TextRank – Based Keyword Extraction for Constructing a Domain-Specific Dictionary

Sridevi Bonthu[1](✉), Hema Sankar Sai Ganesh Babu Muddam[2], Koushik Varma Mudunuri[1], Abhinav Dayal[1], V. V. R. Maheswara Rao[3], and Bharat Kumar Bolla[4]

[1] Computer Science and Engineering Department, Vishnu Institute of Technology, Bhimavaram, Andhra Pradesh, India
sridevi.b@vishnu.edu.in
[2] Tata Consultancy Services, Synergy Park, Hyderabad, India
[3] Computer Science and Engineering Department, Shri Vishnu Engineering College for Women, Bhimavaram, Andhra Pradesh, India
[4] University of Arizona, Tucson, AZ 85721, USA

Abstract. Extracting domain-related keywords from text documents is a crucial task in both Information Retrieval and Natural Language Processing (NLP). This paper presents an approach that combines the TextRank algorithm with various NLP techniques to effectively identify domain-specific keywords. Our method utilizes the power of unsupervised graph-based ranking algorithms and the semantic understanding of NLP models to extract key terms that are highly relevant to a specific domain. The work is carried out on an arXiv research abstract dataset. This work preprocesses the input text to capture linguistic features, extracts the keywords using TextRank and POS filtering approaches, extracts the definitions and finally evaluates the performance. The performance of the extracted keywords is done with the help of manually annotated labels. The proposed method has obtained 83% accuracy. The proposed approach is flexible and adaptable to different domains, as it can be trained on domain-specific data to further improve its performance.

Keywords: Extraction · TextRank · POS tagging · Text mining · domain-specific dictionary · Natural Language Processing

1 Introduction

The proliferation of digital data and the imperative to analyze it efficiently have spurred the emergence of numerous methodologies for data analysis [1]. Among these methodologies is text mining, a process that entails extracting valuable insights from unstructured or semi-structured data. Text mining finds utility across a diverse array of applications, including sentiment analysis, recommendation systems, and content analysis [2]. Within the realm of text mining, a crucial undertaking involves the extraction of keywords along with their corresponding definitions [3]. Keywords represent terms or

P. Pareek et al. (Eds.): IC4S 2023, LNICST 536, pp. 340–349, 2024.
https://doi.org/10.1007/978-3-031-48888-7_29

phrases that hold significance within a specific subject area or field, while their definitions offer a succinct and accurate explanation of their significance. Extracting keywords serves the purpose of identifying the most pertinent topics addressed in a single document or a group of documents [4]. Moreover, the definitions associated with these keywords enhance comprehension and foster a deeper understanding of the concepts under examination.

The internet has witnessed an unprecedented surge in digital content and data, presenting a formidable challenge in efficiently accessing the most pertinent information amidst this vast volume of data. Keyword extraction serves as a foundational method in NLP, enabling the identification of the crucial words or set of words (phrases) within a text corpus [5]. This process of identifying and extracting relevant keywords holds immense value in applications like information retrieval, text classification, and summarization. In our work, we delve into the utilization of the TextRank algorithm for keyword extraction, as well as the identification of prevailing defining patterns to facilitate definition extraction [6]. This paper presents a novel approach to extract keywords and their corresponding definitions from textual data. Our proposed methodology harnesses the power of the TextRank algorithm, an unsupervised graph-based ranking algorithm renowned for identifying significant terms within a document. By integrating regular expressions, we leverage common patterns in keyword definitions. To assess the effectiveness of our methodology, we conduct evaluations on a dataset comprising arXiv paper abstracts, comparing its performance against other cutting-edge keyword extraction techniques.

This study holds great importance as it has the potential to enhance the effectiveness and precision of keyword extraction and definition extraction tasks. The outcomes of this research offer valuable insights that can significantly benefit various natural language processing applications, including text classification, summarization, and sentiment analysis. Additionally, the proposed approach demonstrates its versatility and broad applicability by being adaptable to diverse domains and languages.

The study is driven by the subsequent research inquiries.

1. To what amount can the TextRank algorithm be utilized for effective keyword extraction?
2. What are the prevailing patterns for definition extraction achieved through regular expressions?
3. How does the proposed approach compare to other advanced keyword extraction methods?
4. What are the potential implications of the findings?

