENCY TABLE FOR EXPECTED VALUES (ET) FOR TARGET ARTICLE A AND CONTEXT WORD *airline*

$$G^2 = 2 * \sum_{i,j} n_{ij} * \log\left(\frac{n_{ij}}{m_{ij}}\right) \quad (3)$$

$$m_{ij} = \frac{\sum_{k=1}^2 n_{ik} * \sum_{k=1}^2 n_{kj}}{N} \quad (4)$$

3) *Edit Distance*: The two scoring methods described in the previous sections exploit the contextual relatedness between the input and each KB article. The edit distance or Levenshtein distance [10], $ED(E, T(D_i))$, between the entity mention E and the title $T(D_i)$ of a Wikipedia article D_i is metric that measures how dissimilar are the corresponding strings by counting the minimum number of edits that are needed in order to transform one of the strings into the other. An edit operation in this metric is considered to be the insertion, substitution and deletion of a character.

Our intuition is that entity mentions are likely to be similar to the title of their corresponding Wikipedia entry. An example would be the entity mention *British Airlines PLC* and the corresponding Wikipedia entry *British Airlines*. Their edit distance is equal to 4, since in order to transform the first into the second one we need to delete three word characters (*PLC*) and one space character. In contrast, the edit distance between the entity mention *British Airlines* and the corresponding Wikipedia entry *Bosnian Airlines* is 5. Note that this distance is transformed to a similarity measure with a

range from 0 to 1 by taking the $\frac{1}{ED(E,T(D_i))}$ in Equation 1.

IV. EVALUATION

A. Experimental Setting & Datasets

We evaluate the proposed method on a set of ambiguous entities of the TAC KBP test dataset [1]. The dataset consists of 1675 ambiguous entity mentions, while the knowledge base consists of 818741 Wikipedia entries.

One hundred out of the 1675 instances were randomly chosen to fine-tune the parameters of our method and the rest of instances were used for evaluation. Specifically, parameters p_0 , p_1 and p_2 that control the contribution of each type of information in the final scoring formula (Equation 1) were assigned the following values $p_0 = 0.4$, $p_1 = 0.1$ and $p_2 = 0.5$. The accuracy of the proposed method in the data used for fine-tuning was 82%.

We use the standard measure of F-Score (harmonic mean of precision and recall) in order to assess the performance of our approach. In our setting F-Score is equal to recall and precision, since we have always returned an answer for a given ambiguous entity instance. In particular, let C be the number of correctly disambiguated ambiguous entity instances and N the total number of ambiguous instances. F-Score is defined in Equation 5.

$$F - Score = \frac{C}{N} \quad (5)$$

B. Baselines

The main target of our evaluation is to test whether the combination of the aforementioned types of information, i.e. bag-of-words, bigrams and edit distance offers an improved performance as opposed to each one individually. A second target is to quantify the contribution of each type of information in the overall performance.

To this end, our first baseline, *BoW*, performs entity resolution by only considering a bag-of-words model, where words are weighted using the log-likelihood ratio as described in Section III-B2. Our second baseline, *BiG*, performs entity resolution by only considering the bigram scoring method as described in Section III-B1, while the third baseline, *ED*, ranks documents according to the string similarity of the input entity mention E to title of a Wikipedia article $T(D_i)$ ($\frac{1}{ED(E,T(D_i))}$).

Finally, given that the proposed method depends on the Lucene returned documents, we also include a baseline, in which we measure the maximum F-Score that our

System	Performance (%)
Proposed method (ED + BiG + BoW)	61.1
ED + BiG	58.27
ED	47.2
BiG	20
BoW	8.3
<i>Lucene</i>	90

TABLE IV
EVALUATION RESULTS.

method can achieve by considering whether the correct Wikipedia entry exists in the Lucene returned documents for a given input entity mention. This baseline is referred as *Lucene*.

C. Results

Table IV shows the results of our evaluation. As can be observed, the proposed method, combining bigram features, word features and edit distance, achieves the highest performance, i.e. 61.1%. The edit distance, *ED* achieves an F-Score of 47.2%, which is significantly higher than the F-Score of *BiG* and *BoW*. These performance differences are statistically significant using the McNemar’s test at 95% confidence level.

This result indicates that edit distance is the most significant source of information in this task. The baseline, *BiG*, achieves a significantly higher F-Score than *BoW*, which verifies our initial intuition that bigram features are less ambiguous than word features, although they suffer from data sparsity. Overall, the first conclusion of our experiment is that the most reliable source of information is the edit distance, followed by the bigram and then the bag-of-words contextual features.

Despite that, the combination of bigram features with edit distance (*ED + BiG*) improves upon the F-Score of the latter. Specifically, our experiment in which entity resolution takes place by only considering *ED* and *BiG*, i.e. setting the parameter p_1 equal to 0, and keeping the ratio $\frac{p_0}{p_2}$ fixed, showed that the inclusion of bigram features improved the performance of *ED* by 11%, leading to an overall performance of 58.27%. The performance difference between *ED + BiG* and *ED* is statistically significant (McNemar’s test, 95% confidence level). This results demonstrates that bigram features offer semantic information that cannot be captured by the simple *ED* model, in effect leading to an improved F-Score.

