CROSSAGE: A cross-attentional graph and Transformer architecture for skill and knowledge recognition in job descriptions

Antônio dos Santos Ramos Neto[1], **João Paulo Felix**[1], **Wylliams Santos**[1], **Byron Leite Dantas Bezerra**[1], **Cleyton Mário de Oliveira Rodrigues**[1]

[1]Escola Politécnica de Pernambuco – Universidade de Pernambuco (UPE)

Abstract. *Automatically extracting skills and knowledge from job descriptions supports recruitment, reskilling, and labor market analysis, yet traditional NER models struggle with ambiguous and syntactically complex spans. This work proposes CROSSAGE, a lightweight hybrid architecture that combines contextual embeddings from Transformers with structural features from dependency graphs via cross-attention. Results on the SKILLSPAN dataset show that CROSSAGE with JobSpanBERT achieves the highest F1 for* `SKILL` *entities (49.8), while CROSSAGE (BERT) matched the best baseline for* `KNOWLEDGE` *(64.1) and improves recall (68.8). Gains are especially notable in complex domains like* `house`*, where CROSSAGE reaches 51.5 F1 for* `SKILL`*. These findings highlight CROSSAGE's potential as an effective alternative to heavier hybrid models.*

Resumo. *A extração automática de habilidades e conhecimentos a partir de descrições de vagas apoia o recrutamento, a requalificação e a análise do mercado de trabalho, mas modelos tradicionais de NER enfrentam dificuldades com spans ambíguos e sintaticamente complexos. Este trabalho propõe o CROSSAGE, uma arquitetura híbrida leve que combina embeddings contextuais de Transformers com características estruturais de grafos de dependência por meio de atenção cruzada. Os resultados no dataset SKILLSPAN mostram que o CROSSAGE (JobSpanBERT) atinge o maior F1 para entidades do tipo* `SKILL` *(49,8), enquanto o CROSSAGE (BERT) iguala a melhor baseline para* `KNOWLEDGE` *(64,1) e melhora o recall (68,8). Os ganhos são especialmente notáveis em domínios complexos como* `house`*, onde o CROSSAGE alcança 51,5 de F1 para* `SKILL`*. Esses achados destacam o potencial do CROSSAGE como uma alternativa eficaz a modelos híbridos mais pesados.*

1. Introduction

The automatic detection of skills and knowledge in job descriptions has become an essential tool to enhance professional matching processes. By identifying technical and behavioral competencies in unstructured texts, these systems support candidate screening, optimize talent allocation, and enable the analysis of training gaps relevant to reskilling strategies [Senger et al. 2024]. This is especially important amid rapid changes in the labor market caused by digitalization and evolving occupational demands.

In a broader perspective, automated skill extraction at scale allows continuous tracking of emerging trends across sectors, informing decisions by policymakers, educational institutions, and companies [Tamburri et al. 2020]. This strategic capability helps

reduce mismatches between labor supply and demand, improving alignment between professional profiles and organizational needs.

Practical applications include job matching, resume analysis, and education. Models like JobSpanBERT enhance skill recognition in job recommendation systems [Zhang et al. 2022b], while taxonomies such as ESCO support reskilling by identifying gaps and suggesting learning paths [Zhang et al. 2022a]. Skill extraction also enables forecasting of future labor demands.

However, skill identification is challenging due to entity sparsity in training data, domain-specific terminology, and ambiguity of terms (e.g., *Java*, *Agile*, *Security*). Complex expressions like *advanced statistical modeling* require syntactic awareness, while job descriptions often mix technical and behavioral language in compound structures (e.g., *Docker, Kubernetes, and Jenkins*), making skill extraction more demanding than traditional NER.

Transformer-based models such as BERT [Devlin et al. 2019] are widely used in NER but face limitations when applied to skill recognition. Skills are often compositional, context-dependent, and distributed across complex structures, which undermines the performance of sequential models [Zhang et al. 2022b]. Moreover, Transformers struggle to capture long-range dependencies and syntactic relations between tokens [Senger et al. 2024], and even domain-adapted models like JobBERT show difficulties with subtle or behavioral competencies [Decorte et al. 2021].

To overcome these issues, hybrid architectures have emerged, combining contextual embeddings with structural representations [Zhang 2024]. Entities that are composed, overlapping, or distributed can be more effectively modeled through the integration of Transformer-based self-attention and relational reasoning. Graph neural networks (GNNs), in particular, have shown promise in representing structural relationships derived from syntax, semantics, or co-occurrence [Senger et al. 2024], improving detection of fragmented or syntactically dependent entities.

Furthermore, models that implement cross-attention between Transformer-based and GNN-based representations—such as GNET [Xiang et al. 2024] and BERT-GT [Lai and Lu 2020]—have shown improved performance in complex NER tasks. Cross-attention enables a dynamic combination of contextual and structural perspectives, capturing dependencies that self-attention alone might overlook.