The rest of this paper is organized as follows: Sect. 2 provides an extensive literature review on keyword extraction. Section 3 outlines the details of our proposed methodology. Following that, Sect. 4 presents the experimental results. Finally, in Sect. 5, the paper concludes by summarizing the findings and suggesting potential directions for future research.

2 Related Work

Keyword extraction plays a pivotal role in natural language processing by identifying the most crucial words or phrases within a given text [7]. Multiple methodologies have been devised for this purpose, encompassing statistical, linguistic, and graph-based approaches [4]. Keyword extraction techniques can be categorized into supervised, semi-supervised, or unsupervised methods [8]. One commonly used unsupervised technique is based on TF-IDF, which establishes a baseline by scoring and selecting key-phrases according to their TF-IDF values [9]. Another approach for topic modeling is Latent Dirichlet Allocation (LDA), which performs unsupervised learning to identify the main topics present in a document [10]. In a study conducted by Gu Yijun et al., LDA was utilized to extract document keywords, and their association with keywords within the document itself was found to enhance the results of keyword extraction [11]. Additionally, Rapid Automatic Keyword Extraction (RAKE) is a widely employed algorithm for domain-independent keyword extraction [12]. Among the graph-based techniques, the TextRank algorithm has gained considerable popularity. It operates by analyzing word co-occurrences to determine the significance of individual words within the text. TextRank is an influential ranking algorithm that operates on the principles of graph-based analysis, leveraging the PageRank algorithm [6]. By constructing a graph representation of a given text, where individual keywords or keyphrases serve as nodes of the graph, and establishing connections between nodes that co-occur within the text, TextRank calculates the importance score of each node. This score is determined by applying the PageRank algorithm, which assesses the significance of a node based on the quantity and quality of incoming edges it possesses [13].

In the field of NLP, a notable obstacle entails the identification and extraction of definitions for words or phrases within a given text. To address this challenge, one approach involves identifying prevalent defining patterns commonly employed to introduce definitions, such as the format "A [word] is a [definition]." By utilizing regular expressions, these patterns can be automatically recognized, enabling the extraction of definitions from the text.

Numerous studies have delved into the realm of keyword extraction and definition extraction techniques. However, many of these studies suffer from limitations in their scope, failing to adequately address the challenges encountered in real-world applications. For instance, certain studies rely on small datasets that fail to capture the diverse range of texts and writing styles found in real-world scenarios [14]. Additionally, some studies focus exclusively on specific text types, such as scientific papers, which restricts their applicability to other text genres [15]. Furthermore, certain studies employ outdated or less effective algorithms for keyword extraction and definition extraction, thereby diminishing their overall efficacy [16].

3 Methodology

The research design adopted for this study encompasses a comprehensive and systematic approach to address the objectives of extracting domain-specific keywords and their corresponding definitions from a large corpus of research papers. We have employed a

data-driven methodology, leveraging advanced NLP techniques and machine learning algorithms to systematize the extraction process and ensure scalability. The methodology followed is present the Fig. 1.

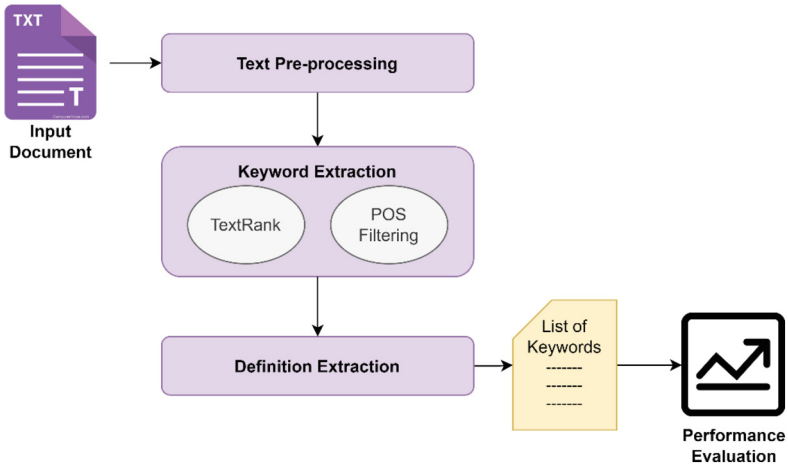

Fig. 1. A framework to identify domain-specific keywords within a given document.