In the same vein, we observe that the proposed method, combining *ED*, *BiG* and *BoW*, achieves a higher

performance than *ED + BiG*. This performance difference, equal to 2.83% (statistically significant, McNemar’s test, 95% confidence level), is caused by the inclusion of the word features in the *ED + BiG* model. Although, the *BoW*, is the least reliable, achieving only 8.3% F-Score, we observe that its weighted inclusion in Formula 1 compensates for its ambiguous features and offers semantic information than cannot be captured by bigram features due to data sparsity.

Overall, the second conclusion of our evaluation is that the three types of knowledge considered by the proposed method capture non-overlapping information, which when combined (in a weighted framework) leads to an improved performance as opposed to each one individually or any subset of them. Finally, in Table IV we observe that the maximum F-Score that our method could capture is 90%. This results indicates that there is a large space for improving both the the proposed method as well as Lucene’s ranking by replacing the standard *TF.IDF* with more solid statistical models such as log-likelihood.

V. CONCLUSION & FUTURE WORK

In this paper, we have presented an unsupervised method for disambiguating entity mentions found in raw text and associating them with their corresponding entry in a knowledge base. Our method combines word and bigram features along with the edit distance between the target entity and the title of a knowledge base article.

Our evaluation has shown that the weighted contribution of each type of information offers a superior performance than each one individually or any subset of them, in effect leading to the conclusion that they are able to capture non-overlapping information that is essential for this task.

Our future work, includes the exploitation of other types of contextual features surrounding an ambiguous entity mention, such named entities and syntactic dependencies as well as the evaluation of the contribution of each type of features. Furthermore, we are also working on exploiting these features in a graph-based framework, since graph-based method have shown to be effective in similar tasks such as word sense disambiguation and sense induction.

ACKNOWLEDGMENTS

This work is supported by the European Commission via the EU FP7 INDECT project, Grant No. 218086, Research area: SEC-2007-1.2-01 Intelligent Urban Environment Observation System. We would also like to

thank the TAC-KBP organizers for releasing the Gold Standard data for this task.

REFERENCES

- [1] M. P. and D. H., T., “Overview of the tac 2009 knowledge base population track,” in *TAC 2009 Workshop*. Gaithersburg, Maryland USA: National Institute of Standards and Technology, 2009.
- [2] J. Artiles, J. Gonzalo, and S. Sekine, “The semeval-2007 weps evaluation: Establishing a benchmark for the web people search task,” in *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*. Prague, Czech Republic: Association for Computational Linguistics, June 2007, pp. 64–69. [Online]. Available: <http://www.aclweb.org/anthology/W/W07/W07-2012>
- [3] G. S. Mann and D. Yarowsky, “Unsupervised personal name disambiguation,” in *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003*. Morristown, NJ, USA: Association for Computational Linguistics, 2003, pp. 33–40.
- [4] A. Bagga and B. Baldwin, “Entity-based cross-document coreferencing using the vector space model,” in *Proceedings of the 17th international conference on Computational linguistics*. Morristown, NJ, USA: Association for Computational Linguistics, 1998, pp. 79–85.
- [5] C. H. Gooi and J. Allan, “Cross-document coreference on a large scale corpus,” in *HLT-NAACL 2004: Main Proceedings*, D. M. Susan Dumais and S. Roukos, Eds. Boston, Massachusetts, USA: Association for Computational Linguistics, May 2 - May 7 2004, pp. 9–16.
- [6] T. Dunning, “Accurate methods for the statistics of surprise and coincidence,” *Comput. Linguist.*, vol. 19, no. 1, pp. 61–74, 1993.
- [7] Y. Tsuruoka and J. Tsujii, “Bidirectional inference with the easiest-first strategy for tagging sequence data,” in *HLT ’05: Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*. Morristown, NJ, USA: Association for Computational Linguistics, 2005, pp. 467–474.
- [8] E. Agirre, D. Martínez, O. L. de Lacalle, and A. Soroa, “Two graph-based algorithms for state-of-the-art wsd,” in *EMNLP ’06: Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*. Morristown, NJ, USA: Association for Computational Linguistics, 2006, pp. 585–593.
- [9] E. Agirre and A. Soroa, “Ubc-as: a graph based unsupervised system for induction and classification,” in *SemEval ’07: Proceedings of the 4th International Workshop on Semantic Evaluations*. Morristown, NJ, USA: Association for Computational Linguistics, 2007, pp. 346–349.
- [10] V. I. Levenshtein, “Binary codes capable of correcting deletions, insertions, and reversals,” *Soviet Physics Doklady*, vol. 10, no. 8, pp. 707–710, 1966.