In this context, this paper presents the CROSSAGE model, a hybrid architecture that combines contextual embeddings from Transformers with structural signals from syntactic graphs via cross-attention to enhance skill and knowledge recognition in job descriptions. Our contributions include: (i) the proposal of the CROSSAGE architecture, which combines contextual and structural information via cross-attention; (ii) an initial empirical evaluation on the SKILLSPAN dataset, comparing CROSSAGE variants to strong transformer-based baselines; and (iii) a discussion of observed strengths and limitations to guide future improvements and research on structure-aware NER in labor market applications.

2. Background

2.1. NER for skill and knowledge recognition

Named entity recognition (NER) plays a key role in identifying fine-grained competencies such as skills and knowledge within job-related texts. Unlike traditional NER domains—such as news or biomedical corpora—this context requires handling nested, overlapping, and often ambiguous spans that reflect complex human capabilities. Addressing these challenges depends not only on suitable model architectures, but also on access to high-quality annotated data tailored to this domain.

The SKILLSPAN dataset [Zhang et al. 2022b] provides span-level annotations for `SKILL` and `KNOWLEDGE` entities across three domains—BIG, HOUSE, and TECH—totaling over 232k tokens. It supports nested and overlapping spans and was annotated using ESCO-based guidelines, achieving Fleiss' κ above 0.70 [Zhang 2024]. Models trained under single-task learning (STL) consistently outperformed multi-task setups, reflecting the distinct syntactic and semantic patterns of each entity type.

Span-based models achieved superior performance on SKILLSPAN compared to token-level baselines. JobSpanBERT reached the highest F1-score for `SKILL` spans (56.64), while JobBERT led in `KNOWLEDGE` spans (63.88) under STL [Zhang et al. 2022b]. JobSpanBERT showed better span precision, particularly for longer and syntactically complex expressions, whereas JobBERT obtained higher recall in identifying knowledge spans.

Hybrid models such as NNOSE [Zhang et al. 2024], which combines JobBERTa with a k-nearest neighbor memory and whitening transformation, achieved the best overall span-level F1-score (64.24). It also improved generalization in cross-dataset evaluations. Complementary strategies—like ESCO-guided pretraining [Zhang et al. 2023], weak supervision with concept similarity [Clavié and Soulié 2023], and few-shot prompting with LLMs [Nguyen et al. 2024]—offered promising alternatives for low-resource or multilingual contexts, though none outperformed supervised span models on SKILLSPAN.

2.2. Components in training NER models

Beyond model architecture, several training components strongly influence NER performance, including attention mechanisms, loss functions, and optimization strategies. NER models leverage self-attention to capture long-range dependencies, with Transformers like BERT using multi-head attention for richer context modeling [Sun et al. 2019, Li et al. 2018]. In hybrid setups, cross-attention enables interaction between Transformer and GNN representations, enhancing semantic and structural understanding [Bajestani et al. 2024, Hu and Weng 2025].

Loss functions such as cross-entropy and focal loss [Li et al. 2022, Dong et al. 2019] guide training, with the latter addressing class imbalance via tunable parameters α and γ. To optimize performance, hyperparameter tuning with bayesian optimization tools like Optuna [Optuna 2025, Abbas et al. 2023] is often used. Early stopping [Wang and Yan 2018] helps prevent overfitting by halting training once validation performance plateaus.

2.3. Graph neural networks in NLP

GNNs have become increasingly popular in NLP for modeling non-sequential structures, where tokens are nodes and edges capture syntactic or statistical relations [Wu et al. 2021, Nikolentzos et al. 2020]. GATs enhance this by assigning attention weights to neighbors, allowing the model to focus on semantically relevant connections [Long et al. 2023, Liu et al. 2022]. This is particularly useful in skill recognition, where certain terms in job descriptions carry more informational weight [Zhou et al. 2020].

Hybrid models that combine GNNs with Transformers leverage both local structure and global context. Cross-attention layers enable GAT-based node embeddings to interact with contextual embeddings from models like BERT [Gao et al. 2025], enriching token representations for NER tasks [Yang and Cui 2021]. In real-world applications, GNNs also outperform sequential models in irregular layouts, as shown by [Carbonell et al. 2021] in structured information extraction from documents with visual features.

2.4. Evaluation metrics for sequence labeling

Model performance in sequence labeling is typically assessed using precision, recall, and F1-score [Zhang et al. 2021, Shaaban et al. 2022]. These metrics can be applied at the token or span level, with span-level evaluation being more stringent due to boundary sensitivity. The F1-score provides a balanced measure, especially important in domains with class imbalance or where both over- and under-prediction are critical.