3.1 Data Acquisition and Pre-Processing

To gather the necessary data for our research, we have curated a robust and diverse dataset sourced from the ArXiv database[1]. The ArXiv database hosts a vast collection of research papers spanning multiple disciplines, thereby providing a rich and extensive source of technical content. The dataset was meticulously selected to ensure its relevance and representativeness, enabling us to perform a comprehensive extraction of technical terms which are domain-specific.

Data preprocessing plays a crucial role in any NLP task as it aims to convert raw text data into a suitable format for analysis [17, 18]. This essential stage involves several key steps. Firstly, non-relevant characters, stopwords, and punctuation marks are eliminated. Additionally, tokenization is performed, breaking the text into individual words or phrases. Following this, part-of-speech tagging is applied to determine the grammatical role of each token. This valuable information is utilized to construct a co-occurrence matrix, which captures the frequency of term appearances within the same sentence as other terms. Ultimately, the generated co-occurrence matrix is utilized as the input for the TextRank algorithm. Figure 2 illustrates the sequential steps undertaken during text preprocessing in this work.

Text normalization is a crucial step in achieving consistency within textual data. It encompasses various operations such as eliminating special characters, converting text to lowercase, and expanding contractions [19]. The process involves segmenting the text

[1] https://www.kaggle.com/datasets/Cornell-University/arxiv.

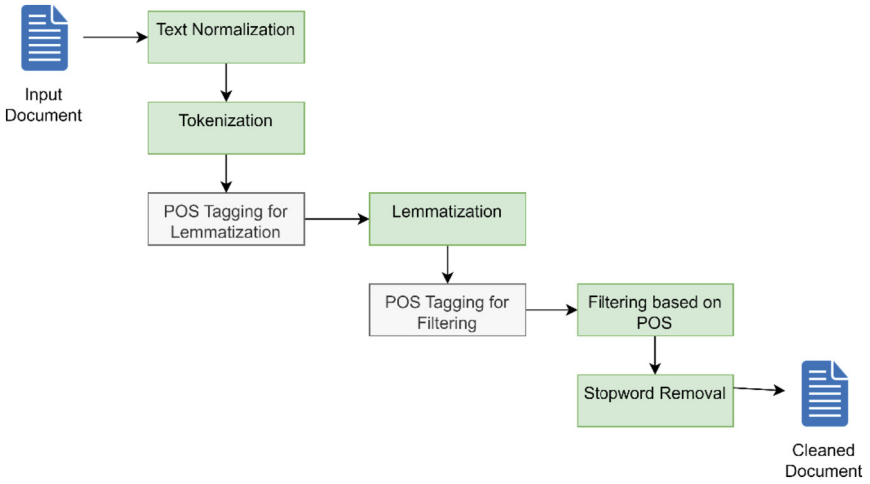

Fig. 2. The pre-processing techniques adopted in transforming the raw text to a clean format.

into individual units or tokens, which can range from words and phrases to individual characters. These tokens serve as the foundation for subsequent processing steps, such as part-of-speech tagging or constructing a co-occurrence matrix. In this particular study, the input text (referred to as "Abstract data") was normalized through the removal of non-printable characters, converting all text to lowercase, eliminating special characters, and removing excessive spaces.

POS tagging and lemmatization are performed in tandem, in the preprocessing of text data for NLP tasks. POS tagging entails assigning a part-of-speech label, such as noun, verb, or adjective, to each word in the text [20]. This labeling information plays a crucial role in lemmatization, which involves transforming each word into its base or dictionary form. For instance, through the process of POS tagging and lemmatization, the word "running" would be converted to "run". This combined approach aids in simplifying the complexity of the text data and enhancing the accuracy of subsequent processing tasks. In this study, the WordNetLemmatizer was utilized to perform lemmatization on the tokens generated in the preceding step.

Filtering the words, by identifying and retaining only certain parts of speech, such as nouns, adjectives and gerunds to improve the relevance and accuracy. Any word in the lemmatized text that does not fall into the categories of noun, adjective, gerund, or a foreign word is classified as a stopword (non-content). Based on the specified conditions, a filter is added to the lemmatized and POS tagged tokens to filter out the non-content (stopwords). This filter includes the POS tags such as *NN, NNS, NNP, NNPS, JJ, JJR, JJS, VBG* and *FW*.