3. Methodology

3.1. Dataset preparation

We used the SKILLSPAN dataset [Zhang et al. 2022b] as our main benchmark. It contains English job descriptions annotated with span-level `SKILL` and `KNOWLEDGE` entities using the BIO scheme. To align with our CROSSAGE architecture, word-level labels were projected to token-level indices after subword tokenization, and non-head subwords were masked to reduce alignment noise.

To incorporate syntactic structure, we constructed token-level dependency graphs based on word-level parses. These graphs encode head–dependent relations and were used as edge indices in the GAT module. The preprocessing pipeline involved four stages: (i) tokenization and label alignment, (ii) span normalization, (iii) syntactic graph construction, and (iv) filtering of invalid samples.

The dataset is divided into training, development, and test splits, with documents drawn from the `tech` and `house` domains. The training set includes 1,237 `SKILL` and 2,188 `KNOWLEDGE` entities in the `tech` domain, and 984 `SKILL` and 781 `KNOWLEDGE` entities in `house`. The development set contains 1,351 entities in `tech` and 812 in `house`, while the test set includes 1,245 in `tech` and 1,019 in `house`. On average, sentences contain approximately one entity, with `house` examples exhibiting longer job descriptions and more behavioral expressions.

Lexical analysis revealed domain-specific patterns: in the `tech` domain, `KNOWLEDGE` entities are dominated by technical terms like *JavaScript*, *Python*, and *Java*, while `SKILL` entities include expressions such as *communication skills*, *passionate*, and

solving business problems. In contrast, the `house` domain features skills like *motivated* and *proactive*, and knowledge areas such as *Engineering*, *English*, and *Project management*.

3.2. Graph construction

To inject syntactic structure into token representations, we constructed dependency graphs using the `spaCy` [Honnibal and Montani 2017] parser (`en_core_web_sm`), which adopts the annotation scheme of the universal dependencies (UD) [Nivre et al. 2020]. Each token is treated as a node, and directed edges represent head–dependent relations derived from the parsed dependency tree.

Because SKILLSPAN is pre-tokenized, we applied a character-offset alignment procedure to match its tokens with spaCy outputs. A tolerance of two characters was used to handle minor inconsistencies (e.g., spacing or punctuation). The resulting aligned edges form the set $\mathcal{E}_{\text{dep}}$, used to define token-level graphs $G = (V, E)$ for each sentence.

The full graph construction process is detailed in Algorithm 1. Only syntactic dependencies are retained to preserve linguistic interpretability and reduce noise in the input structure.

Algorithm 1: Dependency-based graph construction for SKILLSPAN

Input: Token sequence $\mathcal{T} = [t_1, t_2, \ldots, t_n]$
Output: Edge set $\mathcal{E}_{\text{dep}} \subseteq \{(i, j)\}$
Step 1: Concatenate tokens into sentence string $S = t_1 \parallel t_2 \parallel \ldots \parallel t_n$;
Step 2: Parse S using spaCy to obtain dependency tree D;
Step 3: Initialize edge set $\mathcal{E}_{\text{dep}} = \emptyset$;
Step 4: Align each spaCy token d_i to dataset token t_k via character-level matching;
foreach *token* d_i *in* D **do**
  if d_i *has a head* d_h *and* $i \neq h$ **then**
    Let j and k be the dataset indices aligned to d_h and d_i;
    Add edge (j, k) to $\mathcal{E}_{\text{dep}}$;
return $\mathcal{E}_{dep}$

As illustrated in Figure 1, the resulting graphs expose syntactic dependencies among related tokens—such as in the expression *"A degree in Computer Science or related fields or equivalent practical experience."*, where multiple entities are linked under a shared qualifier. These structured representations can improve the model's ability to capture compositional relationships and long-range dependencies in complex entity spans.

3.3. Model architecture

We propose **CROSSAGE** (Cross-attentional graph and encoder), a hybrid architecture for skill and knowledge recognition that integrates a Transformer encoder, a multi-layer GAT, and a cross-attention mechanism.

The input sequence $X = \{x_1, \ldots, x_T\}$ is first encoded using a pretrained Transformer (e.g., BERT or JobSpanBERT), yielding contextual embeddings $\mathbf{h}_t^{(0)} \in \mathbb{R}^d$. In parallel, we construct a token-level dependency graph $G = (V, E)$, where nodes are