3.2 Keyword Extraction Using TextRank Algorithm and POS Filtering

To identify the most significant technical and domain-specific keywords from research papers, we employed a combined approach of the TextRank algorithm and POS filtering. TextRank, a graph-based ranking algorithm, enables the extraction of important

terms by analyzing their co-occurrence patterns within the document [6]. Leveraging the inherent structure and word relationships, TextRank effectively identifies the essential concepts and ideas discussed in the research papers. In conjunction with TextRank, we implemented POS filtering [21] to further refine the extracted keywords. By considering specific parts of speech commonly associated with technical terms, such as nouns, adjectives, and verb forms, we filtered out irrelevant words while retaining those more likely to hold technical significance. This additional filtering step significantly enhances the precision and accuracy of the extracted keywords, ensuring their close alignment with the technical domain being investigated.

3.3 Definition Extraction Using Regular Expression

After obtaining the extracted keywords, the subsequent task was to retrieve their corresponding definitions from the research papers. To achieve this, we employed the use of regular expressions, a powerful tool for pattern matching, to identify common textual patterns that indicate the presence of definitions. Regular expressions enable us to capture specific structures and linguistic cues within the text that are typically associated with definitions. Through thorough analysis and domain expertise, we meticulously designed a set of well-crafted regular expressions tailored to the specific context of technical literature. These patterns encompass diverse sentence structures, syntactic cues, and linguistic patterns commonly employed in technical definitions. By matching these predefined patterns with the surrounding text, we successfully extracted the relevant definitions for the identified keywords.

3.4 Performance Assessment

To assess the effectiveness and reliability of our methodology, we conducted a comprehensive assessment using a variety of evaluation metrics and statistical analyses. The accuracy and coverage of the extracted keywords were evaluated by comparing them to manually created reference glossaries. Precision, recall, and F1-score were calculated as quantitative measures to assess the performance of the keyword extraction process.

For evaluating the extracted definitions, we adopted a multi-faceted approach. Firstly, a sample of extracted definitions underwent meticulous manual evaluation by domain experts who assessed their accuracy and relevance. Their expert insights provided valuable feedback and ensured the quality of the extracted definitions. Additionally, we utilized semantic similarity measures, such as cosine similarity, to compare the extracted definitions with reference definitions available in external resources. This analysis allowed us to estimate the degree of alignment between our extracted definitions and established definitions, further enhancing the evaluation process.

4 Experimentation and Results

In Glossary Term Extraction, training data is commonly provided as a substantial corpus of text documents, such as books or articles, that encompass the domain-specific terminology of interest. In our experimentation, we utilized straightforward abstracts that

encompassed multiple keywords. By leveraging this training data, the Textrank algorithm was employed to identify and rank the most significant terms within a new text document, considering their frequency and co-occurrence with other crucial terms. This automated approach facilitates the extraction of relevant terminology and streamlines the creation of a comprehensive glossary.

The computation of graph-based ranking, which considers edge weights when determining the score allied with each vertex in the graph, is performed using the formula outlined in the provided Eq. 1. Where, E and V are set of edges and vertices, $In(v_i)$ set of vertices that point to it, $Out(v_i)$ set of vertices that vertex v_i points to. In our approach, we establish a co-occurrence relationship, wherein lexical units are connected as vertices in the graph if they co-occur within a specific word space, with the maximum number of words allowed. This window size can be adjusted, typically ranging from 2 to 10 words.

$$WS(V_i) = (1 - d) + d * \sum_{V_j \in In(V_i)} \frac{w_{ji}}{\sum_{V_k \in Out(V_j)} w_{jk}} WS(V_j) \tag{1}$$

Once the graph is constructed, each vertex is initially assigned a score of 1. The ranking algorithm mentioned earlier is then executed on the graph for multiple iterations until convergence, typically around 20 to 30 iterations, with a threshold of *1e-4*. Once scores are assigned to all the keywords, we consider the top one-third of keywords for further analysis and consideration.

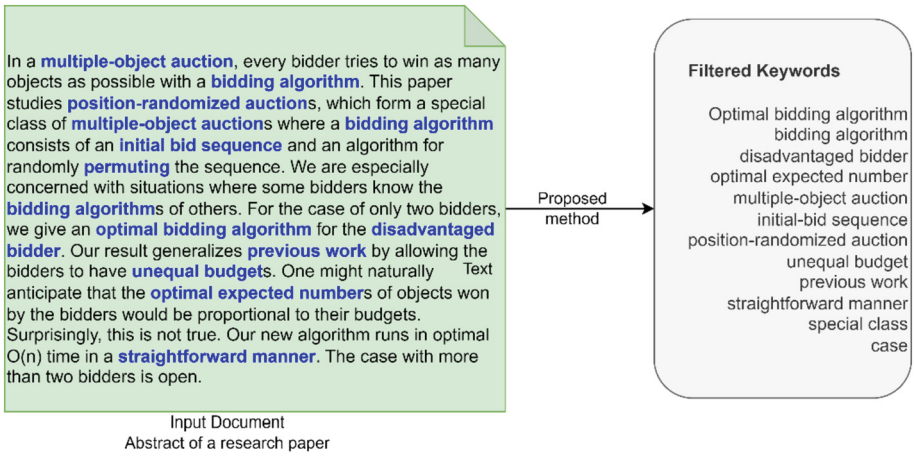

Fig. 3. Keyword extraction

To evaluate the performance of our model, it is essential to have either a predefined list of keywords or a list of annotated keywords. In our experimentation, we manually annotated a list of keywords for this purpose. We then compare these annotated keywords with the filtered keywords generated by the model. Figure 3 presents both the annotated keywords and filtered keywords for a sample abstract. By comparing both lists of keywords, we calculate the confusion matrix, which provides valuable insights. From the

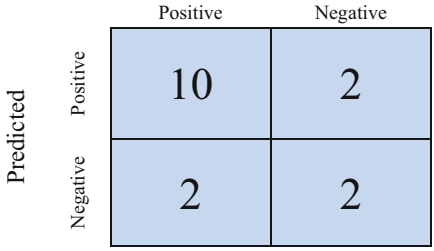

Fig. 4. Confusion matrix

values obtained in the confusion matrix, we can derive metrics such as Precision, Recall, and F1-scores. These metrics allow us to assess the accuracy and effectiveness of our model. The confusion matrix is shown in the Fig. 4. The obtained accuracy is 83%. A sample input, annotated keywords and the filtered keywords are shown in Fig. 3. The words and phrases highlighted in blue are the annotated keywords by the domain expert and the filtered words are present in the right of the figure. Most of the keywords are matching the annotated keywords as conveyed in table. Figure 5 presents the outcome of the definition extraction. For the supplied input document, the word and the definitions will come as *key: value* pairs in the form of a json file.

Cryptology is the study of codes and ciphers, both in terms of their creation and their decoding, also known as cryptanalysis. Cryptography involves the creation of secure communication channels by converting messages into unintelligible forms, which can only be understood by those who possess the key to decoding them. This is achieved by using mathematical algorithms to convert plaintext into ciphertext. Cryptanalysis, on the other hand, is the process of breaking down encrypted messages without knowledge of the key.

Definition Extraction

{
'***cryptanalysis***' : 'the process of breaking down encrypted messages without knowledge of the key',
'***cryptology***': 'the study of codes and ciphers, both in terms of their creation and their decoding, also known as cryptanalysis'
}

Fig. 5. Domain-specific definition extraction

The significance of our research lies in its practical implications. The extracted domain-related keywords can be utilized in various applications, including information retrieval, content analysis, and document categorization within specific domains. This can greatly improve the efficiency of these tasks and provide valuable insights for domain experts.

5 Conclusion

In this work, we presented a comprehensive framework for domain-related keyword extraction, incorporating the TextRank algorithm and several NLP approaches. Our approach not only focused on keyword extraction but also extended to definition extraction,

aiming to provide a more comprehensive understanding of domain-specific content. Through extensive experimentation on arXiv dataset, we evaluated the performance of our proposed method. By comparing the manually annotated keywords with the extracted keywords, we achieved an impressive accuracy of 83%. This demonstrates the effectiveness and reliability of our approach in identifying relevant terms within a specific domain. With a remarkable accuracy of 83% in keyword extraction and the ability to extract definitions, our proposed method showcases its efficacy in capturing domain-specific terms accurately. We anticipate that our work will contribute to advancing the field of keyword extraction and provide valuable insights for domain experts in diverse industries.