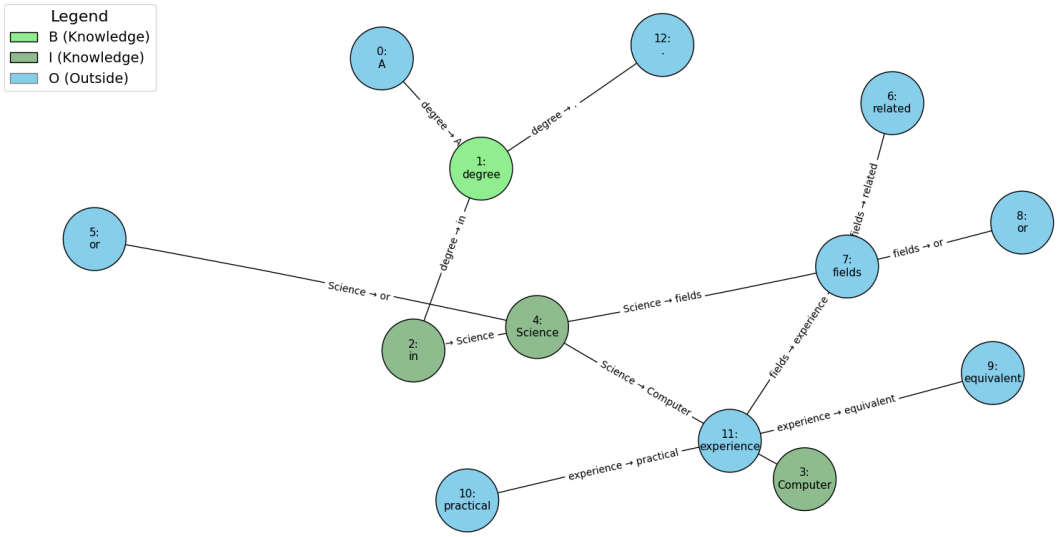

Figure 1. Example of a dependency graph highlighting connections between nested knowledge mentions.

initialized with the same contextual embeddings. A multi-layer GAT updates these via neighbor-aware message passing:

$$\mathbf{h}_{g,i}^{(l)} = \sigma\left(\sum_{j \in \mathcal{N}(i)} \alpha_{ij}^{(l)} \mathbf{W}^{(l)} \mathbf{h}_{g,j}^{(l-1)}\right), \tag{1}$$

where $\alpha_{ij}^{(l)}$ are learned attention coefficients and $\mathbf{W}^{(l)}$ are projection matrices.

To combine structural and semantic information, CROSSAGE applies cross-attention between the contextual ($\mathbf{h}_t^{(l)}$) and graph-based ($\mathbf{h}_g^{(l)}$) representations:

$$\mathbf{h}_t^{(l+1)} = \text{CrossAttn}\left(Q = \mathbf{h}_t^{(l)},\ K = V = \mathbf{h}_g^{(l)}\right) + \mathbf{h}_t^{(l)}, \tag{2}$$

enabling each token to attend over related nodes in the graph. This process is repeated across L layers, producing enriched representations.

The final token embeddings $\mathbf{h}_t^{(L)}$ are mapped to BIO-label logits via a linear layer with softmax:

$$\hat{\mathbf{y}}_i = \text{softmax}(\mathbf{W} \cdot \mathbf{h}_i^{(L)} + \mathbf{b}), \quad \hat{\mathbf{y}}_i \in \mathbb{R}^C \tag{3}$$

To address class imbalance, we use focal loss [Dong et al. 2019]:

$$\mathcal{L}_{\text{focal}} = -\alpha(1 - p_i)^\gamma \log(p_i), \tag{4}$$

where α controls class weighting and γ focuses learning on hard examples.

Table 1. Hyperparameter search space used in Optuna optimization.

Hyperparameter	Type	Search Space
Learning rate (η)	Continuous (log-uniform)	$[1 \times 10^{-5},\ 5 \times 10^{-5}]$
Focal loss (α)	Continuous (linear)	$[0.5,\ 2.0]$
Focal loss (γ)	Continuous (linear)	$[1.0,\ 3.0]$
Number of GAT heads (h)	Integer	$\{2,\ 4\}$
Number of GAT layers (L)	Integer	$\{1,\ 2\}$

CROSSAGE is trained end-to-end, jointly optimizing Transformer, GAT, and attention components. Its modular design supports extension to other sequence labeling or span-based tasks.

3.4. Training procedure

Experiments[1] were conducted on Google Colab Pro [Google 2019] using an NVIDIA T4 GPU (16GB). The implementation used Python with PyTorch 1.13, HuggingFace Transformers 4.31, and PyTorch Geometric 2.3.

The model was trained separately for `SKILL` and `KNOWLEDGE` entities using token-level BIO supervision. The Transformer encoder (BERT or JobSpanBERT) was fine-tuned end-to-end. Training ran for up to 10 epochs with early stopping (patience = 3) based on development span-level F1. Dropout was applied within each cross-attention layer to regularize the attention outputs and prevent overfitting. Specifically, a dropout rate of 0.1 was used after the multi-head attention computation in each cross-attention block.

Focal loss was used to address class imbalance, focusing learning on hard or minority-class examples. Hyperparameter tuning was performed with Optuna [Optuna 2025] to maximize span-level F1 on the dev set. The search space (Table 1) included learning rate, focal loss parameters (α, γ) and GAT settings (number of heads and layers). Eight trials were run per entity type using batch size 16.

Evaluation used span-level F1[2] with the `seqeval` library under the IOB2 scheme, following both strict (exact match) and loose (partial overlap) criteria [Senger et al. 2024]. Models were trained and evaluated strictly on the official SKILLSPAN splits [Zhang et al. 2022b], with all final results reported on the held-out test set. We also performed subgroup analyses on the `tech` and `house` domains to evaluate generalization across stylistic and lexical variation, providing a robust assessment of CROSSAGE in real-world occupational texts.

4. Results and discussions

4.1. Overall performance

Table 2 reports span-level precision, recall, and F1-scores, along with the best hyperparameter configurations found via Optuna for each model–entity pair. We compare the base

[1]Code available at: `https://github.com/tonylincon1/crossage`

[2]In span-level evaluation for NER, a predicted entity is considered correct only if both its label and its span boundaries (start and end positions) exactly match those of a reference entity. Precision is computed as the ratio of correctly predicted spans to the total predicted spans, recall as the ratio of correctly predicted spans to the total reference spans, and the F1-score as the harmonic mean of precision and recall.

Table 2. Span-level results and best Optuna trial configuration per entity and model. Best F1-scores per column are in bold.

Entity	Model	P	R	F1	η	α	γ	Heads	Layers
SKILL	BERT	46.4	**50.5**	48.3	2.43E-5	1.064	1.832	–	–
	CROSSAGE (BERT)	**50.2**	47.4	**48.8**	3.55E-5	1.663	1.880	4	1
	JobSpanBERT	46.5	**51.4**	48.8	3.13E-5	1.568	2.539	–	–
	CROSSAGE (JobSpanBERT)	**51.6**	48.1	**49.8**	4.90E-5	1.125	1.342	4	1
KNOWLEDGE	BERT	**60.9**	67.8	64.1	3.34E-5	0.885	1.103	–	–
	CROSSAGE (BERT)	60.0	**68.8**	64.1	2.51E-5	1.645	1.651	2	1
	JobSpanBERT	57.9	**68.3**	62.7	2.86E-5	1.843	1.550	–	–
	CROSSAGE (JobSpanBERT)	**59.9**	67.2	**63.4**	1.48E-5	1.467	2.251	2	2

Table 3. Span-level results (Precision, Recall, F1) for `SKILL` and `KNOWLEDGE` entities by domain on the test set. Best results per column are in bold.

Model	SKILL						KNOWLEDGE					
	Tech			House			Tech			House		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
BERT	44.4	**51.3**	47.6	47.8	**49.9**	**48.8**	67.5	71.7	69.5	**48.4**	59.2	**53.3**
CROSSAGE (BERT)	**50.0**	49.7	**49.8**	**50.0**	45.7	47.8	**68.7**	**72.8**	**70.7**	45.1	**60.3**	51.6
JobSpanBERT	45.4	**50.9**	**48.0**	47.2	**51.7**	49.4	64.0	**70.9**	**67.3**	46.9	**62.7**	53.7
CROSSAGE (JobSpanBERT)	**48.5**	46.3	47.4	**53.8**	49.4	**51.5**	**64.7**	69.8	67.1	**51.1**	61.9	**56.0**

BERT [Devlin et al. 2019] and JobSpanBERT [Zhang et al. 2022b] models with their corresponding CROSSAGE variants.

For `SKILL` entities, CROSSAGE (JobSpanBERT) achieved the best F1-score (49.8) and precision (51.6), surpassing both its base encoder (48.8 F1) and CROSSAGE (BERT) (48.8 F1). While JobSpanBERT reached the highest recall (51.4), the gains in precision with CROSSAGE highlight the effectiveness of structural integration for improving boundary accuracy.

In the case of `KNOWLEDGE` entities, BERT and CROSSAGE (BERT) both attained the highest F1-score (64.1), though the latter offered improved recall (68.8 vs. 67.8). Similarly, CROSSAGE (JobSpanBERT) yielded a modest F1 gain over its base model (63.4 vs. 62.7), with increased precision but slightly reduced recall. These patterns suggest that CROSSAGE enhances recall and precision trade-offs differently depending on the encoder and entity type.

Hyperparameter tuning revealed that `SKILL` models generally favored higher focal loss γ values, reflecting the difficulty of minority class detection. CROSSAGE variants typically employed shallow GATs (1–2 layers, 2–4 heads), balancing graph signal propagation and over-smoothing.