References

1. Daniel, B.K.: Big Data and data science: a critical review of issues for educational research. Br. J. Edu. Technol. **50**(1), 101–113 (2019)
2. Allahyari, M., et al.: A brief survey of text mining: classification, clustering and extraction techniques. arXiv preprint arXiv:1707.02919 (2017)
3. Campos, R., Mangaravite, V., Pasquali, A., Jorge, A., Nunes, C., Jatowt, A.: YAKE! Keyword extraction from single documents using multiple local features. Inf. Sci. **509**, 257–289 (2020)
4. Bharti, S.K., Babu, K.S.: Automatic keyword extraction for text summarization: a survey. arXiv preprint arXiv:1704.03242 (2017)
5. Liu, D., Li, Y., Thomas, M.A.: A roadmap for natural language processing research in information systems (2017)
6. Pan, S., Li, Z., Dai, J.: An improved TextRank keywords extraction algorithm. In: Proceedings of the ACM Turing Celebration Conference-China, pp. 1–7 (2019)
7. Firoozeh, N., Nazarenko, A., Alizon, F., Daille, B.: Keyword extraction: issues and methods. Nat. Lang. Eng. **26**(3), 259–291 (2020)
8. Thushara, M.G., Mownika, T., Mangamuru, R.: A comparative study on different keyword extraction algorithms. In: 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC). IEEE (2019)
9. Grineva, M., Grinev, M., Lizorkin, D.: Extracting key terms from noisy and multitheme documents. In: Proceedings of the 18th International Conference on World Wide Web (2009)
10. Mulukutla, V. K., et al.: Sentiment analysis of Twitter data on 'The Agnipath Yojana'. In: Morusupalli, R., Dandibhotla, T.S., Atluri, V.V., Windridge, D., Lingras, P., Komati, V.R. (eds.) Multi-disciplinary Trends in Artificial Intelligence. MIWAI 2023. Lecture Notes in Computer Science, vol. 14078. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-36402-0_50
11. Yijun, G., Tian, X.: Study on keyword extraction with LDA and TextRank combination. Data Anal. Knowl. Discov. **30**(7), 41–47 (2014)
12. Rose, S., et al.: Automatic keyword extraction from individual documents. In: Text Mining: Applications and Theory, pp. 1–20 (2010)
13. Florescu, C., Caragea, C.: A position-biased pagerank algorithm for keyphrase extraction. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 31, no. 1. (2017)
14. Yang, F., Zhu, J., Lun, J., Zheng, Z., Tang, Y., Wu, J.: A keyword-based scholar recommendation framework for biomedical literature. In: 2018 IEEE 22nd International Conference on Computer Supported Cooperative Work in Design (CSCWD), pp. 247–252. IEEE (2018)
15. Li, S., et al.: DuIE: a large-scale Chinese dataset for information extraction. In: Tang, J., Kan, MY., Zhao, D., Li, S., Zan, H. (eds.) Natural Language Processing and Chinese Computing: 8th CCF International Conference, NLPCC 2019, Dunhuang, China, 9–14 October 2019, Proceedings, Part II, vol. 8, pp. 791–800. Springer, Heidelberg (2019). https://doi.org/10.1007/978-3-030-32236-6_72

16. Liang, H., Sun, X., Sun, Y., Gao, Y.: Text feature extraction based on deep learning: a review. EURASIP J. Wirel. Commun. Netw. **2017**(1), 1–12 (2017)
17. Anandarajan, M., et al.: Text preprocessing. Practical text analytics: Maximizing the value of text data, pp. 45–59 (2019)
18. Silpa, N., Rao, V.M.M.: Machine learning-based optimal segmentation system for web data using genetic approach. J. Theor. Appl. Inf. Technol. **100**(11) (2022)
19. Millstein, F.: Natural language processing with python: natural language processing using NLTK. Frank Millstein (2020)
20. Kumawat, D., and Jain, V.: POS tagging approaches: a comparison. Int. J. Comput. Appl. **118**(6) (2015)
21. Liu, F., et al.: Unsupervised approaches for automatic keyword extraction using meeting transcripts. In: Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics (2009)