4.2. Domain-specific evaluation

Table 3 presents span-level performance by domain (`tech` and `house`) for both `SKILL` and `KNOWLEDGE` entities. These domains differ linguistically: `tech` contains concise technical terms, while `house` includes longer, behavior-oriented expressions.

For `SKILL` entities in `tech`, CROSSAGE (BERT) achieved the best F1-score (49.8) and highest precision (50.0), whereas BERT had the highest recall (51.3). In `house`, CROSSAGE (JobSpanBERT) outperformed other models (F1 = 51.5), benefiting from domain-adapted embeddings and structural modeling, especially in handling

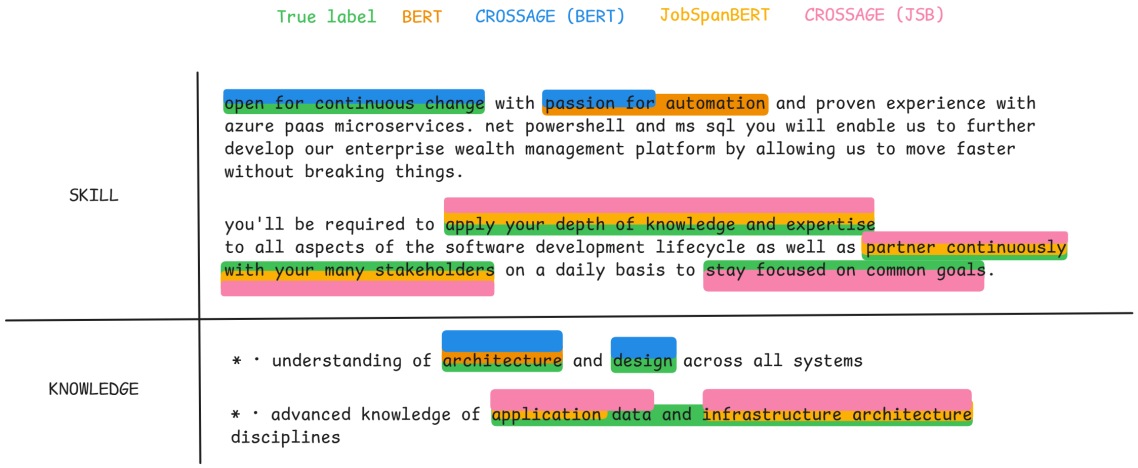

Figure 2. Qualitative comparison of predicted spans across models. Overlaps between true labels (green) and predictions from BERT (orange), CROSSAGE (BERT) (blue), JobSpanBERT (yellow), and CROSSAGE (JSB) (light pink) are shown for selected examples.

abstract or behavioral phrases.

In KNOWLEDGE spans, CROSSAGE (BERT) led performance in the tech domain (F1 = 70.7), highlighting its capacity to detect technical concepts via combined syntactic and contextual cues. Conversely, in house, CROSSAGE (JobSpanBERT) obtained the best F1 (56.0), suggesting better generalization to linguistically diverse or less lexicalized entities.

These findings indicate that hybrid architectures like CROSSAGE are particularly beneficial in domains with higher compositional complexity or abstract spans—especially for SKILL detection in house and KNOWLEDGE recognition in tech.

4.3. Qualitative and error analysis

To illustrate the practical benefits of structural modeling, Figure 2 presents selected examples where CROSSAGE outperformed its base models. These cases were manually chosen to highlight improvements in span boundary alignment and semantic coherence, particularly in multiword expressions.

For instance, in the third example, CROSSAGE (BERT) produced more accurate span predictions in phrases such as *"understand of architeture and design across all systems"*, capturing the full extent of the skill mention more effectively than the standard BERT. Similarly, CROSSAGE (JobSpanBERT) was better at identifying complex expressions like *"application data and infrastructure architecture"*, which often challenge linear models due to their syntactic depth and domain-specific terminology.

These examples suggest that the integration of structural signals enhances the model's ability to recognize semantically dense or behaviorally articulated mentions, even in the presence of ambiguous or lengthy constructs.

4.4. Comparison with related work

Span-based baselines on SKILLSPAN show strong results, with JobSpanBERT reaching 56.64 F1 for SKILL and JobBERT 63.88 for KNOWLEDGE under

STL [Zhang et al. 2022b]. In comparison, CROSSAGE (JobSpanBERT) scored 49.8 for SKILL and CROSSAGE (BERT) 64.1 for KNOWLEDGE, closely matching top models. Our hybrid approach also improves precision on complex spans and recall in domain-specific cases.

While NNOSE [Zhang et al. 2024] attained 64.24 F1 using memory retrieval and whitening, CROSSAGE offers a simpler alternative by leveraging structural cues via cross-attention—achieving competitive results without external memory or ontologies.

5. Conclusion and future work

This paper presented the CROSSAGE model, a hybrid architecture that integrates Transformer-based contextual embeddings with structural information derived from dependency graphs via cross-attention. The model was evaluated on the SKILLSPAN dataset for the task of skill and knowledge recognition in job descriptions, using span-level precision, recall, and F1-score as primary evaluation metrics.

CROSSAGE (JobSpanBERT) achieved the highest F1-score for SKILL entities (49.8), surpassing both its base encoder JobSpanBERT (48.8) and CROSSAGE (BERT) (48.8), with a notable gain in precision (51.6 vs. 46.5). For KNOWLEDGE, CROSSAGE (BERT) reached an F1-score of 64.1—equal to the standard BERT baseline—but offered improved recall (68.8 vs. 67.8), indicating more comprehensive detection of knowledge spans. In domain-specific evaluations, CROSSAGE (BERT) achieved the best F1 for KNOWLEDGE in the tech domain (70.7), while CROSSAGE (JobSpanBERT) led for SKILL in house (51.5), outperforming the base model (49.4). These results suggest that integrating structural signals via cross-attention can enhance boundary accuracy and recall in complex or abstract span contexts.

Although CROSSAGE does not yet outperform memory-augmented models such as NNOSE—whose best reported F1-score on SKILLSPAN reached 64.24 [Zhang et al. 2024]—it offers a lighter-weight alternative that does not rely on external retrieval modules, whitening transformations, or ontological resources. The results validate the potential of graph-based inductive biases to improve fine-grained entity recognition in occupational texts, particularly where entity boundaries are ambiguous.

As future work, we plan to enhance CROSSAGE through improved regularization (e.g., dropout scheduling, graph sparsification, adversarial training) and broader evaluation on datasets like KOMPETENCER [Zhang et al. 2022a] and JOBSTACK [Jensen et al. 2021] to assess generalization across domains and annotation schemes. We also aim to explore multilingual variants using models such as XLM-RoBERTa and mBERT, enabling cross-lingual skill recognition.

Further directions include integrating semantic-enhanced graphs (e.g., from knowledge bases), experimenting with CRF layers for structured decoding, and analyzing efficiency–performance trade-offs via graph pruning. Finally, we intend to expand interpretability and practical validation through attention visualizations.

6. Acknowledgments

This study was financed by the founding public Brazilian agencies CNPq and CAPES (Finance Code 001). In addition, the University of Pernambuco provided full support for this work.

References

Abbas, F., Zhang, F., Ismail, M., Khan, G., Iqbal, J., Alrefaei, A., and Albeshr, M. (2023). Optimizing machine learning algorithms for landslide susceptibility mapping along the karakoram highway, gilgit baltistan, pakistan: a comparative study of baseline, bayesian, and metaheuristic hyperparameter optimization techniques. *Sensors*, 23:6843.

Bajestani, S., Khalilzadeh, M., Azarnoosh, M., and Kobravi, H. (2024). Transentgat: a sentiment-based lexical psycholinguistic graph attention network for personality prediction. *Ieee Access*, 12:59630–59642.

Carbonell, M., Riba, P., Villegas, M., Fornés, A., and Lladós, J. (2021). Named entity recognition and relation extraction with graph neural networks in semi structured documents. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 9622–9627. IEEE.

Clavié, B. and Soulié, G. (2023). Large language models as batteries-included zero-shot esco skills matchers. *arXiv preprint arXiv:2307.03539*.

Decorte, J.-J., Van Hautte, J., Demeester, T., and Develder, C. (2021). Jobbert: Understanding job titles through skills. *arXiv preprint arXiv:2109.09605*.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186.

Dong, X., Chowdhury, S., Qian, L., Li, X., Guan, Y., Yang, J., and Yu, Q. (2019). Deep learning for named entity recognition on chinese electronic medical records: combining deep transfer learning with multitask bi-directional lstm rnn. *Plos One*, 14:e0216046.

Gao, X., Yan, M., Zhang, C., Wu, G., Shang, J., Zhang, C., and Yang, K. (2025). Mdnn-dta: a multimodal deep neural network for drug-target affinity prediction. *Frontiers in Genetics*, 16.

Google (2019). Google colaboratory. `https://colab.research.google.com`. Accessed: 2025-05-24.

Honnibal, M. and Montani, I. (2017). spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Hu, S. and Weng, Q. (2025). Graph-based deep fusion for architectural text representation. *Peerj Computer Science*, 11:e2735.

Jensen, K., Zhang, M., and Plank, B. (2021). De-identification of privacy-related entities in job postings. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics*, United States. Association for Computational Linguistics. NoDaLiDa 2021 ; Conference date: 31-05-2021.

Lai, P.-T. and Lu, Z. (2020). Bert-gt: cross-sentence n-ary relation extraction with bert and graph transformer. *Bioinformatics*, 36(24):5678–5685.

Li, J., Sun, A., Han, J., and Li, C. (2022). A survey on deep learning for named entity recognition. *Ieee Transactions on Knowledge and Data Engineering*, 34:50–70.

Li, Q., Han, Z., and Wu, X. (2018). Deeper insights into graph convolutional networks for semi-supervised learning. *Proceedings of the Aaai Conference on Artificial Intelligence*, 32.

Liu, N., Feng, Q., and Hu, X. (2022). Interpretability in graph neural networks. pages 121–147.

Long, J., Li, Z., Xuan, Q., Fu, C., Peng, S., and Min, Y. (2023). Social media opinion analysis model based on fusion of text and structural features. *Applied Sciences*, 13:7221.

Nguyen, K. C., Zhang, M., Montariol, S., and Bosselut, A. (2024). Rethinking skill extraction in the job market domain using large language models. *arXiv preprint arXiv:2402.03832*.

Nikolentzos, G., Tixier, A., and Vazirgiannis, M. (2020). Message passing attention networks for document understanding. *Proceedings of the Aaai Conference on Artificial Intelligence*, 34:8544–8551.

Nivre, J., de Marneffe, M.-C., Ginter, F., Hajič, J., Manning, C. D., Pyysalo, S., Schuster, S., Tyers, F., and Zeman, D. (2020). Universal Dependencies v2: An evergrowing multilingual treebank collection. In Calzolari, N., Béchet, F., Blache, P., Choukri, K., Cieri, C., Declerck, T., Goggi, S., Isahara, H., Maegaard, B., Mariani, J., Mazo, H., Moreno, A., Odijk, J., and Piperidis, S., editors, *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4034–4043, Marseille, France. European Language Resources Association.

Optuna (2025). Optuna: A hyperparameter optimization framework. `https://optuna.org/`. Accessed: 2025-05-18.

Senger, E., Zhang, M., van der Goot, R., and Plank, B. (2024). Deep learning-based computational job market analysis: A survey on skill extraction and classification from job postings. *arXiv preprint arXiv:2402.05617*.

Shaaban, Y., Korashy, H., and Medhat, W. (2022). Arabic emotion cause extraction using deep learning. *The Egyptian Journal of Language Engineering*, 0:0–0.

Sun, C., Qiu, X., Xu, Y., and Huang, X. (2019). How to fine-tune bert for text classification? pages 194–206.

Tamburri, D. A., Van Den Heuvel, W.-J., and Garriga, M. (2020). Dataops for societal intelligence: a data pipeline for labor market skills extraction and matching. In *2020 IEEE 21st International Conference on Information Reuse and Integration for Data Science (IRI)*, pages 391–394. IEEE.

Wang, W. and Yan, X. (2018). Early stopping criterion combining probability density function with validation error for improving the generalization capability of the back-propagation neural network. *DEStech Transactions on Engineering and Technology Research*.

Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., and Yu, P. (2021). A comprehensive survey on graph neural networks. *Ieee Transactions on Neural Networks and Learning Systems*, 32:4–24.

Xiang, X., Jing, K., and Xu, J. (2024). A neural architecture predictor based on gnn-enhanced transformer. In *International Conference on Artificial Intelligence and Statistics*, pages 1729–1737. PMLR.

Yang, Y. and Cui, X. (2021). Bert-enhanced text graph neural network for classification. *Entropy*, 23:1536.

Zhang, M. (2024). Computational job market analysis with natural language processing. *arXiv preprint arXiv:2404.18977*.

Zhang, M., Jensen, K. N., and Plank, B. (2022a). Kompetencer: Fine-grained skill classification in danish job postings via distant supervision and transfer learning. *arXiv preprint arXiv:2205.01381*.

Zhang, M., Jensen, K. N., Sonniks, S., and Plank, B. (2022b). SkillSpan: Hard and soft skill extraction from English job postings. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4962–4984, Seattle, United States. Association for Computational Linguistics.

Zhang, M., van der Goot, R., Kan, M.-Y., and Plank, B. (2024). Nnose: Nearest neighbor occupational skill extraction. *arXiv preprint arXiv:2401.17092*.

Zhang, M., Van Der Goot, R., and Plank, B. (2023). Escoxlm-r: Multilingual taxonomy-driven pre-training for the job market domain. *arXiv preprint arXiv:2305.12092*.

Zhang, Z., Liu, D., Zhang, M., and Qin, X. (2021). Combining data augmentation and domain information with tener model for clinical event detection. *BMC Medical Informatics and Decision Making*, 21.

Zhou, J., Cui, G., Hu, S., Zhang, Z., Yang, C., Liu, Z., Wang, L., Li, C., and Sun, M. (2020). Graph neural networks: a review of methods and applications. *Ai Open*, 1:57–81.